

**Oracle Functional Testing Advanced
Pack for Oracle Utilities
Reference Guide for Oracle Utilities Work and
Asset Management/ Oracle Utilities Operational
Device Management (v2.1.1)**

Release 5.0.1

E67848-03

November 2016

Oracle Functional Testing Advanced Pack for Oracle Utilities Reference Guide for Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management (v2.1.1), Release 5.0.1

E67848-03

Copyright © 2016 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	i
Audience	i
Related Documents	i
Conventions.....	ii
Abbreviations	ii
Chapter 1	
Component Reference	1-1
Overview	1-1
Oracle Utilities Work and Asset Management Components.....	1-2
Oracle Utilities Operational Device Management Components.....	1-38
Components for BPA Scripts	1-44
Components Not Compatible for Cloud Automation	1-46
Chapter 2	
Function Library Reference	2-1
OUWAMUILIB.....	2-1
Common Methods	2-2
Physical Inventory.....	2-3
Material Request/Issue and Return.....	2-18
Stock Transfer.....	2-39
Blanket Contract.....	2-47
Preventive Maintenance	2-54
Purchase Order.....	2-59
Financial Transaction.....	2-60
Scheduling	2-61
Material Disposition.....	2-77
Material Return	2-78
Chapter 3	
Sample Work Flows	3-1
Sample Flows.....	3-1
W1-ReorderReview_Inventory Flow	3-1
W1-ReplicationOfAssets-ODM Flow	3-2
W1-Material Disposition Flow	3-3
W1-Work Demand Flow	3-4
Executing Sample Flows.....	3-5
Pre-requisites.....	3-5
Setting Up Sample Flows	3-5
Appendix A	
Inbound Web Services	A-1
List of Inbound Web Services	A-1

Preface

This guide describes the Oracle Utilities Work and Asset Management v2.1.1.0 and Oracle Utilities Operational Device Management v2.1.1.0 components and the function libraries used to create those components in Oracle Functional Testing Advanced Pack for Oracle Utilities (OFTAPOU) v5.0.0.1. These components are used to build test flows in Oracle Flow Builder (OFB).

This preface includes the following sections:

- [Audience](#)
- [Related Documents](#)
- [Conventions](#)
- [Abbreviations](#)

Audience

This guide is intended for QA/Test Engineers and Automation Developers to understand the various components and libraries available for them to automate the business test flows for Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management using Oracle Functional Testing Advanced Pack for Oracle Utilities (OFTAPOU) for Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management.

Related Documents

For more information, see the following documents:

- *Oracle Functional Testing Advanced Pack for Oracle Utilities Release Notes*
- *Oracle Functional Testing Advanced Pack for Oracle Utilities Installation and Administration Guide*
- *Oracle Functional Testing Advanced Pack for Oracle Utilities User's Guide*

See also:

- Oracle Utilities Work and Asset Management Documentation Library
- Oracle Utilities Operational Device Management Documentation Library

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Abbreviations

The following terms are used in this document:

- OFB - Oracle Flow Builder
- WAM - Oracle Utilities Work and Asset Management
- ODM - Oracle Utilities Operational Device Management

Chapter 1

Component Reference

This chapter lists the Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management starter components available to create flows in Oracle Flow Builder for testing the Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management applications.

The chapter includes the following sections:

- [Overview](#)
- [Oracle Utilities Work and Asset Management Components](#)
- [Oracle Utilities Operational Device Management Components](#)
- [Components for BPA Scripts](#)
- [Components Not Compatible for Cloud Automation](#)

Overview

Oracle Functional Testing Advanced Pack for Oracle Utilities for Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management is a test starter pack built on top of Oracle Functional Testing Advanced Pack for Oracle Utilities that generates test automation scripts using Oracle Flow Builder.

Oracle Functional Testing Advanced Pack for Oracle Utilities for Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management contains out-of-the-box product-specific components used to build new test flows in Oracle Flow Builder to test the Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management applications. These components correspond to specific business entities, such as business objects, service scripts, or business services used for interfacing with the application. Users can use these components as available or can extend them. Users can also create new components to be used to create flows. This starter pack also contains a set of function libraries that can be used for creating custom components.

Note: See [Chapter 2: Function Library Reference](#) for detailed information about using these function libraries.

Consider this pack to be a starter kit which can be expanded and built upon. A few sample flows are included as an example.

Note: See the *Oracle Functional Testing Advanced Pack for Oracle Utilities User's Guide* for information about creating components and flows.

Oracle Utilities Work and Asset Management Components

This section lists the starter components available in Oracle Utilities Work and Asset Management.

Pre-requisites: The Inbound Web Service using the respective business object should be available in the application.

Additional Notes: Failure while creating, reading, or updating the component is logged to the test execution report, facilitating debugging/analysis of the problem.

The components are categorized under the following functional areas:

- [Administration Data](#)
- [Asset Management](#)
- [General](#)
- [Inventory Management](#)
- [Purchasing](#)
- [Work Management](#)
- [Invoicing](#)
- [Integration](#)

Administration Data

Component	Description
W1-AccountingCalendar	Used to create, update, read, or delete AccountingCalendar in the WAM application. This component calls the W1-AccountingCalendar business object using the Inbound Web Service.
W1-ActivityCompletionEventType	Used to create, update, read, or delete ActivityCompletionEventType in the WAM application. This component calls the W1-ActivityCompletionEventType business object using the Inbound Web Service.
W1- AssetCompletionEventType	Used to create, update, read, or delete AssetCompletionEventType in the WAM application. This component calls the W1-POLineIntegration business object using the Inbound Web Service.
W1- AssetDposCmpEvtType	Used to create, update, read, or delete AssetDposCmpEvtType in the WAM application. This component calls the W1-AssetDposCmpEvtType business object using the Inbound Web Service.

Component	Description
W1- AssetLocationType	<p>Used to create, update, read, or delete AssetLocationType in the WAM application.</p> <p>This component calls the W1- AssetLocationType business object using the Inbound Web Service.</p>
W1- AssetType	<p>Used to create, update, read, or delete AssetType in the WAM application.</p> <p>This component calls the W1-AssetType business object using the Inbound Web Service.</p>
W1- ComponentType	<p>Used to create, update, read, or delete ComponentType in the WAM application.</p> <p>This component calls the W1-ComponentType business object using the Inbound Web Service.</p>
W1- DataAreaCmpEvtType	<p>Used to create, update, read, or delete DataAreaCmpEvtType in the WAM application.</p> <p>This component calls the W1-DataAreaCmpEvtType business object using the Inbound Web Service.</p>
W1- GeneralCompletionEventType	<p>Used to create, update, read, or delete GeneralCompletionEventType in the WAM application.</p> <p>This component calls the W1-GeneralCompletionEventType business object using the Inbound Web Service.</p>
W1-OrganizationType	<p>Used to create, update, read, or delete OrganizationType in the WAM application.</p> <p>This component calls the W1-OrganizationType business object using the Inbound Web Service.</p>
W1- ReplicationComplEventType	<p>Used to create, update, read, or delete ReplicationComplEventType in the WAM application.</p> <p>This component calls the W1-ReplicationComplEventType business object using the Inbound Web Service.</p>
W1-AmtBasedAppvlProf	<p>Used to create, update, read, or delete AmtBasedAppvlProf in the WAM application.</p> <p>This component calls the W1-AmtBasedAppvlProf business object using the Inbound Web Service.</p>

Component	Description
W1-AssessmentClass	Used to create, update, read, or delete AssessmentClass in the WAM application. This component calls the W1-AssessmentClass business object using the Inbound Web Service.
W1-AssessmentGroup	Used to create, update, read, or delete AssessmentGroup in the WAM application. This component calls the W1-AssessmentGroup business object using the Inbound Web Service.
W1-AssetDposCmpEvtType	Used to create, update, read, or delete AssetDposCmpEvtType in the WAM application. This component calls the W1-AssetDposCmpEvtType business object using the Inbound Web Service.
W1-AssetTypeAverageCost	Used to create, update, read, or delete AssetTypeAverageCost in the WAM application. This component calls the W1-AssetTypeAverageCost business object using the Inbound Web Service.
W1-BillableRate	Used to create, update, read, or delete BillableRate in the WAM application. This component calls the W1-BillableRate business object using the Inbound Web Service.
W1-BusinessUnit	Used to create, update, read, or delete BusinessUnit in the WAM application. This component calls the W1-BusinessUnit business object using the Inbound Web Service.
W1-Buyer	Used to create, update, read, or delete Buyer in the WAM application. This component calls the W1-Buyer business object using the Inbound Web Service.
W1-CalendarAnniversary	Used to create, update, read, or delete CalendarAnniversary in the WAM application. This component calls the W1-CalendarAnniversary business object using the Inbound Web Service.
W1-CalendarInterval	Used to create, update, read, or delete CalendarInterval in the WAM application. This component calls the W1-CalendarInterval business object using the Inbound Web Service.

Component	Description
W1-CapabilityType	<p>Used to create, update, read, or delete CapabilityType in the WAM application.</p> <p>This component calls the W1-CapabilityType business object using the Inbound Web Service.</p>
W1-Circle	<p>Used to create, update, read, or delete Circle in the WAM application.</p> <p>This component calls the W1-Circle business object using the Inbound Web Service.</p>
W1-Color	<p>Used to create, update, read, or delete Color in the WAM application.</p> <p>This component calls the W1-Color business object using the Inbound Web Service.</p>
W1-CommodityCategory	<p>Used to create, update, read, or delete CommodityCategory in the WAM application.</p> <p>This component calls the W1-CommodityCategory business object using the Inbound Web Service.</p>
W1-CommodityName	<p>Used to create, update, read, or delete CommodityName in the WAM application.</p> <p>This component calls the W1-CommodityName business object using the Inbound Web Service.</p>
W1-CommodityType	<p>Used to create, update, read, or delete CommodityType in the WAM application.</p> <p>This component calls the W1-CommodityType business object using the Inbound Web Service.</p>
W1-ComponentTypeAverageCost	<p>Used to create, update, read, or delete ComponentTypeAverageCost the in WAM application.</p> <p>This component calls the W1-ComponentTypeAverageCost business object using the Inbound Web Service.</p>
W1-ContactType	<p>Used to create, update, read, or delete ContactType in the WAM application.</p> <p>This component calls the W1-ContactType business object using the Inbound Web Service.</p>
W1-CostCategory	<p>Used to create, update, read, or delete CostCategory in the WAM application.</p> <p>This component calls the W1-CostCategory business object using the Inbound Web Service.</p>

Component	Description
W1-CostCenter	<p>Used to create, update, read, or delete CostCenter in the WAM application.</p> <p>This component calls the W1-CostCenter business object Inbound Web Service.</p>
W1-Course	<p>Used to create, update, read, or delete Course in the WAM application.</p> <p>This component calls the W1-Course business object using the Inbound Web Service.</p>
W1-CrewType	<p>Used to create, update, read, or delete CrewType in the WAM application.</p> <p>This component calls the W1-CrewType business object using the Inbound Web Service.</p>
W1-DailyScheduleType	<p>Used to create, update, read, or delete DailyScheduleType in the WAM application.</p> <p>This component calls the W1-DailyScheduleType business object using the Inbound Web Service.</p>
W1-DistributionCode	<p>Used to create, update, read, or delete DistributionCode in the WAM application.</p> <p>This component calls the W1-DistributionCode business object using the Inbound Web Service.</p>
W1-DocActType	<p>Used to create, update, read, or delete DocActType in the WAM application.</p> <p>This component calls the W1-DocActType business object using the Inbound Web Service.</p>
W1-DocCmplEvtntype	<p>Used to create, update, read, or delete DocCmplEvtntype in the WAM application.</p> <p>This component calls the W1-DocCmplEvtntype business object using the Inbound Web Service.</p>
W1-DocumentType	<p>Used to create, update, read, or delete DocumentType in the WAM application.</p> <p>This component calls the W1-DocumentType business object using the Inbound Web Service.</p>
W1-EmployeeType	<p>Used to create, update, read, or delete EmployeeType in the WAM application.</p> <p>This component calls the W1-EmployeeType business object using the Inbound Web Service.</p>

Component	Description
W1-Equipment	<p>Used to create, update, read, or delete Equipment in the WAM application.</p> <p>This component calls the W1-Equipment business object using the Inbound Web Service.</p>
W1-ExpenseCode	<p>Used to create, update, read, or delete ExpenseCode in the WAM application.</p> <p>This component calls the W1-ExpenseCode business object using the Inbound Web Service.</p>
W1-Factor	<p>Used to create, update, read, or delete Factor in the WAM application.</p> <p>This component calls the W1-Factor business object using the Inbound Web Service.</p>
W1-FactorValue	<p>Used to create, update, read, or delete FactorValue in the WAM application.</p> <p>This component calls the W1-FactorValue business object using the Inbound Web Service.</p>
W1-FailureComponent	<p>Used to create, update, read, or delete FailureComponent in the WAM application.</p> <p>This component calls the W1-FailureComponent business object using the Inbound Web Service.</p>
W1-FailureMode	<p>Used to create, update, read, or delete FailureMode in the WAM application.</p> <p>This component calls the W1-FailureMode business object using the Inbound Web Service.</p>
W1-FailureProfile	<p>Used to create, update, read, or delete FailureProfile in the WAM application.</p> <p>This component calls the W1-FailureProfile business object using the Inbound Web Service.</p>
W1-FailureProfileFailureComp	<p>Used to create, update, read, or delete FailureProfileFailureComp in the WAM application.</p> <p>This component calls the W1-FailureProfileFailureComp business object using the Inbound Web Service.</p>
W1-FailureProfileFailureMode	<p>Used to create, update, read, or delete FailureProfileFailureMode in the WAM application.</p> <p>This component calls the W1-FailureProfileFailureMode business object using the Inbound Web Service.</p>

Component	Description
W1-FailureProfileFailureRepair	<p>Used to create, update, read, or delete FailureProfileFailureRepair in the WAM application.</p> <p>This component calls the W1-FailureProfileFailureRepair business object using the Inbound Web Service.</p>
W1-FailureProfileFailureType	<p>This component can be used to create/update/read/delete FailureProfileFailureType in WAM application.</p> <p>This component calls the business object W1-FailureProfileFailureType using the Inbound Web Service.</p>
W1-FailureRepair	<p>Used to create, update, read, or delete FailureRepair in the WAM application.</p> <p>This component calls the W1-FailureRepair business object using the Inbound Web Service.</p>
W1-FailureType	<p>Used to create, update, read, or delete FailureType in the WAM application.</p> <p>This component calls the W1-FailureType business object using the Inbound Web Service.</p>
W1-FinancialMasterConfig	<p>Used to create, update, read, or delete FinancialMasterConfig in the WAM application.</p> <p>This component calls the W1-FinancialMasterConfig business object using the Inbound Web Service.</p>
W1-Fob	<p>Used to create, update, read, or delete Fob in the WAM application.</p> <p>This component calls the W1-Fob business object using the Inbound Web Service.</p>
W1-Function	<p>Used to create, update, read, or delete Function in the WAM application.</p> <p>This component calls the W1-Function business object using the Inbound Web Service.</p>
W1-FunctionType	<p>Used to create, update, read, or delete FunctionType in the WAM application.</p> <p>This component calls the W1-FunctionType business object using the Inbound Web Service.</p>

Component	Description
W1-GaugeReadWithThresholdMeasType	<p>Used to create, update, read, or delete GaugeReadWithThresholdMeasType in the WAM application.</p> <p>This component calls the W1-GaugeReadWithThresholdMeasType business object using the Inbound Web Service.</p>
W1-GeneralAssetLocationType	<p>This component can be used to create/update/read/delete GeneralAssetLocationType in WAM application. This component calls the business object Inbound Web Service.</p>
W1-GeneralCompletionEventType	<p>Used to create, update, read, or delete GeneralCompletionEventType in the WAM application.</p> <p>This component calls the W1-GeneralCompletionEventType business object using the Inbound Web Service.</p>
W1-InventoryMgmtMasterConfig	<p>Used to create, update, read, or delete InventoryMgmtMasterConfig in the WAM application.</p> <p>This component calls the W1-InventoryMgmtMasterConfig business object using the Inbound Web Service.</p>
W1-LaborEarningType	<p>Used to create, update, read, or delete LaborEarningType in the WAM application.</p> <p>This component calls the W1-LaborEarningType business object using the Inbound Web Service.</p>
W1-LeaveType	<p>Used to create, update, read, or delete LeaveType in the WAM application.</p> <p>This component calls the W1-LeaveType business object using the Inbound Web Service.</p>
W1-Line	<p>Used to create, update, read, or delete Line in the WAM application.</p> <p>This component calls the W1-Line business object using the Inbound Web Service.</p>
W1-LinearAssetLocationType	<p>Used to create, update, read, or delete LinearAssetLocationType in the WAM application.</p> <p>This component calls the W1-LinearAssetLocationType business object using the Inbound Web Service.</p>

Component	Description
W1-MeasurementUOM	Used to create, update, read, or delete MeasurementUOM in the WAM application. This component calls the W1-MeasurementUOM business object using the Inbound Web Service.
W1-MeterReadWithRollOverMeasType	Used to create, update, read, or delete MeterReadWithRollOverMeasType in the WAM application. This component calls the W1-MeterReadWithRollOverMeasType business object using the Inbound Web Service.
W1-MsrtQuckEntryCmplEvtTyp	Used to create, update, read, or delete MsrtQuckEntryCmplEvtTyp in the WAM application. This component calls the W1-MsrtQuckEntryCmplEvtTyp business object using the Inbound Web Service.
W1-MTBFMTTRBuckets	Used to create, update, read, or delete MTBFMTTRBuckets in the WAM application. This component calls the W1-MTBFMTTRBuckets business object using the Inbound Web Service.
W1-NoAppvlRequiredProf	Used to create, update, read, or delete NoAppvlRequiredProf in the WAM application. This component calls the W1-NoAppvlRequiredProf business object using the Inbound Web Service.
W1-NonAmtBasedAppvlProf	Used to create, update, read, or delete NonAmtBasedAppvlProf in the WAM application. This component calls the W1-NonAmtBasedAppvlProf business object using the Inbound Web Service.
W1-NumericGauge	Used to create, update, read, or delete NumericGauge in the WAM application. This component calls the W1-NumericGauge business object using the Inbound Web Service.
W1-NumericMeter	Used to create, update, read, or delete NumericMeter in the WAM application. This component calls the W1-NumericMeter business object using the Inbound Web Service.

Component	Description
W1-OrganizationType	<p>Used to create, update, read, or delete OrganizationType in the WAM application.</p> <p>This component calls the W1-OrganizationType business object using the Inbound Web Service.</p>
W1-OutboundActivityType	<p>Used to create, update, read, or delete OutboundActivityType in the WAM application.</p> <p>This component calls the W1-OutboundActivityType business object using the Inbound Web Service.</p>
W1-Planner	<p>Used to create, update, or delete Planner in the WAM application.</p> <p>This component calls W1-Planner business object using the Inbound Web Service ATW1Planner.</p>
W1-OvertimeType	<p>Used to create, update, read, or delete OvertimeType in the WAM application.</p> <p>This component calls the W1-OvertimeType business object using the Inbound Web Service.</p>
W1-PaymentTerm	<p>Used to create, update, read, or delete PaymentTerm in the WAM application.</p> <p>This component calls the W1-PaymentTerm business object using the Inbound Web Service.</p>
W1-PersonWAM	<p>Used to create, update, read, or delete PersonWAM in the WAM application.</p> <p>This component calls the W1-PersonWAM business object using the Inbound Web Service.</p>
W1-PlantAssetLocationType	<p>Used to create, update, read, or delete PlantAssetLocationType in the WAM application.</p> <p>This component calls the W1-PlantAssetLocationType business object using the Inbound Web Service.</p>
W1-ProjectCategory	<p>Used to create, update, read, or delete ProjectCategory in the WAM application.</p> <p>This component calls the W1-ProjectCategory business object using the Inbound Web Service.</p>
W1-PurchaseCommodity	<p>Used to create, update, read, or delete PurchaseCommodity in the WAM application.</p> <p>This component calls the W1-PurchaseCommodity business object using the Inbound Web Service.</p>

Component	Description
W1-PurchasingMasterConfig	<p>Used to create, update, read, or delete PurchasingMasterConfig in the WAM application.</p> <p>This component calls the W1-PurchasingMasterConfig business object using the Inbound Web Service.</p>
W1-Question	<p>Used to create, update, read, or delete Question in the WAM application.</p> <p>This component calls the W1-Question business object using the Inbound Web Service.</p>
W1-ReadLogEntry	<p>Used to read ReadLogEntry in the WAM application.</p> <p>This component uses the W1-ReadLogEntry business Service using the Inbound Web Service.</p>
W1-ReadWithNoEvaluateMeasType	<p>Used to create, update, read, or delete ReadWithNoEvaluateMeasType in the WAM application.</p> <p>This component calls the W1-ReadWithNoEvaluateMeasType business object using the Inbound Web Service.</p>
W1-ResourceUOM	<p>Used to create, update, read, or delete ResourceUOM in the WAM application.</p> <p>This component calls the W1-ResourceUOM business object using the Inbound Web Service.</p>
W1-ScheduledActivityType	<p>Used to create, update, read, or delete ScheduledActivityType in the WAM application.</p> <p>This component calls the W1-ScheduledActivityType business object using the Inbound Web Service.</p>
W1-ScheduledActivityTypeWAM	<p>Used to create, update, read, or delete ScheduledActivityTypeWAM in the WAM application.</p> <p>This component calls the W1-ScheduledActivityTypeWAM business object using the Inbound Web Service.</p>
W1-SchedulePlanType	<p>Used to create, update, read, or delete SchedulePlanType in the WAM application.</p> <p>This component calls the W1-SchedulePlanType business object using the Inbound Web Service.</p>

Component	Description
W1-ServHistFromActTypeCmplEvtType	<p>Used to create, update, read, or delete ServHistFromActTypeCmplEvtType in the WAM application.</p> <p>This component calls the W1-ServHistFromActTypeCmplEvtType business object using the Inbound Web Service.</p>
W1-ServiceArea	<p>Used to create, update, read, or delete ServiceArea in the WAM application.</p> <p>This component calls the W1-ServiceArea business object using the Inbound Web Service.</p>
W1-ServiceClass	<p>Used to create, update, read, or delete ServiceClass in the WAM application.</p> <p>This component calls the W1-ServiceClass business object using the Inbound Web Service.</p>
W1-ShiftPlanTemplate	<p>Used to create, update, or delete Shift Plan Template in the WAM application.</p> <p>This component calls the business object W1-ShiftPlanTemplate using the Inbound Web Service.</p>
W1-ServiceHistoryType	<p>Used to create, update, read, or delete ServiceHistoryType in the WAM application.</p> <p>This component calls the W1-ServiceHistoryType business object using the Inbound Web Service.</p>
W1-ServiceHistTypeCmpEvtType	<p>Used to create, update, read, or delete ServiceHistTypeCmpEvtType in the WAM application.</p> <p>This component calls the W1-ServiceHistTypeCmpEvtType business object using the Inbound Web Service.</p>
W1-StorageLocationType	<p>Used to create, update, read, or delete StorageLocationType in the WAM application.</p> <p>This component calls the W1-StorageLocationType business object using the Inbound Web Service.</p>
W1-TaxRateSchedule	<p>Used to create, update, read, or delete TaxRateSchedule in the WAM application.</p> <p>This component calls the W1-TaxRateSchedule business object using the Inbound Web Service.</p>
W1-TimePeriod	<p>Used to create, update, read, or delete TimePeriod in the WAM application.</p> <p>This component calls the W1-TimePeriod business object using the Inbound Web Service.</p>

Component	Description
W1-WorkCalendar	<p>Used to create, update, read, or delete WorkCalendar in the WAM application.</p> <p>This component calls the W1-WorkCalendar business object using the Inbound Web Service.</p>
W1-WorkManagementMasterConfig	<p>Used to create, update, read, or delete WorkManagementMasterConfig in the WAM application.</p> <p>This component calls the W1-WorkManagementMasterConfig business object using the Inbound Web Service.</p>
W1-WorkPriorityBuckets	<p>Used to create, update, read, or delete WorkPriorityBuckets in the WAM application.</p> <p>This component calls the W1-WorkPriorityBuckets business object using the Inbound Web Service.</p>
W1-WorkReqCompletionEventType	<p>Used to create, update, read, or delete WorkReqCompletionEventType in the WAM application.</p> <p>This component calls the W1-WorkReqCompletionEventType business object using the Inbound Web Service.</p>
W1-WAssetInstallAgeBuckets	<p>Used to create, update, read, or delete WAssetInstallAgeBuckets in the WAM application.</p> <p>This component calls the W1-WAssetInstallAgeBuckets business object using the Inbound Web Service.</p>
W1-WAssetAgeBuckets	<p>Used to create, update, read, or delete WAssetAgeBuckets in the WAM application.</p> <p>This component calls the W1-WAssetAgeBuckets business object using the Inbound Web Service.</p>
W1-WAMBIExtractParameters	<p>Used to create, update, read, or delete WAMBIExtractParameters in the WAM application.</p> <p>This component calls the W1-WAMBIExtractParameters business object using the Inbound Web Service.</p>
W1-QuestionnaireSvcHistType	<p>Used to create, update, read, or delete QuestionnaireSvcHistType in the WAM application.</p> <p>This component calls the W1-QuestionnaireSvcHistType business object using the Inbound Web Service.</p>

Component	Description
W1-NumberOfCyclesOverdueBucket	<p>Used to create, update, read, or delete NumberOfCyclesOverdueBucket in the WAM application.</p> <p>This component calls the W1-NumberOfCyclesOverdueBucket business object using the Inbound Web Service.</p>
W1-AssetCriticalityBuckets	<p>Used to create, update, read, or delete AssetCriticalityBuckets in the WAM application.</p> <p>This component calls the W1-AssetCriticalityBuckets business object using the Inbound Web Service.</p>
W1-AssetConditionScoreBuckets	<p>Used to create, update, read, or delete AssetConditionScoreBuckets in the WAM application.</p> <p>This component calls the W1-AssetConditionScoreBuckets business object using the Inbound Web Service.</p>
W1-GeneralCompletionEventType	<p>Used to create, update, read, or delete GeneralCompletionEventType in the WAM application.</p> <p>This component calls the W1-GeneralCompletionEventType business object using the Inbound Web Service.</p>
W1- ResourceMgmtMasterConfig	<p>Used to create, update, read, or delete ResourceMgmtMasterConfig in the WAM application.</p> <p>This component calls the W1-ResourceMgmtMasterConfig business object using the Inbound Web Service.</p>

Asset Management

Component	Description
W1-CalendarWarranty	<p>Used to create, update, read, or delete CalendarWarranty in the WAM application.</p> <p>This component calls the W1-CalendarWarranty business object using the Inbound Web Service.</p>
W1-CalendarWarrantyTerm	<p>Used to create, update, read, or delete CalendarWarrantyTerm in the WAM application.</p> <p>This component calls the W1-CalendarWarrantyTerm business object using the Inbound Web Service.</p>

Component	Description
W2-CommunicationComponent	<p>Used to create, update, read, or delete CommunicationComponent in the WAM application.</p> <p>This component calls the W2-CommunicationComponent business object using the Inbound Web Service.</p>
W2-CommunicationRelay	<p>Used to create, update, read, or delete CommunicationRelay in the WAM application.</p> <p>This component calls the W2-CommunicationRelay business object using the Inbound Web Service.</p>
W2-ServicePoint	<p>Used to create, update, read, or delete ServicePoint in the WAM application.</p> <p>This component calls the W2-ServicePoint business object using the Inbound Web Service.</p>
W2-SmartMeter	<p>Used to create, update, read, or delete SmartMeter in the WAM application.</p> <p>This component calls the W2-SmartMeter business object using the Inbound Web Service.</p>
W2-CommunicationComponentSpec	<p>Used to create, update, read, or delete CommunicationComponentSpec in the WAM application.</p> <p>This component calls the W2-CommunicationComponentSpec business object using the Inbound Web Service.</p>
W2-CommunicationRelaySpec	<p>Used to create, update, read, or delete CommunicationRelaySpec in the WAM application.</p> <p>This component calls the W2-CommunicationRelaySpec business object using the Inbound Web Service.</p>
W1-FailureServiceHistory	<p>Used to create, update, read, or delete FailureServiceHistory in the WAM application.</p> <p>This component calls the W1-FailureServiceHistory business object using the Inbound Web Service.</p>
W2-FirmwareSpecification	<p>Used to create, update, read, or delete FirmwareSpecification in the WAM application.</p> <p>This component calls the W2-FirmwareSpecification business object using the Inbound Web Service.</p>

Component	Description
W1-FirmwareVersion	<p>Used to create, update, read, or delete FirmwareVersion in the WAM application.</p> <p>This component calls the W1-FirmwareVersion business object using the Inbound Web Service.</p>
W1-GeneralAssetLocation	<p>Used to create, update, read, or delete GeneralAssetLocation in the WAM application.</p> <p>This component calls the W1-GeneralAssetLocation business object using the Inbound Web Service.</p>
W1-InOutOfServiceGeneralAsset	<p>Used to create, update, read, or delete InOutOfServiceGeneralAsset in the WAM application.</p> <p>This component calls the W1-InOutOfServiceGeneralAsset business object using the Inbound Web Service.</p>
W1-IOSvcGeneralComponent	<p>Used to create, update, read, or delete InOutOfServiceGeneralComponent in the WAM application.</p> <p>This component calls the W1-IOSvcGeneralComponent business object using the Inbound Web Service.</p>
W1-IOSvcFleetAsset	<p>Used to create, update, read, or delete IOSvcFleetAsset in the WAM application.</p> <p>This component calls the W1-IOSvcFleetAsset business object using the Inbound Web Service.</p>
W1-IOSvcInfrastructureAsset	<p>Used to create, update, read, or delete IOSvcInfrastructureAsset in the WAM application.</p> <p>This component calls the W1-IOSvcInfrastructureAsset business object using the Inbound Web Service.</p>
W1-IOSvcITAsset	<p>Used to create, update, read, or delete IOSvcITAsset in WAM the application.</p> <p>This component calls the W1-IOSvcITAsset business object using the Inbound Web Service.</p>
W1-InOutSvcAsset	<p>Used to create, update, read, or delete W1-InOutSvcAsset in WAM the application.</p> <p>This component calls the W1-InOutSvcAsset business object using the Inbound Web Service.</p>

Component	Description
W1-InOutSvcComponent	<p>Used to create, update, read, or delete W1-InOutSvcComponent in WAM the application.</p> <p>This component calls the W1-InOutSvcComponent business object using the Inbound Web Service.</p>
W1-AssetLocation	<p>Used to create, update, read, or delete W1-AssetLocation in WAM the application.</p> <p>This component calls the W1-AssetLocation business object using the Inbound Web Service.</p>
W1-ServiceHistory	<p>Used to create, update, read, or delete ServiceHistory in the WAM application.</p> <p>This component calls the W1-ServiceHistory business object using the Inbound Web Service.</p>
W1-ServiceHistoryWAM	<p>Used to create, update, read, or delete ServiceHistory in the WAM application.</p> <p>This component calls the W1-ServiceHistory(Deficiency/General Weighted/Percent of New)business object using the Inbound Web Service.</p>
W1-StoreroomInventoryMgmt	<p>Used to create, update, read, or delete StoreroomInventoryMgmt in the WAM application.</p> <p>This component calls the W1-StoreroomInventoryMgmt business object using the Inbound Web Service.</p>
W1-TrackedFleetSpecification	<p>Used to create, update, read, or delete TrackedFleetSpecification in the WAM application.</p> <p>This component calls the W1-TrackedFleetSpecification business object using the Inbound Web Service.</p>
W1-TrackedITSpecification	<p>Used to create, update, read, or delete TrackedITSpecification in the WAM application.</p> <p>This component calls the W1-TrackedITSpecification business object using the Inbound Web Service.</p>
W1-TrackedInfrastructureSpec	<p>Used to create, update, read, or delete TrackedInfrastructureSpec in the WAM application.</p> <p>This component calls the W1-TrackedInfrastructureSpec business object using the Inbound Web Service.</p>

Component	Description
W1-MsrmtQuickEntryCmplEvt	<p>Used to create, update, read, or delete MsrmtQuickEntryCmplEvt in the WAM application.</p> <p>This component calls the W1-MsrmtQuickEntryCmplEvt business object using the Inbound Web Service.</p>
W1-IOsvITComponent	<p>Used to create, update, read, or delete IOsvITComponent in the WAM application.</p> <p>This component calls the W1-IOsvITComponent business object using the Inbound Web Service.</p>
W1-LinearAssetLocation	<p>Used to create, update, read, or delete LinearAssetLocation in the WAM application.</p> <p>This component calls the W1-LinearAssetLocation business object using the Inbound Web Service.</p>
W2-ManualMeterSpecification	<p>Used to create, update, read, or delete ManualMeterSpecification in the WAM application.</p> <p>This component calls the W2-ManualMeterSpecification business object using the Inbound Web Service.</p>
W1-Manufacturer	<p>Used to create, update, read, or delete Manufacturer in the WAM application.</p> <p>This component calls the W1-Manufacturer business object using the Inbound Web Service.</p>
W1-MsrmtQuickEntryActivity	<p>Used to create, update, read, or delete MsrmtQuickEntryActivity in the WAM application.</p> <p>This component calls the W1-MsrmtQuickEntryActivity business object using the Inbound Web Service.</p>
W1-Organization	<p>Used to create, update, read, or delete Organization in the WAM application.</p> <p>This component calls the W1-Organization business object using the Inbound Web Service.</p>
W1-PlantAssetLocation	<p>Used to create, update, read, or delete PlantAssetLocation in the WAM application.</p> <p>This component calls the W1-PlantAssetLocation business object using the Inbound Web Service.</p>
W1-QuestionnaireSvcHist	<p>Used to create, update, read, or delete QuestionnaireSvcHist in the WAM application.</p> <p>This component calls the W1-QuestionnaireSvcHist business object using the Inbound Web Service.</p>

Component	Description
W1-RuntimeAnniversary	<p>Used to create, update, read, or delete RuntimeAnniversary in the WAM application.</p> <p>This component calls the W1-RuntimeAnniversary business object using the Inbound Web Service.</p>
W1-RuntimeInterval	<p>Used to create, update, read, or delete RuntimeInterval in the WAM application.</p> <p>This component calls the W1-RuntimeInterval business object using the Inbound Web Service.</p>
W1-RuntimeWarranty	<p>Used to create, update, read, or delete RuntimeWarranty in the WAM application.</p> <p>This component calls the W1-RuntimeWarranty business object using the Inbound Web Service.</p>
W1-RuntimeWarrantyTerm	<p>Used to create, update, read, or delete RuntimeWarrantyTerm in the WAM application.</p> <p>This component calls the W1-RuntimeWarrantyTerm business object using the Inbound Web Service.</p>
W1-ServicePoint	<p>Used to create, update, read, or delete ServicePoint in the WAM application.</p> <p>This component calls the W1-ServicePoint business object using the Inbound Web Service.</p>
W2-SmartMeterSpecification	<p>Used to create, update, read, or delete SmartMeterSpecification in the WAM application.</p> <p>This component calls the W2-SmartMeterSpecification business object using the Inbound Web Service.</p>
W1-StorageLocation	<p>Used to create, update, read, or delete StorageLocation in the WAM application.</p> <p>This component calls the W1-StorageLocation business object using the Inbound Web Service.</p>
W1-TrackedGeneralAsset	<p>Used to create, update, read, or delete TrackedGeneralAsset in the WAM application.</p> <p>This component calls the W1-TrackedGeneralAsset business object using the Inbound Web Service.</p>
W1-TrackedGeneralComponent	<p>Used to create, update, read, or delete TrackedGeneralComponent in the WAM application.</p> <p>This component calls the W1-TrackedGeneralComponent business object using the Inbound Web Service.</p>

Component	Description
W1-TrackedGeneralSpecification	Used to create, update, read, or delete TrackedGeneralSpecification in the WAM application. This component calls the W1-TrackedGeneralSpecification business object using the Inbound Web Service.
W1-TrackedITAsset	Used to create, update, read, or delete TrackedITAsset in the WAM application. This component calls the W1-TrackedITAsset business object using the Inbound Web Service.
W1-TrackedITComponent	Used to create, update, read, or delete TrackedITComponent in the WAM application. This component calls the W1-TrackedITComponent business object using the Inbound Web Service.
W2-Vehicle	Used to create, update, read, or delete Vehicle in the WAM application. This component calls the W2-Vehicle business object using the Inbound Web Service.

General

Component	Description
W1-ApprovalRequest	Used to create, update, read, or delete ApprovalRequest in the WAM application. This component calls the W1-ApprovalRequest business object using the Inbound Web Service.
W1-CreLocationDocumentCmpEvt	Used to create, update, read, or delete CreLocationDocumentCmpEvt in the WAM application. This component calls the W1-CreLocationDocumentCmpEvt business object using the Inbound Web Service.
W1-DocAttachment	Used to create, update, read, or delete DocAttachment in the WAM application. This component calls the W1-DocAttachment business object using the Inbound Web Service.
W1-CreDocumentPayCmpEvt	Used to create, update, read, or delete CreDocumentPayCmpEvt in the WAM application. This component calls the W1-CreDocumentPayCmpEvt business object using the Inbound Web Service.

Component	Description
W1-DocUrl	Used to create, update, read, or delete DocUrl in the WAM application. This component calls the W1-DocUrl business object using the Inbound Web Service.
W1-FinancialTransaction	Used to create, update, read, or delete FinancialTransaction in the WAM application. This component calls the W1-FinancialTransaction business object using the Inbound Web Service.
W1-StandardNote	Used to create, update, read, or delete StandardNote in the WAM application. This component calls the W1-StandardNote business object using the Inbound Web Service.

Inventory Management

Component	Description
W1-AcceptLineMaterial	Used to create, update, read, or delete AcceptLineMaterial in the WAM application. This component calls the W1-AcceptLineMaterial business object using the Inbound Web Service.
W1-AcceptLineCraft	Used to create, update, read, or delete AcceptLineCraft in the WAM application. This component calls the W1-AcceptLineCraft business object using the Inbound Web Service.
W1-StockTransaction	Used to create, update, read, or delete StockTransaction in the WAM application. This component calls the W1-StockTransaction business object using the Inbound Web Service.
W1-BOMGenStockItem	Used to create, update, read, or delete BOMGenStockItem in the WAM application. This component calls the W1-BOMGenStockItem business object using the Inbound Web Service.
W1-IndependentPartBOM	Used to create, update, read, or delete IndependentPartBOM in the WAM application. This component calls the W1-IndependentPartBOM business object using the Inbound Web Service.

Component	Description
W1-InvAdjustInvTracked	<p>Used to create, update, read, or delete InvAdjustInvTracked in the WAM application.</p> <p>This component calls the W1-InvAdjustInvTracked business object using the Inbound Web Service.</p>
W1-InvAdjLotManaged	<p>Used to create, update, read, or delete InvAdjLotManaged in the WAM application.</p> <p>This component calls the W1-InvAdjLotManaged business object using the Inbound Web Service.</p>
W1-InvAdjRegularInventory	<p>Used to create, update, read, or delete InvAdjRegularInventory in the WAM application.</p> <p>This component calls the W1-InvAdjRegularInventory business object using the Inbound Web Service.</p>
W1-MaterialRequestHeader	<p>Used to create, update, read, or delete MaterialRequestHeader in the WAM application.</p> <p>This component calls the W1-MaterialRequestHeader business object using the Inbound Web Service.</p>
W1-MaterialRequestLine	<p>Used to create, update, read, or delete MaterialRequestLine in the WAM application.</p> <p>This component calls the W1-MaterialRequestLine business object using the Inbound Web Service.</p>
W1-MaterialRequestTemplate	<p>Used to create, update, read, or delete MaterialRequestTemplate in the WAM application.</p> <p>This component calls the W1-MaterialRequestTemplate business object using the Inbound Web Service.</p>
W1-MaterialReturnHeader	<p>Used to create, update, read, or delete MaterialReturnHeader in the WAM application.</p> <p>This component calls the W1-MaterialReturnHeader business object using the Inbound Web Service.</p>
W1-PhysicalInventoryCountLine	<p>Used to create, update, read, or delete PhysicalInventoryCountLine in the WAM application.</p> <p>This component calls the W1-PhysicalInventoryCountLine business object using the Inbound Web Service.</p>

Component	Description
W1-PhysicalInventoryCountMaster	<p>Used to create, update, read, or delete PhysicalInventoryCountMaster in the WAM application.</p> <p>This component calls the W1-PhysicalInventoryCountMaster business object using the Inbound Web Service.</p>
W1-PhysicalInventoryHeader	<p>Used to create, update, read, or delete PhysicalInventoryHeader in the WAM application.</p> <p>This component calls the W1-PhysicalInventoryHeader business object using the Inbound Web Service.</p>
W1-ReceiptHeader	<p>Used to create, update, read, or delete ReceiptHeader in the WAM application.</p> <p>This component calls the W1-ReceiptHeader business object using the Inbound Web Service.</p>
W1-ReceiptLine	<p>Used to create, update, read, or delete ReceiptLine in the WAM application.</p> <p>This component calls the W1-ReceiptLine business object using the Inbound Web Service.</p>
W1-ReceiptLineCraft	<p>Used to create, update, read, or delete ReceiptLineCraft in the WAM application.</p> <p>This component calls the W1-ReceiptLineCraft business object using the Inbound Web Service.</p>
W1-ReorderReview	<p>Used to create, update, read, or delete ReorderReview in the WAM application.</p> <p>This component calls the W1-ReorderReview business object using the Inbound Web Service.</p>
W1-ReturnLine	<p>Used to create, update, read, or delete ReturnLine in the WAM application.</p> <p>This component calls the W1-ReturnLine business object using the Inbound Web Service.</p>
W1-SIDDirectPurchase	<p>Used to create, update, read, or delete SIDDirectPurchase in the WAM application.</p> <p>This component calls the W1-SIDDirectPurchase business object using the Inbound Web Service.</p>
W1-SIDSpecifiDirectPurchase	<p>Used to create, update, read, or delete SIDSpecifiDirectPurchase in the WAM application.</p> <p>This component calls the W1-SIDSpecifiDirectPurchase business object using the Inbound Web Service.</p>

Component	Description
W1-SIDExpensedAtReceipt	<p>Used to create, update, read, or delete SIDExpensedAtReceipt in the WAM application.</p> <p>This component calls the W1-SIDExpensedAtReceipt business object using the Inbound Web Service.</p>
W1-SIDInventory	<p>Used to create, update, read, or delete SIDInventory in the WAM application.</p> <p>This component calls the W1-SIDInventory business object using the Inbound Web Service.</p>
W1-SIDSpecificLot	<p>Used to create, update, read, or delete SIDSpecificLot in the WAM application.</p> <p>This component calls the W1-SIDSpecificLot business object using the Inbound Web Service.</p>
W1-StockItem	<p>Used to create, update, read, or delete StockItem in the WAM application.</p> <p>This component calls the W1-StockItem business object using the Inbound Web Service.</p>
W1-StockTransferHeader	<p>Used to create, update, read, or delete StockTransferHeader in the WAM application.</p> <p>This component calls the W1-StockTransferHeader business object using the Inbound Web Service.</p>
W1-StockTransferIssueLine	<p>Used to create, update, read, or delete StockTransferIssueLine in the WAM application.</p> <p>This component calls the W1-StockTransferIssueLine business object using the Inbound Web Service.</p>
W1-StockTransferReceiptHeader	<p>Used to create, update, read, or delete StockTransferReceiptHeader in the WAM application.</p> <p>This component calls the W1-StockTransferReceiptHeader business object using the Inbound Web Service.</p>
W1-StockTransferReceiptLine	<p>Used to create, update, read, or delete StockTransferReceiptLine in the WAM application.</p> <p>This component calls the W1-StockTransferReceiptLine business object using the Inbound Web Service.</p>

Component	Description
W1-StockTransferRequestLine	Used to create, update, read, or delete StockTransferRequestLine in the WAM application. This component calls the W1-StockTransferRequestLine business object using the Inbound Web Service.

Purchasing

Component	Description
W1-BCAccessList	Used to create, update, read, or delete BCAccessList in the WAM application. This component calls the W1-BCAccessList business object using the Inbound Web Service.
W1-BCCraftLine	Used to create, update, read, or delete BCCraftLine in the WAM application. This component calls the W1-BCCraftLine business object using the Inbound Web Service.
W1-BCEquipmentLine	Used to create, update, read, or delete BCEquipmentLine in the WAM application. This component calls the W1-BCEquipmentLine business object using the Inbound Web Service.
W1-BCHeader	Used to create, update, read, or delete BCHeader in the WAM application. This component calls the W1-BCHeader business object using the Inbound Web Service.
W1-BCHeaderIntegration	Used to create, update, read, or delete BCHeaderIntegration in the WAM application. This component calls the W1-BCHeaderIntegration business object using the Inbound Web Service.
W1-BCOtherLine	Used to create, update, read, or delete BCOtherLine in the WAM application. This component calls the W1-BCOtherLine business object using the Inbound Web Service.
W1-BCStockItemLine	Used to create, update, read, or delete BCStockItemLine in the WAM application. This component calls the W1-BCStockItemLine business object using the Inbound Web Service.

Component	Description
W1-CraftPOLine	<p>Used to create, update, read, or delete CraftPOLine in the WAM application.</p> <p>This component calls the W1-CraftPOLine business object using the Inbound Web Service.</p>
W1-EquipmentPOLine	<p>Used to create, update, read, or delete EquipmentPOLine in the WAM application.</p> <p>This component calls the W1-EquipmentPOLine business object using the Inbound Web Service.</p>
W1-EquipmentPRLine	<p>Used to create, update, read, or delete EquipmentPRLine in the WAM application.</p> <p>This component calls the W1-EquipmentPRLine business object using the Inbound Web Service.</p>
W1-Expedite	<p>Used to create, update, read, or delete Expedite in the WAM application.</p> <p>This component calls the W1-Expedite business object using the Inbound Web Service.</p>
W1-OtherResourcePOLine	<p>Used to create, update, read, or delete OtherResourcePOLine in the WAM application.</p> <p>This component calls the W1-OtherResourcePOLine business object using the Inbound Web Service.</p>
W1-POLineIntegration	<p>Used to create, update, read, or delete POLineIntegration in the WAM application.</p> <p>This component calls the W1-POLineIntegration business object using the Inbound Web Service.</p>
W1-POHeaderIntegration	<p>Used to create, update, read, or delete POHeaderIntegration in the WAM application.</p> <p>This component calls the W1-POHeaderIntegration business object using the Inbound Web Service.</p>
W1-PRCreBC	<p>Used to create, update, read, or delete PRCreBC in the WAM application.</p> <p>This component calls the W1-PRCreBC business object using the Inbound Web Service.</p>
W1-PRCrLine	<p>Used to create, update, read, or delete PRCrLine in the WAM application.</p> <p>This component calls the W1-PRCrLine business object using the Inbound Web Service.</p>

Component	Description
W1-PREqLine	<p>Used to create, update, read, or delete PREqLine in the WAM application.</p> <p>This component calls the W1- PREqLine business object using the Inbound Web Service.</p>
W1-PRHeader	<p>Used to create, update, read, or delete PRHeader in the WAM application.</p> <p>This component calls the W1-PRHeader business object using the Inbound Web Service.</p>
W1-PRORLine	<p>Used to create, update, read, or delete PRORLine in the WAM application.</p> <p>This component calls the W1-PRORLine business object using the Inbound Web Service.</p>
W1-PRSILine	<p>Used to create, update, read, or delete PRSILine in the WAM application.</p> <p>This component calls the W1-PRSILine business object using the Inbound Web Service.</p>
W1-PurchaseOrderHeader	<p>Used to create, update, read, or delete PurchaseOrderHeader in the WAM application.</p> <p>This component calls the W1-PurchaseOrderHeader business object using the Inbound Web Service.</p>
W1-StockItemPOLine	<p>Used to create, update, read, or delete StockItemPOLine in the WAM application.</p> <p>This component calls the W1-StockItemPOLine business object using the Inbound Web Service.</p>
W1-Vendor	<p>Used to create, update, read, or delete Vendor in the WAM application.</p> <p>This component calls the W1-Vendor business object using the Inbound Web Service.</p>
W1-VendorLocation	<p>Used to create, update, read, or delete VendorLocation in the WAM application.</p> <p>This component calls the W1-VendorLocation business object using the Inbound Web Service.</p>

Work Management

Component	Description
W1-ActivityComplInboundComm	<p>Used to create, update, read, or delete ActivityComplInboundComm in the WAM application.</p> <p>This component calls the W1-ActivityComplInboundComm business object using the Inbound Web Service.</p>
W1-ActivityEquipRequirement	<p>Used to create, update, read, or delete ActivityEquipRequirement in the WAM application.</p> <p>This component calls the W1-ActivityEquipRequirement business object using the Inbound Web Service.</p>
W1-ActivityLaborRequirement	<p>Used to create, update, read, or delete ActivityLaborRequirement in the WAM application.</p> <p>This component calls the W1-ActivityLaborRequirement business object using the Inbound Web Service.</p>
W1-ActivityMaterialRequirement	<p>Used to create, update, read, or delete ActivityMaterialRequirement in the WAM application.</p> <p>This component calls the W1-ActivityMaterialRequirement business object using the Inbound Web Service.</p>
W1-ActivityOtherRequirement	<p>Used to create, update, read, or delete ActivityOtherRequirement in the WAM application.</p> <p>This component calls the W1-ActivityOtherRequirement business object using the Inbound Web Service.</p>
W1-AdHocActivityCore	<p>Used to create, update, read, or delete AdHocActivityCore in the WAM application.</p> <p>This component calls the W1-AdHocActivityCore business object using the Inbound Web Service.</p>
W1-AssetLocationSpecificTempl	<p>Used to create, update, read, or delete AssetLocationSpecificTempl in the WAM application.</p> <p>This component calls the W1-AssetLocationSpecificTempl business object using the Inbound Web Service.</p>

Component	Description
W1-AssetLocSpecTemplWorkOrder	<p>Used to create, update, read, or delete AssetLocSpecTemplWorkOrder in the WAM application.</p> <p>This component calls the W1-AssetLocSpecTemplWorkOrder business object using the Inbound Web Service.</p>
W1-AssetLocSpecTemplateAct	<p>Used to create, update, read, or delete AssetLocSpecTemplateAct in the WAM application.</p> <p>This component calls the W1-AssetLocSpecTemplateAct business object using the Inbound Web Service.</p>
W1-AssetReplicationActivity	<p>Used to create, update, read, or delete AssetReplicationActivity in the WAM application.</p> <p>This component calls the W1-AssetReplicationActivity business object using the Inbound Web Service.</p>
W1-AssetReplicationComplEvent	<p>Used to create, update, read, or delete AssetReplicationComplEvent in the WAM application.</p> <p>This component calls the W1-AssetReplicationComplEvent business object using the Inbound Web Service.</p>
W1-CreAssetAttachCmpEvt	<p>Used to create, update, read, or delete CreAssetAttachCmpEvt in the WAM application.</p> <p>This component calls the W1-CreAssetAttachCmpEvt business object using the Inbound Web Service.</p>
W1-CreAssetDocumentCmpEvt	<p>Used to create, update, read, or delete CreAssetDocumentCmpEvt in the WAM application.</p> <p>This component calls the W1-CreAssetDocumentCmpEvt business object using the Inbound Web Service.</p>
W1-CreAssetPayCmpEvt	<p>Used to create, update, read, or delete CreAssetPayCmpEvt in the WAM application.</p> <p>This component calls the W1-CreAssetPayCmpEvt business object using the Inbound Web Service.</p>

Component	Description
W1-CreNodeAttachCmpEvt	<p>Used to create, update, read, or delete CreNodeAttachCmpEvt in the WAM application.</p> <p>This component calls the W1-CreNodeAttachCmpEvt business object using the Inbound Web Service.</p>
W1-CreServiceHistoryAttCmpEvt	<p>Used to create, update, read, or delete CreServiceHistoryAttCmpEvt in the WAM application.</p> <p>This component calls the W1-CreServiceHistoryAttCmpEvt business object using the Inbound Web Service.</p>
W1-SubscriptionShiftPlanTemp	<p>Used to create, update, read, or delete SubscriptionShiftPlanTemp in the WAM application.</p> <p>This component calls the W1-SubscriptionShiftPlanTemp business object using the Inbound Web Service.</p>
W1-CommonShiftPlanTemplate	<p>Used to create, update, read, or delete CommonShiftPlanTemplate in the WAM application.</p> <p>This component calls the W1-CommonShiftPlanTemplate business object using the Inbound Web Service.</p>
W1-Craft	<p>Used to create, update, read, or delete Craft in the WAM application.</p> <p>This component calls the W1-Craft business object using the Inbound Web Service.</p>
W1-CreActivityPayCmpEvt	<p>Used to create, update, read, or delete CreActivityPayCmpEvt in the WAM application.</p> <p>This component calls the W1-CreActivityPayCmpEvt business object using the Inbound Web Service.</p>
W1-CreateActivityInboundComm	<p>Used to create, update, read, or delete CreateActivityInboundComm in the WAM application.</p> <p>This component calls the W1-CreateActivityInboundComm business object using the Inbound Web Service.</p>

Component	Description
W1-CreComActivityInboundComm	<p>Used to create, update, read, or delete CreComActivityInboundComm in the WAM application.</p> <p>This component calls the W1-CreComActivityInboundComm business object using the Inbound Web Service.</p>
W1-CreServiceHistoryPayCmpEvt	<p>Used to create, update, read, or delete CreServiceHistoryPayCmpEvt in the WAM application.</p> <p>This component calls the W1-CreServiceHistoryPayCmpEvt business object using the Inbound Web Service.</p>
W1-Crew	<p>Used to create, update, read, or delete Crew in the WAM application.</p> <p>This component calls the W1-Crew business object using the Inbound Web Service.</p>
W1-CreWorkReqDocumentCmpEvt	<p>Used to create, update, read, or delete CreWorkReqDocumentCmpEvt in the WAM application.</p> <p>This component calls the W1-CreWorkReqDocumentCmpEvt business object using the Inbound Web Service.</p>
W1-CreWorkRequestPayCmpEvt	<p>Used to create, update, read, or delete CreWorkRequestPayCmpEvt in the WAM application.</p> <p>This component calls the W1-CreWorkRequestPayCmpEvt business object using the Inbound Web Service.</p>
W1-CrewShiftTemplate	<p>Used to create, update, read, or delete CrewShiftTemplate in the WAM application.</p> <p>This component calls the W1-CrewShiftTemplate business object using the Inbound Web Service.</p>
W1-CrewWithCraft	<p>Used to create, update, read, or delete CrewWithCraft in the WAM application.</p> <p>This component calls the W1-CrewWithCraft business object using the Inbound Web Service.</p>
W1-CreSvcHistCmpEvt	<p>Used to create, update, read, or delete CreSvcHistCmpEvt in the WAM application.</p> <p>This component calls the W1-CreSvcHistCmpEvt business object using the Inbound Web Service.</p>

Component	Description
W1-CreAssetCmpEvt	<p>Used to create, update, read, or delete CreAssetCmpEvt in the WAM application.</p> <p>This component calls the W1-CreAssetCmpEvt business object using the Inbound Web Service.</p>
W1-CreActivityCmpEvt	<p>Used to create, update, read, or delete CreActivityCmpEvt in the WAM application.</p> <p>This component calls the W1-CreActivityCmpEvt business object using the Inbound Web Service.</p>
W1- CreAssetDocumentCmpEvt	<p>Used to create, update, read, or delete CreAssetDocumentCmpEvt in the WAM application.</p> <p>This component calls the W1-CreAssetDocumentCmpEvt business object using the Inbound Web Service.</p>
W1- MeasQuckEntryActivityType	<p>Used to create, update, read, or delete MeasQuckEntryActivityType in the WAM application.</p> <p>This component calls the W1-MeasQuckEntryActivityType business object using the Inbound Web Service.</p>
W1-RCMCalendarInterval	<p>Used to create, update, read, or delete RCMCalendarInterval in the WAM application.</p> <p>This component calls the W1- RCMCalendarInterval business object using the Inbound Web Service.</p>
W1-RCMRuntimeAnniversary	<p>Used to create, update, read, or delete RCMRuntimeAnniversary in the WAM application.</p> <p>This component calls the W1-RCMRuntimeAnniversary business object using the Inbound Web Service.</p>
W1-RCMRuntimeInterval	<p>Used to create, update, read, or delete RCMRuntimeInterval in the WAM application.</p> <p>This component calls the W1- RCMRuntimeInterval business object using the Inbound Web Service.</p>
W1-DailySchedule	<p>Used to create, update, read, or delete DailySchedule in the WAM application.</p> <p>This component calls the W1-DailySchedule business object using the Inbound Web Service.</p>

Component	Description
W1-Employee	Used to create, update, read, or delete Employee in the WAM application. This component calls the W1-Employee business object using the Inbound Web Service.
W1-EquipForTemplateAct	Used to create, update, read, or delete EquipForTemplateAct in the WAM application. This component calls the W1-EquipForTemplateAct business object using the Inbound Web Service.
W1-GeneralTemplateActivity	Used to create, update, read, or delete GeneralTemplateActivity in the WAM application. This component calls the W1-GeneralTemplateActivity business object using the Inbound Web Service.
W1-GeneralTemplateWorkOrder	Used to create, update, read, or delete GeneralTemplateWorkOrder in the WAM application. This component calls the W1-GeneralTemplateWorkOrder business object using the Inbound Web Service.
W1-LaborForTemplateAct	Used to create, update, read, or delete LaborForTemplateAct in the WAM application. This component calls the W1-LaborForTemplateAct business object using the Inbound Web Service.
W1-Leave	Used to create, update, read, or delete Leave in the WAM application. This component calls the W1-Leave business object using the Inbound Web Service.
W1-LocalActivityCoreWAM	Used to create, update, read, or delete LocalActivityCoreWAM in the WAM application. This component calls the W1-LocalActivityCoreWAM business object using the Inbound Web Service.
W1-MaintenanceSchedule	Used to create, update, read, or delete MaintenanceSchedule in the WAM application. This component calls the W1-MaintenanceSchedule business object using the Inbound Web Service.
W1-ODCEquipment	Used to create, update, read, or delete ODCEquipment in the WAM application. This component calls the W1-ODCEquipment business object using the Inbound Web Service.

Component	Description
W1-OperationalReadingCmpEvt	<p>Used to create, update, read, or delete OperationalReadingCmpEvt in the WAM application.</p> <p>This component calls the W1-OperationalReadingCmpEvt business object using the Inbound Web Service.</p>
W1-Other	<p>Used to create, update, read, or delete Other in the WAM application.</p> <p>This component calls the W1-Other business object using the Inbound Web Service.</p>
W1-OtherDirectCharges	<p>Used to create, update, read, or delete OtherDirectCharges in the WAM application.</p> <p>This component calls the W1-OtherDirectCharges business object using the Inbound Web Service.</p>
W1-OtherDirectChargesOther	<p>Used to create, update, read, or delete OtherDirectChargesOther in the WAM application.</p> <p>This component calls the W1-OtherDirectChargesOther business object using the Inbound Web Service.</p>
W1-OtherForTemplateAct	<p>Used to create, update, read, or delete OtherForTemplateAct in the WAM application.</p> <p>This component calls the W1-OtherForTemplateAct business object using the Inbound Web Service.</p>
W1-Project	<p>Used to create, update, read, or delete Project in the WAM application.</p> <p>This component calls the W1-Project business object using the Inbound Web Service.</p>
W1-RCMCalendarAnniversary	<p>Used to create, update, read, or delete RCMCalendarAnniversary in the WAM application.</p> <p>This component calls the W1-RCMCalendarAnniversary business object using the Inbound Web Service.</p>
W1-RetrieveComplEvents	<p>Used to retrieve the completion events from an activity.</p> <p>This component calls the W1-RETCEVT business service using the Inbound Web Service.</p>
W1-SchedulePlan	<p>Used to create, update, read, or delete SchedulePlan in the WAM application.</p> <p>This component calls the W1-SchedulePlan business object using the Inbound Web Service.</p>

Component	Description
W1-TemplateActivityLaborReq	<p>Used to create, update, read, or delete TemplateActivityLaborReq in the WAM application.</p> <p>This component calls the W1-TemplateActivityLaborReq business object using the Inbound Web Service.</p>
W1-TempActMaterialReq	<p>Used to create, update, read, or delete TemplateActMaterialReq in the WAM application.</p> <p>This component calls the W1-TempActMaterialReq business object using the Inbound Web Service.</p>
W1-Timesheet	<p>Used to create, update, read, or delete Timesheet in the WAM application.</p> <p>This component calls the W1-Timesheet business object using the Inbound Web Service.</p>
W1-TimesheetDetail	<p>Used to create, update, read, or delete TimesheetDetail in the WAM application.</p> <p>This component calls the W1-TimesheetDetail business object using the Inbound Web Service.</p>
W1-TrnAssetCmpEvent	<p>Used to create, update, read, or delete TrnAssetCmpEvent in the WAM application.</p> <p>This component calls the W1-TrnAssetCmpEvent business object using the Inbound Web Service.</p>
W1- UpdAssetPayCmpEvt	<p>Used to create, update, read, or delete UpdAssetPayCmpEvt in the WAM application.</p> <p>This component calls the W1- UpdAssetPayCmpEvt business object using the Inbound Web Service.</p>
W1- UpdNodePayCmpEvt	<p>Used to create, update, read, or delete UpdNodePayCmpEvt in the WAM application.</p> <p>This component calls the W1-UpdNodePayCmpEvt business object using the Inbound Web Service.</p>
W1-WorkOrder	<p>Used to create, update, read, or delete WorkOrder in the WAM application.</p> <p>This component calls the W1-WorkOrder business object using the Inbound Web Service.</p>
W1-WorkRequest	<p>Used to create, update, read, or delete WorkRequest in the WAM application.</p> <p>This component calls the W1-WorkRequest business object using the Inbound Web Service.</p>

Component	Description
W1-WorkRequestCompEvt	Used to create, update, read, or delete WorkRequestCompEvt in the WAM application. This component calls the W1-WorkRequestCompEvt business object using the Inbound Web Service.

Invoicing

Component	Description
W1-InvoiceHeader	Used to create, update, read, or delete InvoiceHeader in the WAM application. This component calls the W1-InvoiceHeader business object using the Inbound Web Service.
W1-InvoiceLine	Used to create, update, read, or delete InvoiceLine in the WAM application. This component calls the W1-InvoiceLine business object using the Inbound Web Service.

Integration

Component	Description
W1- ReadAssetLocationId	Used to read an Asset Location ID (asset node) using an External ID. This component calls the CM-ASTLOCID business service using the Inbound Web Service (ATW1READASSETLOCID).
W1-ReadServiceHistoryId	Used to read an Service History ID (install event from Oracle Utilities Meter Data Management) using an External ID. This component calls the CM-READSRVHSTID business service using the Inbound Web Service (ATW1READSRVHSTID).
W1- ReadContact	Used to read a Contact using an External ID. This component calls the CM-READCNCT business service using the Inbound Web Service (ATW1READCONTACT).

Oracle Utilities Operational Device Management Components

This section lists the starter components available in Oracle Utilities Operational Device Management.

Pre-requisites: The Inbound Web Service using the respective business object should be available in the application.

Additional Notes: Failure while creating, reading, or updating the component is logged to the test execution report, facilitating debugging/analysis of the problem.

The components are categorized under the following functional areas:

- [Administration Data](#)
- [Asset Management](#)
- [Configuration Management](#)
- [Work Management](#)

Administration Data

Component	Description
W1- AssetAgeBktConfig	Used to create, update, read, or delete AssetAgeBktConfig in the WAM application. This component calls the W1- AssetAgeBktConfig business object using the Inbound Web Service.
W1- BasicActivityType	Used to create, update, read, or delete BasicActivityType in the WAM application. This component calls the W1- BasicActivityType business object using the Inbound Web Service.
W1- BIExtractParameters	Used to create, update, read, or delete BIExtractParameters in the WAM application. This component calls the W1- BIExtractParameters business object using the Inbound Web Service.
W1- ConfigType	Used to create, update, read, or delete ConfigType in the WAM application. This component calls the W1- ConfigType business object using the Inbound Web Service.
W1- OutboundActivityType	Used to create, update, read, or delete OutboundActivityType in the WAM application. This component calls the W1- OutboundActivityType business object using the Inbound Web Service.

Component	Description
W1- GeneralCommunicationType	<p>Used to create, update, read, or delete GeneralCommunicationType in the WAM application.</p> <p>This component calls the W1-GeneralCommunicationType business object using the Inbound Web Service.</p>
W1- AssetInstallAgeBktConfig	<p>Used to create, update, read, or delete AssetInstallAgeBktConfig in the WAM application.</p> <p>This component calls the W1-AssetInstallAgeBktConfig business object using the Inbound Web Service.</p>
W1- AssetInstoreAgeBktConfig	<p>Used to create, update, read, or delete AssetInstoreAgeBktConfig in the WAM application.</p> <p>This component calls the W1-AssetInstoreAgeBktConfig business object using the Inbound Web Service.</p>
W1- AssetSearchMasterConfig	<p>Used to create, update, read, or delete AssetSearchMasterConfig in the WAM application.</p> <p>This component calls the W1-AssetSearchMasterConfig business object using the Inbound Web Service.</p>

Asset Management

Component	Description
W1- Business	Used to create, update, read, or delete Business in the WAM application. This component calls the W1- Business business object using the Inbound Web Service.
W1- MDMAAsset	Used to create, update, read, or delete MDMAAsset in the WAM application. This component calls the W1- MDMAAsset business object using the Inbound Web Service.
W1- Person	Used to create, update, read, or delete Person in the WAM application. This component calls the W1- Person business object using the Inbound Web Service.
W1- SynchronizationAddContact	Used to create, update, read, or delete SynchronizationAddContact in the WAM application. This component calls the W1- SynchronizationAddContact business object using the Inbound Web Service.
W1- BasicTest	Used to create, update, read, or delete BasicTest in the WAM application. This component calls the W1- BasicTest business object using the Inbound Web Service.
W2-SmartTest	Used to create, update, read, or delete SmartTest in the WAM application. This component calls the W2- SmartTest business object using the Inbound Web Service.

Configuration Management

Component	Description
W1- AssetCompCfgRpt	Used to create, update, read, or delete AssetCompCfgRpt in the WAM application. This component calls the W1- AssetCompCfgRpt business object using the Inbound Web Service.
W1- AssetCompCfgWO	Used to create, update, read, or delete AssetCompCfgWO in the WAM application. This component calls the W1- AssetCompCfgWO business object using the Inbound Web Service.

Component	Description
W1- AssetCompToAssetCfgRpt	<p>Used to create, update, read, or delete AssetCompToAssetCfgRpt in the WAM application.</p> <p>This component calls the W1- AssetCompToAssetCfgRpt business object using the Inbound Web Service.</p>
W1- AssetCompToAssetCfgWO	<p>Used to create, update, read, or delete AssetCompToAssetCfgWO in the WAM application.</p> <p>This component calls the W1- AssetCompToAssetCfgWO business object using the Inbound Web Service.</p>
W1- AssetCompToAssetCompCfgRpt	<p>Used to create, update, read, or delete AssetCompToAssetCompCfgRpt in the WAM application.</p> <p>This component calls the W1- AssetCompToAssetCompCfgRpt business object using the Inbound Web Service.</p>
W1- AssetCompToAssetCompCfgWO	<p>Used to create, update, read, or delete AssetCompToAssetCompCfgWO in the WAM application.</p> <p>This component calls the W1- AssetCompToAssetCompCfgWO business object using the Inbound Web Service.</p>
W1- AssetCompToAssetCompConfig	<p>Used to create, update, read, or delete AssetCompToAssetCompConfig in the WAM application.</p> <p>This component calls the W1- AssetCompToAssetCompConfig business object using the Inbound Web Service.</p>
W1- AssetCompToAssetConfig	<p>Used to create, update, read, or delete AssetCompToAssetConfig in the WAM application.</p> <p>This component calls the W1- AssetCompToAssetConfig business object using the Inbound Web Service.</p>
W1- AssetToAssetCfgRpt	<p>Used to create, update, read, or delete AssetToAssetCfgRpt in the WAM application.</p> <p>This component calls the W1- AssetToAssetCfgRpt business object using the Inbound Web Service.</p>
W1- AssetToAssetCfgWO	<p>Used to create, update, read, or delete AssetToAssetCfgWO in the WAM application.</p> <p>This component calls the W1- AssetToAssetCfgWO business object using the Inbound Web Service.</p>

Component	Description
W1- AssetToAssetCompCfgRpt	<p>Used to create, update, read, or delete AssetToAssetCompCfgRpt in the WAM application.</p> <p>This component calls the W1- AssetToAssetCompCfgRpt business object using the Inbound Web Service.</p>
W1- AssetToAssetCompCfgWO	<p>Used to create, update, read, or delete AssetToAssetCompCfgWO in the WAM application.</p> <p>This component calls the W1- AssetToAssetCompCfgWO business object using the Inbound Web Service.</p>
W1- AssetToAssetCompConfig	<p>Used to create, update, read, or delete AssetToAssetCompConfig in the WAM application.</p> <p>This component calls the W1- AssetToAssetCompConfig business object using the Inbound Web Service.</p>
W1- AssetToAssetConfig	<p>Used to create, update, read, or delete AssetToAssetConfig in the WAM application.</p> <p>This component calls the W1- AssetToAssetConfig business object using the Inbound Web Service.</p>
W1- SingleAssetCfgRpt	<p>Used to create, update, read, or delete SingleAssetCfgRpt in the WAM application.</p> <p>This component calls the W1- SingleAssetCfgRpt business object using the Inbound Web Service.</p>
W1- SingleAssetCfgWO	<p>Used to create, update, read, or delete SingleAssetCfgWO in the WAM application.</p> <p>This component calls the W1- SingleAssetCfgWO business object using the Inbound Web Service.</p>
W1- SingleAssetConfig	<p>Used to create, update, read, or delete SingleAssetConfig in the WAM application.</p> <p>This component calls the W1- SingleAssetConfig business object using the Inbound Web Service.</p>
W1- AssetCompConfig	<p>Used to create, update, read, or delete AssetCompConfig in the WAM application.</p> <p>This component calls the W1- AssetCompConfig business object using the Inbound Web Service.</p>

Work Management

Component	Description
W1- ActivityComplInboundComm	<p>Used to create, update, read, or delete ActivityComplInboundComm in the WAM application.</p> <p>This component calls the W1- ActivityComplInboundComm business object using the Inbound Web Service.</p>
W1- AdHocActivityCore	<p>Used to create, update, read, or delete AdHocActivityCore in the WAM application.</p> <p>This component calls the W1- AdHocActivityCore business object using the Inbound Web Service.</p>
W1- AssetCriteriaActGenerator	<p>Used to create, update, read, or delete AssetCriteriaActGenerator in the WAM application.</p> <p>This component calls the W1- AssetCriteriaActGenerator business object using the Inbound Web Service.</p>
W1- AssetListActGenerator	<p>Used to create, update, read, or delete AssetListActGenerator in the WAM application.</p> <p>This component calls the W1- AssetListActGenerator business object using the Inbound Web Service.</p>
W1- BasicWorkOrder	<p>Used to create, update, read, or delete BasicWorkOrder in the WAM application.</p> <p>This component calls the W1- BasicWorkOrder business object using the Inbound Web Service.</p>
W1- CompnAssetListActGenerator	<p>Used to create, update, read, or delete CompnAssetListActGenerator in the WAM application.</p> <p>This component calls the W1- CompnAssetListActGenerator business object using the Inbound Web Service.</p>
W1- CompnCriteriaActGenerator	<p>Used to create, update, read, or delete CompnCriteriaActGenerator in the WAM application.</p> <p>This component calls the W1- CompnCriteriaActGenerator business object using the Inbound Web Service.</p>

Component	Description
W1- CreComActivityInboundComm	Used to create, update, read, or delete CreComActivityInboundComm in the WAM application. This component calls the W1- CreComActivityInboundComm business object using the Inbound Web Service.
W1- CreateActivityInboundComm	Used to create, update, read, or delete CreateActivityInboundComm in the WAM application. This component calls the W1- CreateActivityInboundComm business object using the Inbound Web Service.
W1- LocalActivityCore	Used to create, update, read, or delete LocalActivityCore in the WAM application. This component calls the W1- LocalActivityCore business object using the Inbound Web Service.
W1- NodeCriteriaActGenerator	Used to create, update, read, or delete NodeCriteriaActGenerator in the WAM application. This component calls the W1- NodeCriteriaActGenerator business object using the Inbound Web Service.
W1- OutboundActivityCore	Used to create, update, read, or delete OutboundActivityCore in the WAM application. This component calls the W1- OutboundActivityCore business object using the Inbound Web Service.

Components for BPA Scripts

The following table lists the components for BPA scripts.

Additional Notes: Failure while using the component is logged to the test execution report specifying the reason for the failure to debug/analyze and raise bugs accordingly.

Component	Functional Area	Description
W1-AddActivityToCrewShift	Work Management	Allows to add the Crew Shift to Activity by launching the UI.
W1-MaterialReqLineIssueAll	Inventory Management	Allows to click Issue All for a particular Open Material request line in the Material Request Header page.

Component	Functional Area	Description
W1-MaterialReqLineIssueLine	Inventory Management	Allows to click Issue All for a particular Open Material request line in the Material Request Header page.
W1-StkTranOpenReqLineIssueAll	Inventory Management	Allows to click IssueAll of the Stock Transfer Open Request line.
W1-StkTranReceiptLineReceive	Inventory Management	Allows to accept all the stock transferred to the requesting storeroom.
W1-ReceiptLineReceiveAll	Inventory Management	Allows to receive all the stock from a receipt line on the Receipt Header.
W1-ReceivePartialEditQuantity	Inventory Management	Allows to receive partial quantity after setting the quantity on the Receipt line.
W1-InReceiptLineAcceptEditQty	Inventory Management	Allows to accept the stock from a single receiving line after setting the quantity.
W1-InReceiptLineReturnEditQty	Inventory Management	Allows to return the stock of a single receipt line on the Receipt Header.
W1-MaterialDispCmpleteExchnage	Work Management	Allows to complete the exchange on the Material Disposition page.
W1-PhyInvAcceptOnHand	Inventory Management	Allows to click Accept On Hand in the First Count zone in the Physical Inventory Header page.
W1-PhyInvSetFrstCntAndSave	Inventory Management	Allows to set the first count value and click Save in the First Count zone of the Physical Inventory Header page.
W1-AddSingleActivityToCrewShift	Work Management	Allows to add Crew Shift to a single Activity by launching the UI.
W1-BCCreatePO	Purchasing	Allows to create purchase order from a blanket contract record.
W1-BCCreatePR	Purchasing	Allows to create purchase requisition from a blanket contract record.
W1-POForceComplete	Purchasing	Allows to force complete a purchase order.

Components Not Compatible for Cloud Automation

The following table lists the components that are not compatible for cloud automation.

Component	Functional Area	Description
W1-PurchaseOrderHeader	Purchasing	Used to create, update, read, or delete PurchaseOrderHeader in the WAM application. This component calls the W1-PurchaseOrderHeader business object using the Inbound Web Service.
W1-AssetCriteriaActGenerator	Work Management	Used to create, update, read, or delete AssetCriteriaActGenerator in the WAM application. This component calls the W1-AssetCriteriaActGenerator business object using the Inbound Web Service.
W1-AssetListActGenerator	Work Management	Used to create, update, read, or delete AssetListActGenerator in the WAM application. This component calls the W1-AssetListActGenerator business object using the Inbound Web Service.
W1-CompnAssetListActGenerator	Work Management	Used to create, update, read, or delete CompnAssetListActGenerator in the WAM application. This component calls the W1-CompnAssetListActGenerator business object using the Inbound Web Service.
W1-CompnCriteriaActGenerator	Work Management	Used to create, update, read, or delete CompnCriteriaActGenerator in the WAM application. This component calls the W1-CompnCriteriaActGenerator business object using the Inbound Web Service.

Component	Functional Area	Description
W1-NodeCriteriaActGenerator	Work Management	<p>Used to create, update, read, or delete NodeCriteriaActGenerator in the WAM application.</p> <p>This component calls the W1-NodeCriteriaActGenerator business object using the Inbound Web Service.</p>
W1-ReorderReview	Inventory Management	<p>Used to create, update, read, or delete ReorderReview in the WAM application.</p> <p>This component calls the W1-ReorderReview business object using the Inbound Web Service.</p>
W1-NumericGauge	Administration Data	<p>Used to create, update, read, or delete NumericGauge in the WAM application.</p> <p>This component calls the W1-NumericGauge business object using the Inbound Web Service.</p>
W1-NumericMeter	Administration Data	<p>Used to create, update, read, or delete NumericMeter in the WAM application.</p> <p>This component calls the W1-NumericMeter business object using the Inbound Web Service.</p>
W1-ApprovalRequest	General	<p>Used to create, update, read, or delete ApprovalRequest in the WAM application.</p> <p>This component calls the W1-ApprovalRequest business object using the Inbound Web Service.</p>
W1-FinancialTransaction	General	<p>Used to create, update, read, or delete FinancialTransaction in the WAM application.</p> <p>This component calls the W1-FinancialTransaction business object using the Inbound Web Service.</p>

Component	Functional Area	Description
W1-ServiceHistoryWAM	Asset Management	<p>Used to create, update, read, or delete ServiceHistory in the WAM application.</p> <p>This component calls the W1-ServiceHistory (Deficiency/ General Weighted/ Percent of New) business object using the Inbound Web Service.</p>
W1-BOMGenStockItem	Inventory Management	<p>Used to create, update, read, or delete BOMGenStockItem in the WAM application.</p> <p>This component calls the W1-BOMGenStockItem business object using the Inbound Web Service.</p>

Chapter 2

Function Library Reference

This chapter lists the Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management function library [OUWAMUILIB](#) and the functions available to create components and flows in Oracle Flow Builder for testing the Oracle Utilities Work and Asset Management/ Oracle Utilities Operational Device Management application.

OUWAMUILIB

The OUWAMUILIB library comprises functions that help to carry out operations in the Oracle Utilities Work and Asset Management application UI that are generally achieved using BPA scripts.

Note: Private functions in the library are used to implement the public functions mentioned below. They cannot be used on their own in the Oracle Flow Builder scripts.

This section provides a list of functions included in the library, along with their usage details, in the following categories:

- [Common Methods](#)
- [Physical Inventory](#)
- [Material Request/Issue and Return](#)
- [Stock Transfer](#)
- [Blanket Contract](#)
- [Preventive Maintenance](#)
- [Purchase Order](#)
- [Financial Transaction](#)
- [Scheduling](#)
- [Material Disposition](#)
- [Material Return](#)

Common Methods

login

Allows the user to login to the application using the user name and password provided as inputs. On success, it opens the **Home** page of the application. Else, the flow fails.

Example:

```
login(String strUser, String strPasswd)
```

Input Parameters: String strUser, String strPasswd

Return Type: void

selectMenu

Allows the user to navigate to a Menu Item based on the input parameters. On success, it navigates to the required page of the application. Else, the flow fails.

This method is called when the login is successful or the user is in the **Home** page of the application.

Example:

```
selectMenu(String sTopMenu, String sMenu, String sSubMenu, String sSearchorAdd)
```

Input Parameters: String sTopMenu, String sMenu, String sSubMenu, String sSearchorAdd

Return Type: void

selectBox

Allows to select a value from the drop-down based on the input value.

Example:

```
selectBox(String objPath, String value)
```

Input Parameters: String objPath, String value)

Return Type: void

clickTableCell

Allows to click on a table cell based on the content of the table cell.

Example:

```
clickTableCell(String objPath, String strCellData, int intCol, int intColToClick)
```

Input Parameters: String objPath, String strCellData, int intCol, int intColToClick

Return Type: void

clickTableCell

Allows to click on a table cell based on the row and column of the table cell.

Example:

```
clickTableCell(String objPath, int intRow, int intCol)
```

Input Parameters: String objPath, int intRow, int intCol

Return Type: void

clickTableCellCheckBox

Allows the user to click the check box located in a table cell based on the row and column of the table cell.

Example:

```
clickTableCellCheckBox(String objPath, int intRow, int intCol)
```

Input Parameters: String objPath, int intRow, int intCol

Return Type: void

checkBox

Allows to click the check box based on the object path provided as input.

Example:

```
checkBox(String objPath, String value)
```

Input Parameters: String objPath, String value

Return Type: void

Physical Inventory

physicalInventory_searchAndSelectById

Searches for the Physical Inventory Header based on the provided ID. If found, the Physical Inventory Header opens.

On success the Physical Inventory Header record opens. On failure, the Physical Inventory Header is not selected and the flow fails.

Example:

```
physicalInventory_searchAndSelectById(String strPhyInvRecId)
```

Input Parameters: String strPhyInvRecId

Return Type: void

physicalInventory_FirstCnt_AcceptOnHand

Clicks the acceptOnHand button under first count zone in the Physical Inventory Header page based on the provided row ID. If not found, does not click the AcceptOnHand button and the flow fails.

This method should be called only after the physicalInventory_searchAndSelectById method is called.

Example:

```
physicalInventory_FirstCnt_AcceptOnHand(String rowNumber)
```

Input Parameters: String rowNumber

Return Type: void

physicalInventory_FirstCnt_Discard

Clicks the Discard button under first count zone in the Physical Inventory Header page based on the provided row ID. If not found, does not click the Discard button and the flow fails.

This method should be called only after the physicalInventory_searchAndSelectById method is called.

Example:

```
physicalInventory_FirstCnt_Discard(String rowNum)
```

Input Parameters: String rowNum

Return Type: void

physicalInventory_setFirstCount

Sets the first count value under first count zone in the Physical Inventory Header page based on the provided row ID and value. If not found, the first count value will not be set and the flow fails.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_setFirstCount(String rowNum, String value)
```

Input Parameters: String rowNum, String value

Return Type: void

physicalInventory_FirstCount_Save

Clicks the **Save** button under first count zone in the Physical Inventory Header page. If not found, the **Save** button will not be clicked and the flow fails.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_FirstCount_Save()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCount_DiscardAll

Clicks the **DiscardAll** button under first count zone in the Physical Inventory Header page. If not found, the **DiscardAll** button will not be clicked and the flow fails.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_FirstCount_DiscardAll()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_LmtToBin

Sets the Limit To Bin value under first count zone in the Physical Inventory Header page based on value provided. If not found, the Limit To Bin value is not set and the flow fails.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_FirstCnt_LmtToBin(String strBin)
```

Input Parameters: String strBin

Return Type: void

physicalInventory_FirstCnt_LmtToBin_Search

Clicks the **Search** button under first count zone in the Physical Inventory Header page. If not found, the Search button value will not be clicked and the flow fails.

This method should be called only after the physicalInventory_FirstCnt_LmtToBin method is called.

Example:

```
physicalInventory_FirstCnt_LmtToBin_Search()
```

Input Parameters: NA

Return Type: void

physicalInventory_setReCount

Sets the Recount value under the Recount Zone in the Physical Inventory Header page based on the row number and value provided. If not found, the Recount value is not set and the flow fails.

This method should be called only after physicalInventory_FirstCount_Save method is called.

Example:

```
physicalInventory_setReCount(String rowNumber, String value)
```

Input Parameters: String rowNumber, String value

Return Type: void

physicalInventory_Recount_Undo

Clicks the **Undo** button under the Recount Zone in the Physical Inventory Header page based on the row number. If not found, the **Undo** button will not be clicked and the flow fails.

This method should be called only after the physicalInventory_FirstCount_Save method is called.

Example:

```
physicalInventory_Recount_Undo(String rowNumber)
```

Input Parameters: String rowNumber

Return Type: void

physicalInventory_Recount_Discard

Clicks the **Discard** button under the Recount Zone in the Physical Inventory Header page based on the row number. If not found, the **Discard** button will not be clicked and the flow fails.

This method should be called only after the physicalInventory_FirstCount_Save method is called.

Example:

```
physicalInventory_Recount_Discard(String rowNumber)
```

Input Parameters: String rowNumber

Return Type: void

physicalInventory_Recount_Save

Clicks the **Save** button under the Recount Zone in the Physical Inventory Header page. If not found, the **Save** button will not be clicked and the flow fails.

This method should be called only after the physicalInventory_setReCount method is called.

Example:

```
physicalInventory_Recount_Save()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_DiscardAll

Clicks the **DiscardAll** button under the Recount Zone in the Physical Inventory Header page. If not found, the **DiscardAll** button will not be clicked and the flow fails.

This method should be called only after the `physicalInventory_FirstCount_Save` method is called.

Example:

```
physicalInventory_Recount_DiscardAll()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_LmtToBin

Sets the Limit To Bin value under Recount zone in the Physical Inventory Header page based on value provided. If not found, the Limit To Bin value will not be set and the flow fails.

Example:

```
physicalInventory_Recount_LmtToBin(String strBin)
```

Input Parameters: String strBin

Return Type: void

physicalInventory_Recount_LmtToBin_Search

Clicks the **Search** button under Recount zone in the Physical Inventory Header page. If not found, the **Search** button value will not be clicked and the flow fails.

This method should be called only after the `physicalInventory_Recount_LmtToBin` method is called.

Example:

```
physicalInventory_Recount_LmtToBin_Search()
```

Input Parameters: NA

Return Type: void

physicalInventory_CompleteCount_Undo

Clicks the **Undo** button under the CompleteCount Zone in the Physical Inventory Header page based on the occurrence of the **Undo** button. If not found, the **Undo** button will not be clicked and the flow fails.

This method should be called only after the `physicalInventory_Recount_save` or `physicalInventory_FirstCount_AcceptOnHand` methods are called.

Example:

```
physicalInventory_CompleteCount_Undo(String occurrenceOfUndoButton)
```

Input Parameters: String occurrenceOfUndoButton

Return Type: void

physicalInventory_CompleteCount_LmtToBin

Sets the Limit To Bin value under Complete Count zone in the Physical Inventory Header page based on value provided. If not found, the Limit To Bin value will not be set and the flow fails.

This method should be called only after the `physicalInventory_Recount_save` or `physicalInventory_FirstCount_AcceptOnHand` methods are called.

Example:

```
physicalInventory_CompleteCount_LmtToBin(String strBin)
```

Input Parameters: String strBin

Return Type: void

physicalInventory_CompleteCount_LmtToBin_Search

Clicks the **Search** button under CompleteCount zone in the Physical Inventory Header page. If not found, the **Search** button will not be clicked and the flow fails.

This method should be called only after the physicalInventory_CompleteCount_LmtToBin method is called.

Example:

```
physicalInventory_CompleteCount_LmtToBin_Search()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_AuditSerialNumber

Allows the user to click **Audit Serial Number** in the provided row in the **Fist Count** zone of the **Physical Inventory Header** page.

This method should be called only after the physicalInventory_searchAndSelectById method is called.

Example:

```
physicalInventory_FirstCnt_AuditSerialNumber(String rowNumber)
```

Input Parameters: String rowNumber

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToSerialNumber

Allows the user to set the value for **Limit To Serial Number** field in **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the physicalInventory_FirstCnt_AuditSerialNumber method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToSerialNumber  
(String strSerialNum)
```

Input Parameters: String strSerialNum

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToBadgeNumber

Allows the user to set the value for **Limit To Badge Number** field in **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the physicalInventory_FirstCnt_AuditSerialNumber method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToBadgeNumber (String  
strBadgeNum)
```

Input Parameters: String strBadgeNum

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToPalletNumber

Allows the user to set the value for **Limit To pallet Number** field in **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_FirstCnt_AuditSerialNumber` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToPalletNumber
(String strPalletNum)
```

Input Parameters: String strPalletum

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_clickSearch

Allows the user to click **Search** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_AuditSerialNumber` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_clickSearch()
```

Input Parameters:NA

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_CheckAllAssets

Allows the user to select the check box to select all the assets in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_clickSearch` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_CheckAllAssets()
```

Input Parameters:NA

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_CheckSingleAsset

Allows the user to select the check box to select a single asset in the **Tracked Assets Audit** page based on the row number provided as input.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_clickSearch` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_CheckSingleAsset(String
strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound

Allows user to click **Found** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_CheckSingleAsset` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToSerialNumberFound

Allows the user to set the value for **Limit To Serial Number** field in the **Found Assets** zone of in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToSerialNumberFound(
String strSerialNum)
```

Input Parameters: String strSerialNum

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToBadgeNumberFound

Allows the user to set the value for **Limit To Badge Number** field in the **Found Assets** zone of the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToBadgeNumberFound(
String strBadgeNum)
```

Input Parameters: String strBadgeNum

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToPalletNumberFound

Allows the user to set the value for **Limit To Pallet Number** field in the **Found Assets** zone in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_setLmtToPalletNumberFound(
String strPalletNum)
```

Input Parameters: String strPalletum

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_FoundAssets_clickSearch

Allows to click **Search** in the **Found Assets** zone in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_AuditSerialNumber` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_FoundAssets_clickSearch()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_FoundAssets_RemoveSingleAsset

Allows to click **Remove** associated with the row number provided as input, in the **Found Assets** zone in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_FoundAssets_RemoveSingleAsset(String strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_SetAssetNotinStores

Allows to set the Asset Id in the **Asset Not in Stores** zone of the **Tracked Assets Audit** page based on the row number and Asset ID provided as input.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_SetAssetNotinStores (String strAssetId, String strRow)
```

Input Parameters: String strAssetId, String strRow

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_ClickSave

Allows the user to click **Save** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` or `physicalInventory_FirstCnt_TrckedAstsAudit_SetAssetNotinStores` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_ClickSave()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_ClickCancel

Allows the user to click **Cancel** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` or `physicalInventory_FirstCnt_TrckedAstsAudit_SetAssetNotinStores` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_ClickCancel()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_TrckedAstsAudit_ClickAssetCountFinal

Allows the user to click **Assets Count Final** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_FirstCnt_TrckedAstsAudit_ClickFound` or `physicalInventory_FirstCnt_TrckedAstsAudit_SetAssetNotinStores` method is called.

Example:

```
physicalInventory_FirstCnt_TrckedAstsAudit_ClickAssetCountFinal()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_EnterLotRecount

Allows the user to click **Enter Lot Recount** in **Recount Information** zone in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_Recount_EnterLotRecount()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_EnterLotRecount_SetLotCount

Allows the user to set the **Lot Count** value to the value provided as input in **Lot Counts Information** page.

This method should be called only after the `physicalInventory_Recount_EnterLotRecount` method is called.

Example:

```
physicalInventory_Recount_EnterLotRecount_SetLotCount(String strRow,
String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

physicalInventory_EnterLotRecount_LotCounts_DefaultOnHand

Allows the user to click **Default on Hand** in the **Lot Counts Information** page.

This method should be called only after the `physicalInventory_Recount_EnterLotRecount` method is called.

Example:

```
physicalInventory_EnterLotRecount_LotCounts_DefaultOnHand (String
strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_EnterLotRecount_LotCounts_ClickOnSave

Allows the user to click **Save** in the **Lot Counts Information** page.

This method should be called only after the `physicalInventory_Recount_EnterLotRecount` method is called.

Example:

```
physicalInventory_EnterLotRecount_LotCounts_ClickOnSave()
```


Input Parameters: NA
Return Type: void

physicalInventory_EnterLotRecount_LotCounts_ClickOnCancel

Allows the user to click **Cancel** in the **Lot Counts Information** page.

This method should be called only after the `physicalInventory_Recount_EnterLotRecount` method is called.

Example:

```
physicalInventory_EnterLotRecount_LotCounts_ClickOnCancel()
```

Input Parameters: NA
Return Type: void

physicalInventory_EnterLotRecount_LotCounts_ClickOnLotCountsFinal

Allows the user to click **LotCountsFinal** in the **Lot Counts Information** page.

This method should be called only after the `physicalInventory_Recount_EnterLotRecount` method is called.

Example:

```
physicalInventory_EnterLotRecount_LotCounts_ClickOnLotCountsFinal()
```

Input Parameters: NA
Return Type: void

physicalInventory_CompletedCount_ClickOnAdjustInventory

Allows the user to click **Adjust Inventory** of the specified row under **Completed Information** zone of the **Physical Inventory Header** page.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_CompletedCount_ClickOnAdjustInventory(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

physicalInventory_CompletedCount_ClickOnDiscard

Allows the user to click **Discard** of the specified row and column numbers under **Completed Information** zone in the **Physical Inventory Header** page.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_CompletedCount_ClickOnDiscard(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

physicalInventory_Recount_ClickReAuditSerialNumbers

Allows the user to click **ReAuditSerialNumbers** of the specified row number under **Recount Information** zone in the **Physical Inventory Header** page.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_Recount_ClickReAuditSerialNumbers(String rowNumber)
```

Input Parameters: String strRow, String strColumn

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_setLmtToSerialNumber

Allows the user to set the value for **Limit To Serial Number** field in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_Recount_ClickReAuditSerialNumbers` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_setLmtToSerialNumber (String strSerialNum)
```

Input Parameters: String strSerialNum

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_setLmtToBadgeNumber

Allows the user to set the value for **Limit To Badge Number** field in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_Recount_ClickReAuditSerialNumbers` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_setLmtToBadgeNumber (String strBadgeNum)
```

Input Parameters: String strBadgeNum

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_setLmtToPalletNumber

Allows the user to set the value for **Limit To Pallet Number** field in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_Recount_ClickReAuditSerialNumbers` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_setLmtToPalletNumber (String strPalletNum)
```

Input Parameters: String strPalletum

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_clickSearch

Allows the user to click **Search** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_ClickReAuditSerialNumbers` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_clickSearch()
```

Input Parameters:NA

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_CheckAllAssets

Allows the user to select the check box to select all the assets in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_clickSearch` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_CheckAllAssets()
```

Input Parameters:NA

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_CheckSingleAsset

Allows the user to select a check box to select a single asset in the **Tracked Assets Audit** page based on the row number provided as input.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_clickSearch` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_CheckSingleAsset(String strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_ClickFound

Allows the user to click **Found** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_CheckSingleAsset` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_ClickFound()
```

Input Parameters:NA

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_setLmtToSerialNumberFound

Allows the user to set the value for **Limit To Serial Number** field in the **Found Assets** zone in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_setLmtToSerialNumberFound(String strSerialNum)
```

Input Parameters: String strSerialNum

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_setLmtToBadgeNumberFound

Allows the user to set the value for **Limit To Badge Number** field in the **Found Assets** zone in the **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_setLmtToBadgeNumberFound(String strBadgeNum)
```

Input Parameters: String strBadgeNum

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_setLmtToPalletNumberFound

Allows the user to set the value for **Limit To Pallet Number** field in the **Found Assets** zone of **Tracked Assets Audit** page to the value provided as input.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_setLmtToPalletNumberFound(String strPalletNum)
```

Input Parameters: String strPalletum

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_FoundAssets_clickSearch

Allows the user to click Search in the **Found Assets** zone on the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_ClickReAuditSerialNumbers` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_FoundAssets_clickSearch()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_FoundAssets_RemoveSingleAsset

Allows the user to click **Remove** associated with the row number provided as input, in the **Found Assets** zone on the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_FoundAssets_RemoveSingleAsset(String strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_SetAssetNotinStores

Allows the user to set the Asset ID in the **Asset Not in Stores** zone on the **Tracked Assets Audit** page based on the row number and Asset ID provided as input.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_SetAssetNotinStores (String
strAssetId, String strRow)
```

Input Parameters: String strAssetId, String strRow

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_ClickSave

Allows the user to click **Save** on the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` or `physicalInventory_Recount_TrckedAstsAudit_SetAssetNotinStores` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_ClickSave ()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_ClickCancel

Allows the user to click **Cancel** in the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` or `physicalInventory_Recount_TrckedAstsAudit_SetAssetNotinStores` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_ClickCancel ()
```

Input Parameters: NA

Return Type: void

physicalInventory_Recount_TrckedAstsAudit_ClickAssetCountFinal

Allows the user to click **Assets Count Final** on the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_Recount_TrckedAstsAudit_ClickFound` or `physicalInventory_Recount_TrckedAstsAudit_SetAssetNotinStores` method is called.

Example:

```
physicalInventory_Recount_TrckedAstsAudit_ClickAssetCountFinal ()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_ClickEnterLotCount

Allows the user to click **Enter Lot Count** in the specified row of the **First Count Information** zone of the **Tracked Assets Audit** page.

This method should be called only after the `physicalInventory_searchAndSelectById` method is called.

Example:

```
physicalInventory_FirstCnt_ClickEnterLotCount(String strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_FirstCnt_LotCounts_SetLotCount

Allows the user to set the Lot Count value of the specified row number to the value provided as input in **Lot Counts** page.

This method should be called only after the `physicalInventory_FirstCnt_ClickEnterLotCount` method is called.

Example:

```
physicalInventory_FirstCnt_LotCounts_SetLotCount(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

physicalInventory_FirstCnt_LotCounts_DefaultOnHand

Allows the user to click **Default On Hand** button on the specified row number on the **Lot Counts** page.

This method should be called only after the `physicalInventory_FirstCnt_ClickEnterLotCount` method is called.

Example:

```
physicalInventory_FirstCnt_LotCounts_DefaultOnHand(String strRow)
```

Input Parameters: String strRow

Return Type: void

physicalInventory_FirstCnt_LotCounts_ClickOnSave

Allows the user to click **Save** on the **Lot Counts** page.

This method should be called only after the `physicalInventory_FirstCnt_ClickEnterLotCount` method is called.

Example:

```
physicalInventory_FirstCnt_LotCounts_ClickOnSave()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_LotCounts_ClickOnCancel

Allows the user to click **Cancel** on the **Lot Counts** page.

This method should be called only after the `physicalInventory_FirstCnt_ClickEnterLotCount` method is called.

Example:

```
physicalInventory_FirstCnt_LotCounts_ClickOnCancel()
```

Input Parameters: NA

Return Type: void

physicalInventory_FirstCnt_LotCounts_ClickOnLotCountsFinal

Allows the user to click **Lot Counts Final** on the Lot Counts page.

This method should be called only after the physicalInventory_FirstCnt_ClickEnterLotCount method is called.

Example:

```
physicalInventory_FirstCnt_LotCounts_ClickOnLotCountsFinal()
```

Input Parameters: NA

Return Type: void

Material Request/Issue and Return

MaterialRequest_searchAndSelectById

Searches for the Material Request Header based on the ID provided. If not found, opens the Material Request Header.

On success it opens the Material Request Header record. On failure, the Material Request Header is not selected and the flow fails.

Example:

```
MaterialRequest_searchAndSelectById(String strMaterialReqId)
```

Input Parameters: String strMaterialReqId

Return Type: void

MaterialRequest_openReqLines_IssueSingleLine

Clicks the **IssueAll** button under Open request lines zone in the Material Request Header page based on the row and column numbers provided. If not found, the **IssueAll** button will not be clicked and the flow fails.

This method should be called only after calling the MaterialRequest_searchAndSelectById method.

Example:

```
MaterialRequest_openReqLines_IssueSingleLine(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

MaterialRequest_openReqLines_NoIssueSingleLine

Clicks the **NoIssue** button under Open request lines zone in the Material Request Header page based on the row and column numbers provided. If not found, the **NoIssue** button will not be clicked and the flow fails.

This method should be called only after calling the MaterialRequest_searchAndSelectById method.

Example:

```
MaterialRequest_openReqLines_NoIssueSingleLine(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

MaterialRequest_openReqLines_Issuepartial

Clicks the **Issuepartial** button under Open request lines zone in the Material Request Header page based on the row and column numbers provided. If not found, the **Issuepartial** button will not be clicked and the flow fails.

This method should be called only after calling the MaterialRequest_searchAndSelectById method.

Example:

```
MaterialRequest_openReqLines_Issuepartial(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

MaterialRequest_openReqLines_SetLimitToBin

Sets value for Limit To Bin under Open request lines zone in the Material Request Header page based on the value provided. If not found, the Limit To Bin value will not be set and the flow fails.

This method should be called only after calling the MaterialRequest_searchAndSelectById method.

Example:

```
MaterialRequest_openReqLines_SetLimitToBin(String strBin)
```

Input Parameters: String strBin
Return Type: void

MaterialRequest_openReqLines_LmtToBin_Search

Clicks the **Search** button under Open request lines zone in the Material Request Header page. If not found, the **Search** button will not be clicked and the flow fails.

This method should be called only after calling the MaterialRequest_openReqLines_SetLimitToBin method.

Example:

```
MaterialRequest_openReqLines_LmtToBin_Search()
```

Input Parameters: NA
Return Type: void

MaterialRequest_IssueTrackAssets_SetSerialNumber

Set the Serial Number value on the Issue Tracked Assets page. If not found, the Serial number will not be set and the flow fails.

This method should be called only after calling either MaterialRequest_openReqLines_IssueSingleLine or MaterialRequest_openReqLines_IssueAll method.

Example:

```
MaterialRequest_IssueTrackAssets_SetSerialNumber(String strSrNumber)
```

Input Parameters: String strSrNumber
Return Type: void

MaterialRequest_IssueTrackAssets_SetBadgeNumber

Sets the Badge Number value on the Issue Tracked Assets page. If not found, the Badge number will not be set and the flow fails.

This method should be called only after calling either MaterialRequest_openReqLines_IssueSingleLine or MaterialRequest_openReqLines_IssueAll methods.

Example:

```
MaterialRequest_IssueTrackAssets_SetBadgeNumber (String strBadgeNumber)
```

Input Parameters: String strBadgeNumber)

Return Type: void

MaterialRequest_IssueTrackAssets_clickRefresh

Clicks the **Refresh** button on Issue Tracked Assets page. If not found, the flow fails.

This method should be called only after calling either MaterialRequest_openReqLines_IssueSingleLine or MaterialRequest_openReqLines_IssueAll method.

Example:

```
MaterialRequest_IssueTrackAssets_clickRefresh ()
```

Input Parameters: NA

Return Type: void

MaterialRequest_IssueTrackAssets_clickSave

Clicks the **Save** button on Issue Tracked Assets page. If not found, the flow fails.

This method should be called only after calling either MaterialRequest_IssueTrackAssets_CheckAllAssets or MaterialRequest_IssueTrackAssets_CheckSingleAsset methods.

Example:

```
MaterialRequest_IssueTrackAssets_clickSave ()
```

Input Parameters: NA

Return Type: void

MaterialRequest_IssueTrackAssets_clickCancel

Clicks the **Cancel** button on Issue Tracked Assets page. If not found, the flow fails.

This method should be called only after calling either MaterialRequest_IssueTrackAssets_CheckAllAssets or MaterialRequest_IssueTrackAssets_CheckSingleAsset methods.

Example:

```
MaterialRequest_IssueTrackAssets_clickCancel ()
```

Input Parameters: NA

Return Type: void

MaterialRequest_IssueTrackAssets_CheckAllAssets

Checks the check box to select all assets on the Issue Tracked Assets page. If not found, the flow fails.

This method should be called only after calling either `MaterialRequest_IssueTrackAssets_CheckAllAssets` or `MaterialRequest_IssueTrackAssets_CheckSingleAsset` method.

Example:

```
MaterialRequest_IssueTrackAssets_CheckAllAssets ()
```

Input Parameters: NA

Return Type: void

MaterialRequest_openReqLines_IssueAll

Allows to click the **IssueAll** button under Open request Lines Zone in the Material Request Header page. If not found, the flow fails.

This method should be called only after calling the `MaterialRequest_searchAndSelectById` method.

Example:

```
MaterialRequest_openReqLines_IssueAll ()
```

Input Parameters: NA

Return Type: void

MaterialRequest_openReqLines_NoIssueAll

Allows to click the **NoIssueAll** button under Open request Lines Zone in the Material Request Header page. If not found, the flow fails.

This method should be called only after calling the `MaterialRequest_searchAndSelectById` method.

Example:

```
MaterialRequest_openReqLines_NoIssueAll ()
```

Input Parameters: NA

Return Type: void

MaterialRequest_closedReqLines_Undo

Allows to click the Undo button under closed request lines zone of Material Request Header page based on the row and column numbers provided. If not found, the flow fails.

This method should be called only after the `MaterialRequest_IssueTrackAssets_clickSave` method is called.

Example:

```
MaterialRequest_closedReqLines_Undo (String strRow, String strCol)
```

Input Parameters: String strRow, String strCol

Return Type: void

MaterialRequest_closedReqLines_selectSupressIssueDtIs

Allows to select a value from the SupressIssueDetails drop down under closed request lines zone of Material request Header page based on the value provided. If not found the flow fails.

This method should be called only after the `MaterialRequest_IssueTrackAssets_clickSave` method is called.

Example:

```
MaterialRequest_closedReqLines_selectSupressIssueDtls(String Value)
```

Input Parameters: String Value

Return Type: void

MaterialRequest_closedReqLines_clickRefresh

Allows to click the **Refresh** button under closed request lines zone of Material Request Header page. If not found, the flow fails.

This method should be called only after the MaterialRequest_IssueTrackAssets_clickSave method is called.

Example:

```
MaterialRequest_closedReqLines_clickRefresh()
```

Input Parameters: NA

Return Type: void

MaterialRequest_canceledReqLines_setLimitToBin

Allows to set a value for Limit To Bin under canceled request lines zone of Material Request Header page based on the value provided. If not found, the flow fails.

This method should be called only when there are canceled request lines present on a Material Request Header.

Example:

```
MaterialRequest_canceledReqLines_setLimitToBin(String strValue)
```

Input Parameters: String strValue

Return Type: void

MaterialRequest_canceledReqLines_clickRefresh

Allows to click the **Refresh** button under canceled request lines zone of Material Request Header page. If not found, the flow fails.

This method should be called only when there are canceled request lines present on a Material Request Header.

Example:

```
MaterialRequest_canceledReqLines_clickRefresh()
```

Input Parameters: NA

Return Type: void

MaterialReturn_searchAndSelectById

Searches for the Material Return Header based on the ID provided. If found, opens the Material Return Header.

On success it will open the Material Return Header record. On failure, the Material Return Header is not selected, and the flow fails.

Example:

```
MaterialReturn_searchAndSelectById(String strMaterialRetId)
```

Input Parameters: String strMaterialRetId

Return Type: void

MaterialReturn_quickAdd_setStockCode

Allows to set the Stock Code value based on the value provided as input under quick add zone in Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called.

Example:

```
MaterialReturn_quickAdd_setStockCode(String strValue)
```

Input Parameters: String strValue

Return Type: void

MaterialReturn_quickAdd_setReturnQty

Allows the user to set the Return Quantity value based on the value provided as input under quick add zone in Material Return Header page.

Notes: This method should be called only after the MaterialReturn_searchAndSelectById method is called.

Example:

```
MaterialReturn_quickAdd_setReturnQty(String strValue)
```

Input Parameters: String strValue

Return Type: void

MaterialReturn_quickAdd_selectExpenseCode

Allows the user to select a value from expense code drop down based on the value provided as input, under quick add zone in Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called.

Example:

```
MaterialReturn_quickAdd_selectExpenseCode(String Value)
```

Input Parameters: String Value

Return Type: void

MaterialReturn_quickAdd_clickAddReturnBtn

Allows the user to click **Add Return** button under quick add zone in Material Return Header page.

This method should be called only after the method MaterialReturn_quickAdd_selectExpenseCode, MaterialReturn_quickAdd_setReturnQty, MaterialReturn_quickAdd_setStockCode, and MaterialReturn_quickAdd_setStockCode are called.

Example:

```
MaterialReturn_quickAdd_clickAddReturnBtn()
```

Input Parameters: NA

Return Type: void

MaterialReturn_openReturnLines_editQtySingleLine

Allows the user to click the **EditQty** button under open return lines zone in the Material Return Header page based on the row and column numbers provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_editQtySingleLine(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

MaterialReturn_openReturnLines_deleteSingleLine

Allows the user to click the **Delete** button in Open Return Lines zone in the Material Return Header page based on the row and column numbers provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_deleteSingleLine(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

MaterialReturn_openReturnLines_selectAllLines

Allows the user to click the **SelectAll** check box under Open Return Lines zone in the Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_selectAllLines()
```

Input Parameters: NA
Return Type: void

MaterialReturn_openReturnLines_selectSingleLine

Allows the user to click the check box to select a line under Open Return Lines zone in the Material Return Header page based on the rowNumber provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_selectSingleLine(String strRow)
```

Input Parameters: String strRow
Return Type: void

MaterialReturn_openReturnLines_deleteMultipleLines

Allows the user to click the **Delete** button under Open Return Lines zone in the Material Return Header page.

This method should be called only after the MaterialReturn_openReturnLines_selectSingleLine method is called and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_deleteMultipleLines()
```

Input Parameters: NA
Return Type: void

MaterialReturn_openReturnLines_setLimitToBin

Allows the user to set the Limit To Bin value under Open Return Lines zone in the Material Return Header page based on the value provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_setLimitToBin(String strValue)
```

Input Parameters: String strValue

Return Type: void

MaterialReturn_openReturnLines_clickSearch

Allows the user to click the **Search** button under Open Return Lines zone in the Material Return Header page.

This method should be called only after the MaterialReturn_openReturnLines_setLimitToBin method is called, and there are some open return lines.

Example:

```
MaterialReturn_openReturnLines_clickSearch()
```

Input Parameters: NA

Return Type: void

MaterialReturn_readyForReturn_setLimitToBin

Allows the user to set Limit to bin to the value provided as input under Ready for Return Lines zone in the Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_setLimitToBin(String strValue)
```

Input Parameters: String strValue

Return Type: void

MaterialReturn_readyForReturn_clickSearch

Allows the user to click Search button under Ready for return lines zone in Material Return Header page.

This method should be called only after calling the MaterialReturn_readyForReturn_setLimitToBin method.

Example:

```
MaterialReturn_readyForReturn_clickSearch()
```

Input Parameters: NA

Return Type: void

MaterialReturn_readyForReturn_AcceptAllLines

Allows the user to click **Accept All** button under Ready for Return Lines Zone in the Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_AcceptAllLines()
```

Input Parameters: NA

Return Type: void

MaterialReturn_readyForReturn_RejectAllLines

Allows the user to click **Reject All** button under Ready for Return Lines zone in the Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_RejectAllLines()
```

Input Parameters: NA

Return Type: void

MaterialReturn_readyForReturn_AcceptALL_SingleLine

Allows the user to click **Accept All** button associated to single return line under Ready for Return Lines zone in the Material Return Header page based on the row and column provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_AcceptALL_SingleLine(String strRow,String strColumn)
```

Input Parameters: String strRow,String strColumn

Return Type: void

MaterialReturn_readyForReturn_Partial_SingleLine

This method allows the user to click **Return Partial** button associated to single return line under Ready for return lines zone in Material Return Header page based on the row and column provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_Partial_SingleLine(String strRow,String strColumn)
```

Input Parameters: String strRow,String strColumn

Return Type: void

MaterialReturn_readyForReturn_Reject_SingleLine

Allows the user to click **Reject** button associated to single return line under Ready for Return Lines zone in Material Return Header page based on the row and column provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_Reject_SingleLine(String strRow,String strColumn)
```

Input Parameters: String strRow,String strColumn
Return Type: void

MaterialReturn_readyForReturn_selectAllLines

Allows the user to check the check box to select all the return lines under Ready for return lines zone in Material Return Header page.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_selectAllLines ()
```

Input Parameters: NA
Return Type: void

MaterialReturn_readyForReturn_selectMultipleLines

Allows the user to check the check boxes to select multiple return lines under Ready for return lines zone in Material Return Header page based on the row numbers provided as input.

Maximum six lines can be selected by calling this method. If there are more than six lines to be selected, then call this method multiple times based on the requirement. If there are less than six lines to be selected (say four lines), enter four inputs to the methods and leave the other inputs as blank.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some ready for return lines.

Example:

```
MaterialReturn_readyForReturn_selectMultipleLines (String  
strRow1,String strRow2,String strRow3,String strRow4,String  
strRow5,String strRow6)
```

Input Parameters: String strRow1,String strRow2,String strRow3,String
strRow4,String strRow5,String strRow6
Return Type: void

MaterialReturn_ClosedReturnLines_UndoSingleLine

This method allows the user to click the **Undo** button associated to single return line under Closed for Return Lines zone in the Material Return Header page based on the row and column provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some closed return lines.

Example

```
MaterialReturn_ClosedReturnLines_UndoSingleLine (String strRow,String  
strColumn)
```

Input Parameters: String strRow,String strColumn
Return Type: void

MaterialReturn_ClosedReturnLines_setLimitTo_StckItemDesc

Allows the user to set the Limit to Stock Item Description value under Closed Return Lines zone in the Material Return Header page based on the value provided as input.

This method should be called only after the MaterialReturn_searchAndSelectById method is called, and there are some closed return lines.

Example:

```
MaterialReturn_ClosedReturnLines_setLimitTo_StckItemDesc(String  
strValue)
```

Input Parameters: String strValue
Return Type: void

MaterialReturn_ClosedReturnLines_clickSearch

Allows the user to click the **search** button under Closed Return Lines zone in the Material Return Header page based on the value provided as input.

This method should be called only after the MaterialReturn_ClosedReturnLines_setLimitTo_StckItemDesc method is called.

Example:

```
MaterialReturn_ClosedReturnLines_clickSearch()
```

Input Parameters: NA
Return Type: void

MaterialReturn_EditAcceptedQty_setAcceptedQty

Allows the user to set Accepted Quantity value in the Edit Accepted Quantity window to value provided as input.

This method should be called only after the MaterialReturn_openReturnLines_editQtySingleLine method is called.

Example:

```
MaterialReturn_EditAcceptedQty_setAcceptedQty(String strValue)
```

Input Parameters: String strValue
Return Type: void

MaterialReturn_EditAcceptedQty_clickSave

Allows the user to click on **Save** button in the Edit Accepted Quantity window.

This method should be called only after the MaterialReturn_openReturnLines_editQtySingleLine method is called.

Example:

```
MaterialReturn_EditAcceptedQty_clickSave()
```

Input Parameters: NA
Return Type: NA

MaterialReturn_EditAcceptedQty_clickCancel

Allows the user to click the **Cancel** button in the Edit Accepted Quantity window.

This method should be called only after the MaterialReturn_openReturnLines_editQtySingleLine method is called.

Example:

```
MaterialReturn_EditAcceptedQty_clickCancel()
```

Input Parameters: NA
Return Type: void

ReceiptHeader_searchAndSelectById

Searches for the Receipt Header based on the ID provided. If found, will open the Receipt Header.

On success it will open the Receipt Header record. On failure, the Receipt Header is not selected, and the flow fails.

Example:

```
ReceiptHeader_searchAndSelectById(String strReceiptHeaderId)
```

Input Parameters: String strReceiptHeaderId

Return Type: void

Receiving_ReceiptLines_ReceiveAll_Singleline

Allows the user to click the **Receive All** button associated to single receipt line under open receipt lines zone in Receipt Header page based on the row and column provided as input.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be some open receipt lines for the receipt header.

Example:

```
Receiving_ReceiptLines_ReceiveAll_Singleline(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

Receiving_ReceiptLines_ReceivePartial_Singleline

Allows the user to click **Receive Partial** button associated to single receipt line under Open Receipt Lines zone in the Receipt Header page based on the row and column provided as input.

This method should be called only after calling ReceiptHeader_searchAndSelectById, and there should be some open receipt lines for the receipt header.

Example:

```
Receiving_ReceiptLines_ReceivePartial_Singleline(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

Receiving_ReceiptLines_Delete_Singleline

Allows the user to click **Delete** button associated to single receipt line under open receipt lines zone in Receipt Header page based on the row and column provided as input.

This method should be called only after calling ReceiptHeader_searchAndSelectById, and there should be some open receipt lines for the receipt header.

Example:

```
Receiving_ReceiptLines_Delete_Singleline(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

Receiving_openReceiptLines_SelectLimitToStoreRoom

Allows the user to select a value from Limit to Store Room drop down under open receipt lines zone in Receipt Header page based on the value provided as input.

This method should be called only after calling ReceiptHeader_searchAndSelectById, and there should be some open receipt lines for the receipt header.

Example:

```
Receiving_openReceiptLines_SelectLimitToStoreRoom(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_openReceiptLines_SetLimitToRsrcTypeDesc

Allows the user to set a value to Limit to resource type description to the value provided as input under open receipt lines zone in Receipt Header page.

This method should be called only after calling ReceiptHeader_searchAndSelectById, and there should be some open receipt lines for the receipt header.

Example:

```
Receiving_openReceiptLines_SetLimitToRsrcTypeDesc(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_openReceiptLines_clickSearch

Allows the user to click the **Search** button under Open Receipt Lines zone in the Receipt Header page.

This method should be called only after calling the Receiving_openReceiptLines_SelectLimitToStoreRoom or Receiving_openReceiptLines_SetLimitToRsrcTypeDesc methods.

Example:

```
Receiving_openReceiptLines_clickSearch()
```

Input Parameters: NA

Return Type: void

Receiving_receivePartial_SetQuantity

Allows the user to set a value to quantity to the value provided as input in Receipt Partial window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receiving_receivePartial_SetQuantity(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_receivePartial_SetComments

A the user to set a value to Comments to the value provided as input in Receipt Partial window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receiving_receivePartial_SetComments(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_receivePartial_checkForceComplete

Allows the user to check the **Force Complete** check box in Receipt Partial window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receiving_receivePartial_checkForceComplete(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receivepartial_ReceiveStock_AddBinLine

Allows the user to click **Add bin line** button in the Receive Stock window.

This method should be called only after calling the Receiving_ReceiptLines_ReceiveAll_Singleline method.

Example:

```
Receivepartial_ReceiveStock_AddBinLine()
```

Input Parameters: NA

Return Type: void

Receivepartial_ReceiveStock_DeleteBinLine

Allows the user to click **Delete bin line** button in Receive Stock window based on the row number provided as input.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receivepartial_ReceiveStock_DeleteBinLine(String strRow)
```

Input Parameters: String strRow

Return Type: void

Receivepartial_ReceiveStock_SetBin

Allows the user to set a value to Bin to the value provided as input in Receipt Stock window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receivepartial_ReceiveStock_SetBin(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receivepartial_ReceiveStock_SelectBinType

Allows the user to select a value from the Bin type drop down based on the value provided as input in Receive Stock window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receivepartial_ReceiveStock_SelectBinType(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue)

Return Type: void

Receivepartial_ReceiveStock_clickAccept

Allows the user to click the **Accept** button in Receive Stock window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receivepartial_ReceiveStock_clickAccept()
```

Input Parameters: NA

Return Type: void

Receivepartial_ReceiveStock_clickCancel

Allows the user to click **Cancel** button in the Receive Stock window.

This method should be called only after calling the Receiving_ReceiptLines_ReceivePartial_Singleline method.

Example:

```
Receivepartial_ReceiveStock_clickCancel()
```

Input Parameters: NA

Return Type: void

Receiving_InReceiptLines_AcceptSingleLine

Allows the user to click the **Accept** button associated to a single inreceipt Line under InReceipt Lines zone of a receipt header page based on the row and column numbers provided as the input.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be InReceipt lines for the receipt header.

Example:

```
Receiving_InReceiptLines_AcceptSingleLine(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

Receiving_InReceiptLines_ReturnSingleLine

Allows the user to click the **Return** button associated to a single inreceipt Line under InReceipt Lines zone of a receipt header page based on the row and column numbers provided as the input.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be InReceipt lines for the receipt header.

Example:

```
Receiving_InReceiptLines_ReturnSingleLine(String strRow, String
strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

Receiving_InReceiptLines_UndoSingleLine

Allows the user to click the **Undo** button associated to a single inreceipt Line under InReceipt Lines zone of a receipt header page based on the row and column numbers provided as the input.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be InReceipt lines for the receipt header.

Example:

```
Receiving_InReceiptLines_UndoSingleLine(String strRow, String
strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

Receiving_InReceiptLines_SelectLimitToStoreRoom

Allows the user to select a value from Limit to Storeroom drop down under InReceipt Lines zone of a receipt header page based on the value provided as the input.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be InReceipt lines for the receipt header.

Example:

```
Receiving_InReceiptLines_SelectLimitToStoreRoom(String strValue)
```

Input Parameters: String strValue
Return Type: void

Receiving_InReceiptLines_SetLmtToRsrcTypeDesc

Allows the user to set Limit to Resource Type Description value under InReceipt Lines zone of a receipt header page based on the value provided as the input.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be InReceipt lines for the receipt header.

Example:

```
Receiving_InReceiptLines_SetLmtToRsrcTypeDesc(String strValue)
```

Input Parameters: String strValue
Return Type: void

Receiving_InReceiptLines_clickSearch

Allows the user to click the **Search** button under the InReceipt Lines zone of the Receipt Header page.

This method should be called only after calling the ReceiptHeader_searchAndSelectById method, and there should be InReceipt lines for the receipt header.

Example:

```
Receiving_InReceiptLines_clickSearch()
```

Input Parameters: NA
Return Type: void

Receiving_InReceiptLines_AcceptStock_SetQty

Allows the user to set Quantity in accept Stock window based on the value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InReceiptLines_AcceptStock_SetQty(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_InReceiptLines_AcceptStock_SetComments

Allows the user to set Comments in accept Stock window based on the value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InReceiptLines_AcceptStock_SetComments(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_InReceiptLines_AcceptStock_CheckForceComplete

Allows the user to check the **Force Complete** check box in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InReceiptLines_AcceptStock_CheckForceComplete ()
```

Input Parameters: NA

Return Type: void

Receiving_InreceiptLines_AcceptStock_AddBinLine

Allows the user to click the **Add Bin Line** button in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_AddBinLine ()
```

Input Parameters: NA

Return Type: void

Receiving_InreceiptLines_AcceptStock_DeleteBinLine

Allows the user to click **Delete Bin Line** button in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_DeleteBinLine(String strRow)
```

Input Parameters: String strRow

Return Type: void

Receiving_InreceiptLines_AcceptStock_SetBin

Allows the user to Set Bin value associated to a single line based on the row and column provided as inputs in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example

```
Receiving_InreceiptLines_AcceptStock_SetBin(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_SelectBinType

Allows the user to select a value from **Bin Type** drop down associated to each line based on the value and row number provided as inputs in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_SelectBinType(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_clickAccept

Allows the user to click **Accept** button in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_clickAccept()
```

Input Parameters: NA

Return Type: void

Receiving_InreceiptLines_AcceptStock_clickCancel

This method allows the user to click **Cancel** button in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_clickCancel()
```

Input Parameters: NA

Return Type: void

Receiving_InreceiptLines_AcceptStock_AddLotLine

This method allows the user to click the **AddLotLine** button in accept Stock window.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_AddLotLine()
```

Input Parameters: NA

Return Type: void

Receiving_InreceiptLines_AcceptStock_DeleteLotLine

Allows the user to click the **DeleteLotLine** button in accept Stock window.

This method should be called only after calling the `Receiving_InReceiptLines_AcceptSingleLine` method.

Example:

```
Receiving_InreceiptLines_AcceptStock_DeleteLotLine(String strRow)
```

Input Parameters: String strRow

Return Type: void

Receiving_InreceiptLines_AcceptStock_setLotId

Allows the user to set a LotId associated to each line in accept Stock window based on the row number and value provided as input.

This method should be called only after calling the `Receiving_InReceiptLines_AcceptSingleLine` method.

Example:

```
Receiving_InreceiptLines_AcceptStock_setLotId(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_setQuantity

Allows the user to set Quantity associated to each line in accept Stock window based on the row number and value provided as input.

This method should be called only after calling the `Receiving_InReceiptLines_AcceptSingleLine` method.

Example:

```
Receiving_InreceiptLines_AcceptStock_setQuantity(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_setExpirationDate

Allows the user to set Expiration date associated to each line in accept Stock window based on the row number and value provided as input.

This method should be called only after calling the `Receiving_InReceiptLines_AcceptSingleLine` method.

Example:

```
Receiving_InreceiptLines_AcceptStock_setExpirationDate(String strRow, String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_setVendorExpDate

Allows the user to set Vendor Expiry Date associated to each line in accept Stock window based on the row number and value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_setVendorExpDate(String strRow,  
String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_LotSetBin

Allows the user to set LotBin associated to each line in accept Stock window based on the row number and value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_LotSetBin(String strRow, String  
strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_AcceptStock_LotSelectBinType

Allows the user to select a value from Bin Type drop down associated to each line in accept Stock window based on the row number and value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_AcceptSingleLine method.

Example:

```
Receiving_InreceiptLines_AcceptStock_LotSelectBinType(String strRow,  
String strValue)
```

Input Parameters: String strRow, String strValue

Return Type: void

Receiving_InreceiptLines_ReturnStock_SelectTypeofReturn

Allows the user to select a value from the Type of Return drop down in Return Stock window based on the value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_ReturnSingleLine method.

Example:

```
Receiving_InreceiptLines_ReturnStock_SelectTypeofReturn(String  
strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_InreceiptLines_ReturnStock_SetQuantity

Allows the user to set a value to Quantity in Return Stock window based on the value provided as input.

This method should be called only after calling the Receiving_InReceiptLines_ReturnSingleLine method.

Example:

```
Receiving_InreceiptLines_ReturnStock_SetQuantity(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_InreceiptLines_ReturnStock_clickForceCmpl

Allows the user to click the **Force Complete** button in Return Stock window.

This method should be called only after calling the Receiving_InReceiptLines_ReturnSingleLine method.

Example:

```
Receiving_InreceiptLines_ReturnStock_clickForceCmpl(String strValue)
```

Input Parameters: String strValue

Return Type: void

Receiving_InreceiptLines_ReturnStock_clickReturn

Allows the user to click the **Return** button in Return Stock window.

This method should be called only after calling the Receiving_InReceiptLines_ReturnSingleLine method.

Example:

```
Receiving_InreceiptLines_ReturnStock_clickReturn()
```

Input Parameters: NA

Return Type: void

Receiving_InreceiptLines_ReturnStock_clickCancel

Allows the user to click the **Cancel** button in the Return Stock window.

This method should be called only after calling the Receiving_InReceiptLines_ReturnSingleLine method.

Example:

```
Receiving_InreceiptLines_ReturnStock_clickCancel()
```

Input Parameters: NA

Return Type: void

Receiving_AcceptReturnDetail_clickUndo

Allows the user to click the **Undo** button associated to each line in Accept return detail zone in receipt Header page based on the row and column provided as input.

This method should be called only after calling the Receiving_InreceiptLines_AcceptStock_clickAccept method.

Example:

```
Receiving_AcceptReturnDetail_clickUndo(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

Stock Transfer

stockTransfer_searchAndSelectById

Searches for the stock transfer based on the ID provided. If found, opens the stock transfer record.

On success it opens the stock transfer record. On failure, the Stock Transfer record is not selected, and the flow fails.

Example:

```
stockTransfer_searchAndSelectById(String strStkTranId)
```

Input Parameters: String strStkTranId
Return Type: void

stockTransfer_QuickAdd_StockCode

Assigns the input string as a Stock Code in the Quick Add Zone. On success, it assigns the Stock Code. On failure, the Stock Code is not assigned, and the flow fails

The Stock Transfer record should be available in the UI by executing the stockTransfer_searchAndSelectById method successfully.

Example:

```
stockTransfer_QuickAdd_StockCode(String stkCode)
```

Input Parameters: String stkCode
Return Type: void

stockTransfer_QuickAdd_RequestedQty

Assigns the input string as a Requested Quantity in the Quick Add Zone. On success, it assigns the Requested Quantity. On failure, the requested quantity is not assigned the flow fails.

The Stock Transfer record should be available in the UI by executing the stockTransfer_searchAndSelectById method successfully.

Example:

```
stockTransfer_QuickAdd_RequestedQty(String requestedQty)
```

Input Parameters: String requestedQty
Return Type: void

stockTransfer_QuickAdd_AddRequest

Adds the request to transfer stock from one storeroom to another. On success the request is added successfully. On failure, the request is not added.

The Stock Transfer record should be available in the UI by successful execution of stockTransfer_searchAndSelectById. The Stock Code and Requested Quantity fields should be populated.

Example:

```
stockTransfer_QuickAdd_AddRequest(String stkCode,String requestedQty)
```

Input Parameters: String stkCode,String requestedQty
Return Type: void

stockTransfer_QuickAdd_Cancel

Cancels the request to transfer stock from one storeroom to another. On success the request is not added successfully.

The Stock Transfer record should be available in the UI by successful execution of `stockTransfer_searchAndSelectById`. The Stock Code and Requested Quantity fields should be populated.

Example

```
stockTransfer_QuickAdd_Cancel(String stkCode,String requestedQty)
```

Input Parameters: String stkCode,String requestedQty

Return Type: void

stockTransfer_OpenReqLines_SelectAll

Selects all the open request lines available for a stock transfer record. On success all the request lines will be selected. On failure, the request lines will not be selected.

The Stock Transfer Open Request lines should be available in the stock transfer record.

Example:

```
stockTransfer_OpenReqLines_SelectAll()
```

Input Parameters: NA

Return Type: void

stockTransfer_OpenReqLines_EditQuantity

Opens the Stock Transfer Request line for modify the quantity. On success the Stock Transfer Request line is displayed. On failure, the stock transfer request line is not displayed.

The Stock Transfer record should be available along with the Open Request Lines.

Example:

```
stockTransfer_OpenReqLines_EditQuantity(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

stockTransfer_OpenReqLines_Delete

Description: This function will delete the Open Request Lines. On Success the lines are successfully deleted. On failure the lines are not deleted.

The Stock Transfer record should be available along with the Open Request Lines.

Example

```
stockTransfer_OpenReqLines_Delete(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

stockTransfer_EditQuantity_RequestedQuantity

Modifies the requested quantity for a open request line. On Success, the quantity is successfully modified. On failure, the quantity is not modified.

The Stock Transfer record should be available along with the Open Request Lines.

Example:

```
stockTransfer_EditQuantity_RequestedQuantity(String qty)
```

Input Parameters: String qty

Return Type: void

stockTransfer_EditQuantity_SaveQuantity

Saves the modifications done for the requested quantity for a open request line.

The Stock Transfer record should be available along with the Open Request Lines and the quantity should be modified for the open request line.

Example:

```
stockTransfer_EditQuantity_SaveQuantity(String qty)
```

Input Parameters: String qty

Return Type: void

stockTransfer_EditQuantity_CancelQuantity

Cancels the modifications done for the edited quantity of a open request line for a Stock Transfer. On Success, the modifications done for the edited quantity will not be saved. On failure, the modifications done for the quantity will be saved.

Stock Transfer record should be available along with the Open Request Lines and the quantity should be modified for a open request line.

Example:

```
stockTransfer_EditQuantity_CancelQuantity(String qty)
```

Input Parameters: String qty

Return Type: void

stockTransfer_OpenReqLines_IssueAll

Issues all the requested quantity for a stock transfer. On success, the requested quantity will be issued to the requested storeroom. On failure, the requested quantity will not be issue.

The Stock Transfer record should be available along with the Open Request Lines.

Example:

```
stockTransfer_OpenReqLines_IssueAll(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

stockTransfer_OpenReqLines_IssueSubstituePartial_Click

Opens the request line for partial issue of the requested quantity. On success, the request line will be available for modification. On failure, the requested line will not be displayed.

The Stock Transfer record should be available along with the Open Request Lines.

Example:

```
stockTransfer_OpenReqLines_IssueSubstituePartial_Click(String strRow,  
String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

stockTransfer_OpenReqLines_IssueSubstitutePartial_OrigStkItemIssueQty

Modifies the original stock item issue quantity. On success, the original stock item issue quantity will be modified. On failure the original stock item issue quantity will not be modified.

The Stock Transfer record should be available along with the Open Request Lines and the line should be available for issuing partial quantity.

Example:

```
stockTransfer_OpenReqLines_IssueSubstitutePartial_OrigStkItemIssueQty(
String OrigStkItemIssueQty)
```

Input Parameters: String OrigStkItemIssueQty

Return Type: void

stockTransfer_OpenReqLines_IssueSubstitutePartial_AltStkItem

Modifies the alternate stock item issue quantity. On success, the alternate stock item issue quantity will be modified. On failure the original stock item issue quantity will not be modified.

The Stock Transfer record should be available along with the Open Request Lines and the line should be available for issuing partial quantity.

Example:

```
stockTransfer_OpenReqLines_IssueSubstitutePartial_AltStkItem(String
strRow, String strColumn,String AltStkItemIssueQty)
```

Input Parameters: String strRow, String strColumn,String
AltStkItemIssueQty

Return Type: void

stockTransfer_OpenReqLines_IssueSubstitutePartial_SubStkItem

Modifies the substitute stock item detail and issue quantity. On success, the substitute stock item detail and issue quantity will be modified. On failure the substitute stock item detail and issue quantity will not be modified.

The Stock Transfer record should be available along with the Open Request Lines and the line should be available for issuing partial quantity.

Example:

```
stockTransfer_OpenReqLines_IssueSubstitutePartial_SubStkItem(String
strRow, String strColumn,String SubStkItemDetailId,String
SubStkItemDetailIssueQty)
```

Input Parameters: String strRow, String strColumn,String
SubStkItemDetailId,String SubStkItemDetailIssueQty

Return Type: void

stockTransfer_OpenReqLines_IssueSubstituePartial_Save

Saves the Substitue /Partial quantity to be issued details. On success, the Substitue/Partial quantity to be issued are successfully saved. On failure, the Substitue /Partial quantity to be issued are not successfully saved.

The Stock Transfer record should be available along with the Open Request Lines and the line should be available for issuing partial quantity.

Example:

```
stockTransfer_OpenReqLines_IssueSubstituePartial_Save(String
OrigStkItemIssueQty,String AltStkItemIssueQty,String
SubStkItemDetailId,String SubStkItemDetailIssueQty)
```

```

Input Parameters: String OrigStkItemIssueQty, String
AltStkItemIssueQty, String SubStkItemDetailId, String
SubStkItemDetailIssueQty
Return Type: void

```

stockTransfer_OpenReqLines_IssueSubstituePartial_Cancel

Cancels the Substitue/Partial quantity to be issued details. On success, the Substitue /Partial quantity to be issued are successfully cancelled. On failure, the Substitue /Partial quantity to be issued are not successfully cancelled.

The Stock Transfer record should be available along with the Open Request Lines and the line should be available for issuing partial quantity.

Example:

```

stockTransfer_OpenReqLines_IssueSubstituePartial_Cancel (String
OrigStkItemIssueQty, String AltStkItemIssueQty, String
SubStkItemDetailId, String SubStkItemDetailIssueQty)

```

```

Input Parameters: String OrigStkItemIssueQty, String
AltStkItemIssueQty, String SubStkItemDetailId, String
SubStkItemDetailIssueQty
Return Type: void

```

stockTransfer_OpenReqLines_NoIssue

Does not issue any requested quantity to the requesting storeroom. On success, no quantity is issued to the requesting storeroom.

The Stock Transfer record should be available along with the Open Request Lines.

Example:

```

stockTransfer_OpenReqLines_NoIssue (String strRow, String strColumn)
Input Parameters: String strRow, String strColumn
Return Type: void

```

stockTransfer_OpenReqLines_IssueAsset

Opens the stock transfer record to issue tracked assets to the requesting storeroom. On success, it will open the stock transfer record to issue tracked assets to the requesting storeroom. On failure the stock transfer record is not opened.

Example:

```

stockTransfer_OpenReqLines_IssueAsset (String strRow, String strColumn)

Input Parameters: String strRow, String strColumn
Return Type: void

```

stockTransfer_OpenReqLines_IssueTrackedAssets_SerialNumber

Assigns the input to the serial number of the tracked asset. On success it will assign the serial number. On failure the serial number is not assigned.

The Stock Transfer record should be available and the Tracked Asset request line should be available.

Example:

```

stockTransfer_OpenReqLines_IssueTrackedAssets_SerialNumber (String
serialNumber)

Input Parameters: String serialNumber
Return Type: void

```


stockTransfer_OpenReqLines_IssueTrackedAssets_BadgeNumber

Assigns the input to the badge number of the tracked asset. On success it will assign the badge number. On failure the badge number is not assigned.

The Stock Transfer record should be available and the Tracked asset request line should be available.

Example:

```
stockTransfer_OpenReqLines_IssueTrackedAssets_BadgeNumber (String
badgeNumber)
Input Parameters: String badgeNumber
Return Type: void
```

stockTransfer_OpenReqLines_IssueTrackedAssets_Select

Select the Tracked Assets to be transferred to the requesting storeroom. On success the tracked assets will be selected. On failure the tracked assets will not be selected.

The Stock Transfer record should be available and the Tracked Asset request line should be available.

Example:

```
stockTransfer_OpenReqLines_IssueTrackedAssets_Select (String strRow,
String strColumn,String serialNumber,String badgeNumber)

Input Parameters: String strRow, String strColumn,String
serialNumber,String badgeNumber
Return Type: void
```

stockTransfer_OpenReqLines_IssueTrackedAssets_Save

Issues the tracked assets to the requesting storeroom. On success, the tracked assets will be issued to the requesting storeroom. On failure the assets are not issued.

The Stock Transfer record should be available and the Tracked asset request line should be available.

Example:

```
stockTransfer_OpenReqLines_IssueTrackedAssets_Save ()

Input Parameters: NA
Return Type: void
```

stockTransfer_OpenReqLines_IssueTrackedAssets_Cancel

Cancels the request to issue the tracked assets to the requesting storeroom. On success, the tracked assets will be not issued to the requesting storeroom. On failure the assets are issued.

The Stock Transfer record should be available and the Tracked asset request line should be available.

Example:

```
stockTransfer_OpenReqLines_IssueTrackedAssets_Cancel ()

Input Parameters: NA
Return Type: void
```

stockTransfer_OpenReqLines_IssueTrackedAssets_Refresh

Resets all the values. On success, all the values will be null. On failure the values are not reset.

The Stock Transfer record should be available and the Tracked asset request line should be available.

Example:

```
stockTransfer_OpenReqLines_IssueTrackedAssets_Refresh()
```

Input Parameters: NA

Return Type: void

stockTransfer_IssueLot

Clicks the Issue Lot for issue lot based stock items.

The Stock Transfer record should be available and lot managed stock items should be available in the Open Request Lines.

Example:

```
stockTransfer_IssueLot(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

stockTransfer_IssueLot_Select

Select the stock items to be issued. On success, the lot to be issued is selected. On failure the lot to be issued is not selected.

The Stock Transfer record should be available and lot managed stock items should be available in the Open Request Lines.

Example:

```
stockTransfer_IssueLot_Select(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

stockTransfer_IssueLot_Qty

Takes the input and assign it to the quantity to be issued for the lot. On success, the quantity is assigned successfully. On failure, the quantity is not assigned successfully.

The Stock Transfer record should be available and lot managed stock items should be available in the Open Request Lines.

Example:

```
stockTransfer_IssueLot_Qty(String strRow, String strColumn,String  
issueQty)
```

Input Parameters: String strRow, String strColumn,String issueQty

Return Type: void

stockTransfer_IssueLot_Save

Save the request for lot managed stock items. On success, the lot managed stock items are transferred to the requesting storeroom. On failure the lot managed stock items are not transferred.

The Stock Transfer record should be available and lot managed stock items should be available in the Open Request Lines.

Example:

```
stockTransfer_IssueLot_Save()
```

Input Parameters: NA

Return Type: void

stockTransfer_IssueLot_Cancel

Cancels the request for lot managed stock items. On success, the lot managed stock items are not transferred to the requesting storeroom. On failure the lot managed stock items are transferred.

The Stock Transfer record should be available and lot managed stock items should be available in the Open Request Lines.

Example:

```
stockTransfer_IssueLot_Cancel()
```

Input Parameters: NA

Return Type: void

stockTransfer_IssueLot_Refresh

Resets all the values of the requested quantity.

The Stock Transfer record should be available and lot managed stock items should be available in the Open Request Lines.

Example:

```
stockTransfer_IssueLot_Refresh()
```

Input Parameters: NA

Return Type: void

stockTransfer_LimitToBinSearch_binNumber

Assigns the input to the bin number to be searched. On success, the bin number is assigned to the input string. On failure the bin number is not assigned.

Stock transfer record should be available along with open request lines

Example:

```
stockTransfer_LimitToBinSearch_binNumber(String binNumber)
```

Input Parameters: String binNumber

Return Type: void

stockTransfer_LimitToBinSearch_Search

Searches according to the bin number.

The Stock transfer record should be available along with open request lines.

Example:

```
stockTransfer_LimitToBinSearch_Search(String binNumber)
```

Input Parameters: String binNumber

Return Type: void

stockTransfer_DelOpenRequestLines_Select

Selects the request lines necessary for deletion. On success, the lines are successfully selected. On failure the lines are not selected.

The Stock transfer record should be available along with open request lines.

Example:

```
stockTransfer_DelOpenRequestLines_Select(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

stockTransfer_DelOpenRequestLines_Delete

Deletes the request lines marked for deletion. On success, the lines are successfully deleted. On failure the lines are not deleted.

The Stock transfer record should be available along with open request lines.

Example:

```
stockTransfer_DelOpenRequestLines_Delete(String rowNumber)
```

Input Parameters: String rowNumber
Return Type: void

stockTransfer_StockTransferReceipt

Opens the StockTransferReceipt page.

Example:

```
stockTransfer_StockTransferReceipt(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn
Return Type: void

stockTransfer_StockTransferReceipt_click

Accepts all the stock transferred to the requesting storeroom. On success, the stock is successfully transferred to the requesting storeroom. On failure the stock is not transferred.

The Stock Transfer Receipt page should be available.

Example:

```
stockTransfer_StockTransferReceipt_click()
```

Input Parameters: NA
Return Type: void

Blanket Contract

blanketContract_searchAndSelectById

Searches for the blanket contract based on the id provided. If found, opens the blanket contract record. On success it will open the blanket contract record. On failure, the blanket contract record is not selected, and the flow fails.

Example:

```
blanketContract_searchAndSelectById(String strBCId)
```

Input Parameters: String strBCId
Return Type: void

blanketContract_AdditionalProcesses_CreateRevForTimePeriod_Click

Creates a revision for time period. On success, a new revision page is available. On failure a new revision page is not available.

The Blanket Contract record should be available.

Example:

```
blanketContract_AdditionalProcesses_CreateRevForTimePeriod_Click()
```

Input Parameters: NA

Return Type: void

blanketContract_AdditionalProcesses_EnterInitiationDate

Takes the input and assigns to the initiation date. On success, the initiation date is successfully initialized. On failure, the date is not initialized.

The Blanket Contract record should be available.

Example:

```
blanketContract_AdditionalProcesses_EnterInitiationDate(String  
initiationDate)
```

Input Parameters: String (initiationDate)

Return Type: void

blanketContract_AdditionalProcesses_CreateRevForTimePeriod_Save

Saves the new revision for time period created. On success, the revision is created successfully. On failure, the revision is not created successfully.

The Blanket Contract record should be available and a new revision should be available.

Example:

```
blanketContract_AdditionalProcesses_CreateRevForTimePeriod_Save(String  
currRevNumber)
```

Input Parameters: String currRevNumber

Return Type: void

blanketContract_AdditionalProcesses_CreateRevForTimePeriod_Cancel

Cancels the new revision for time period created. On success, the revision is not created successfully. On failure the revision is created successfully.

The Blanket Contract record should be available and a new revision should be available.

Example:

```
blanketContract_AdditionalProcesses_CreateRevForTimePeriod_Cancel(Stri  
ng initiationDate)
```

Input Parameters: String initiationDate

Return Type: void

blanketContract_AdditionalProcesses_CreateRevision

Creates a new revision. On success, the revision is created successfully. On failure the revision is not created successfully.

The Blanket Contract record should be available.

Example:

```
blanketContract_AdditionalProcesses_CreateRevision()
```

Input Parameters:

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_Select

Allows the user to create a purchase order. On success, a purchase order page is available. On failure a purchase order page is not available.

The Blanket Contract record should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_Select()
```

Input Parameters: NA

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_EnterDocument Type

Assigns a document type based on the input provided. On success, the document type is successfully assigned. On failure the document type is not assigned.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_EnterDocumentType(String docType)
```

Input Parameters: String docType)

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_ShipToLoc

Assigns Ship to Location based on the input provided. On success, the Ship to Location is successfully assigned. On failure the Ship to Location is not assigned.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_ShipToLoc(String shipToLoc)
```

Input Parameters: String shipToLoc

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_ReqdByDate

Assigns Required by Date based on the input provided. On success, the Required by Date is successfully assigned. On failure the Required by Date is not assigned.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_ReqdByDate(String reqdByDate)
```

Input Parameters: String reqdByDate

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_Storeroom

Assigns Storeroom based on the input provided. On success, the Storeroom is successfully assigned. On failure the Storeroom is not assigned.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_Storerroom(String strRow, String strColumn,String storerroomOption)
```

Input Parameters: String strRow, String strColumn,String storerroomOption

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_Qty

Assigns the quantity based on the input provided. On success, the quantity is successfully assigned. On failure the quantity is not assigned.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_Qty(String strRow, String strColumn,String qty)
```

Input Parameters: String strRow, String strColumn,String qty

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_SelectAllLines

Selects all the Purchase order lines. On success, all the Purchase order lines will be selected. On failure no lines will be selected.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_SelectAllLines()
```

Input Parameters: NA

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_SingleBCLine_Select

Selects a single blanket contract line. On success, a single blanket contract line will be selected. On failure a single blanket contract line will not be selected.

The Blanket Contract record should be available and purchase order page is available

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_SingleBCLine_Select(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_Save

Creates a purchasing document. On success, the Purchase order is created. On failure the Purchase order is not created.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_Save()
```

Input Parameters: NA

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_Cancel

Cancels a single Blanket contract line. On success, the blanket contract line is cancelled. On failure the blanket contract line is not cancelled.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_Cancel (String
strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseOrder_MultipleBCLine_Save

Saves a multiple Blanket contract lines. On success, the blanket contract lines are saved. On failure the blanket contract lines are not saved.

The Blanket Contract record should be available and purchase order page is available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseOrder_MultipleBCLine
_Save (String strRow1, String strRow2, String strRow3, String
strRow4, String strRow5, String strRow6)
```

Input Parameters: String strRow1, String strRow2, String strRow3, String strRow4, String strRow5, String strRow6

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Select

Allows the user to create a purchase requisition. on success, a purchase requisition page is available. On failure a purchase order requisition is not available.

The Blanket Contract record should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Select (S
tring docType)
```

Input Parameters: String docType

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_EnterDocumentType

Assigns a document type based on the input provided. On success, the document type is successfully assigned. On failure the document type is not assigned.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_EnterDoc
umentType (String docType)
```

Input Parameters: String docType

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_ShipToLoc

Assigns Ship to Location based on the input provided. On success, the Ship to Location is successfully assigned. On failure the Ship to Location is not assigned.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_ShipToLoc(String shipToLoc)
```

Input Parameters: String shipToLoc
Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_ReqdByDate

Assigns the Required by Date based on the input provided. On success, the Required by Date is successfully assigned. On failure the Required by Date is not assigned.

Notes: Blanket Contract record should be available and Purchase Requisition page should be available

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_ReqdByDate(String reqdByDate)
```

Input Parameters: String reqdByDate
Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Storeroom

Assigns Storeroom based on the input provided. On success, the Storeroom is successfully assigned. On failure the Storeroom is not assigned.

Notes: Blanket Contract record should be available and Purchase Requisition page should be available

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Storeroom(String strRow, String strColumn,String storeroomOption)
```

Input Parameters: String strRow, String strColumn,String storeroomOption
Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Qty

Assigns the quantity based on the input provided. On success, the quantity is successfully assigned. On failure the quantity is not assigned.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Qty(String strRow, String strColumn,String qty)
```

Input Parameters: String strRow, String strColumn,String qty
Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_SelectAllLines

Selects all the Purchase requisition lines. On success, all the Purchase requisition lines will be selected. On failure no lines will be selected.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_SelectAllLines()
```

Input Parameters: NA

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_SingleBCLine_Select

This function will select a single blanket contract line. On success, a single blanket contract line will be selected. On failure a single blanket contract line will not be selected.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_SingleBCLine_Select(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn)

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Save

Creates a purchasing Requisition. On success, the Purchase Requisition is created. On failure the Purchase Requisition is not created.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Save()
```

Input Parameters: NA

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Cancel

Cancels creating a purchasing Requisition. On success, the Purchase Requisition is not created. On failure the Purchase Requisition is created.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Cancel(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

blanketContract_AdditionalProcesses_CreatePurchaseRequisition_MultipleBCLine_Save

Save a multiple Blanket contract lines. On success, the blanket contract lines are saved. On failure the blanket contract lines are not saved.

The Blanket Contract record should be available and Purchase Requisition page should be available.

Example:

```
blanketContract_AdditionalProcesses_CreatePurchaseRequisition_Multiple
BCLine_Save(String strRow1,String strRow2,String strRow3,String
strRow4,String strRow5,String strRow6)
```

Input Parameters: String strRow1,String strRow2,String strRow3,String strRow4,String strRow5,String strRow6

Return Type: void

Preventive Maintenance

preventiveMaintenance_searchAndSelectById

Searches for the Preventive Maintenance Trigger based on the ID provided. If found, opens the Preventive Maintenance trigger record.

On success it opens the Preventive Maintenance Trigger record. On failure, the Preventive Maintenance trigger record is not selected and the flow fails.

Example:

```
preventiveMaintenance_searchAndSelectById(String strPMTrigId)
```

Input Parameters: String strPMTrigId

Return Type: void

preventiveMaintenance_AssetLocation_AddBtn

Opens the Asset Location page. On success, the Asset Location page will be opened. On failure, the page will not be opened.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_AssetLocation_AddBtn()
```

Input Parameters: NA

Return Type: void

preventiveMaintenance_AssetLocation_EnterAssetLoc

Enters the asset location based on the input string. On success, the asset location is successfully populated. On failure, the asset location is not populated.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_AssetLocation_EnterAssetLoc(String AssetLocId)
```

Input Parameters: String AssetLocId

Return Type: void

preventiveMaintenance_AssetLocation_AssetLoc_Save

Saves the asset location. On success, the asset location is saved successfully. On failure, it is not saved.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_AssetLocation_AssetLoc_Save()
```

Input Parameters: NA

Return Type: void

preventiveMaintenance_AssetLocation_AssetLoc_Cancel

Cancels the changes made to asset location. On success, the asset location is cancelled successfully. On failure, the changes are not cancelled.

The Preventive Maintenance Trigger record should be available.

Example

```
preventiveMaintenance_AssetLocation_AssetLoc_Cancel()
```

Input Parameters: NA

Return Type: void

preventiveMaintenance_AssetLocation_AssetLoc_Remove

Removes an asset location. On success, the asset location is removed successfully. On failure, the changes are not removed.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_AssetLocation_AssetLoc_Remove(String strRow,  
String strColumn)
```

Input Parameters: String strRow, String strColumn)

Return Type: void

preventiveMaintenance_AssetLocation_Previous

Shows up the previous asset location list. On success, the previous asset location will be shown. On failure, the it is not shown.

The Preventive Maintenance Trigger record and asset location list should be available.

Example:

```
preventiveMaintenance_AssetLocation_Previous()
```

Input Parameters: NA

Return Type: void

preventiveMaintenance_AssetLocation_Next

Shows up the next asset location list. On success, the next asset location will be shown. On failure, it will not be shown.

The Preventive Maintenance Trigger record and asset location list should be available.

Example:

```
preventiveMaintenance_AssetLocation_Next()
```

Input Parameters: NA
Return Type: void

preventiveMaintenance_Activity_Previous

Shows up previous activities on preventive maintenance. On success, the previous activities are shown. On failure, the previous activities will not be shown.

The Preventive Maintenance Trigger record and asset location list should be available.

Example:

```
preventiveMaintenance_Activity_Previous()
```

Input Parameters: NA
Return Type: void

preventiveMaintenance_Activity_Next

Shows up next activities on preventive maintenance. On success, the next activities will be shown. On failure, they will not be shown.

The Preventive Maintenance Trigger record and activities list should be available.

Example:

```
preventiveMaintenance_Activity_Next()
```

Input Parameters: NA
Return Type: void

preventiveMaintenance_FailureCause

Selects the failure cause from the failure cause list based on the input provided. On success, the failure cause is successfully selected. On failure, it is not successfully selected.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_FailureCause(String failureCause)
```

Input Parameters: String failureCause
Return Type: void

preventiveMaintenance_FailureMode

Selects the failure mode from the failure mode list based on the input provided. On success, the failure mode is successfully selected. On failure, it is not selected.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_FailureMode(String failureMode)
```

Input Parameters: String failureMode
Return Type: void

preventiveMaintenance_FailureRepair

Selects the failure repair from the failure repair list based on the input provided. On success, the failure repair is successfully selected. On failure, it is not selected.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_FailureRepair(String failureRepair)
```

Input Parameters: String failureRepair

Return Type: void

preventiveMaintenance_FailureComponent

Selects the failure component from the failure component list based on the input provided. On success, the failure component is successfully selected. On failure, it is not selected.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_FailureComponent(String failureComponent)
```

Input Parameters: String failureComponent

Return Type: void

preventiveMaintenance_FailureInfoAdd

Saves the failure information. On success, the failure information is saved. On failure, it is not saved.

The Preventive Maintenance Trigger record should be available and failure information should be selected.

Example:

```
preventiveMaintenance_FailureInfoAdd()
```

Input Parameters: NA

Return Type: void

preventiveMaintenance_OpenFailureCause

Opens the Failure Cause from the Planned Failure Information zone. On success, the failure cause successfully opens. On failure, it does not open.

The Preventive Maintenance Trigger record and failure information should be available.

Example:

```
preventiveMaintenance_OpenFailureCause(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

preventiveMaintenance_OpenFailureMode

Opens the Failure Mode from the Planned Failure Information zone. On success, the failure Mode successfully opens. On failure, it does not open.

The Preventive Maintenance Trigger record and failure information should be available.

Example:

```
preventiveMaintenance_OpenFailureMode(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

preventiveMaintenance_OpenFailureRepair

Opens the Failure Repair from the Planned Failure Information zone. On success, the Failure Repair successfully opens. On failure, it does not open.

The Preventive Maintenance Trigger record and failure information should be available.

Example:

```
preventiveMaintenance_OpenFailureRepair(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

preventiveMaintenance_OpenFailureComponent

Opens Failure Component from the Planned Failure Information zone. On success, the failure Component successfully opens. On failure, it does not open.

The Preventive Maintenance Trigger record should be available and Failure information should be available.

Example:

```
preventiveMaintenance_OpenFailureComponent(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

preventiveMaintenance_PlannedFailureInfoDelete

Deletes the planned failure information depending on the inputs. On success, the planned failure information is successfully deleted. On failure, it is not deleted.

The Preventive Maintenance Trigger record and failure information should be available.

Example:

```
preventiveMaintenance_PlannedFailureInfoDelete(String strRow, String strColumn)
```

Input Parameters: String strRow, String strColumn

Return Type: void

preventiveMaintenance_RetMaintTriggerFailureInfoCountFilter

Opens the Retrieve Maintenance Trigger failure information count. On success, the information is successfully visible. On failure, the information is not visible.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_RetMaintTriggerFailureInfoCountFilter()
```

Input Parameters: NA

Return Type: void

preventiveMaintenance_SetRetMaintTriggerFailureInfoCount

Sets the Retrieve Maintenance Trigger Failure information count based on the inputs. On success, the information count is set successfully. On failure, it is not set.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_SetRetMaintTriggerFailureInfoCount(String
lbl1,String lbl2,String lbl3,String lbl4,String lbl5)
```

Input Parameters: String lbl1,String lbl2,String lbl3,String
lbl4,String lbl5

Return Type: void

preventiveMaintenance_RetMaintTriggerFailureInfoCountSave

Save the Retrieve Maintenance Trigger Failure information count record. On success, Retrieve Maintenance Trigger Failure information count is successfully saved. On failure, it is not successfully saved.

The Preventive Maintenance Trigger record should be available.

Example:

```
preventiveMaintenance_RetMaintTriggerFailureInfoCountSave()
```

Input Parameters: NA

Return Type: void

Purchase Order

purchaseOrder_searchAndSelectById

Searches for the Purchase Order record based on the ID provided. If found, opens the Purchase Order record.

Example:

```
purchaseOrder_searchAndSelectById(String strPOId)
```

Input Parameters: String strPOId

Return Type: void

purchaseOrder_CreateRevisionClick

Allows the user to click the **Create Revision** button on the purchase Order header page.

This method should be called only after the purchaseOrder_searchAndSelectById method is called.

Example:

```
purchaseOrder_CreateRevisionClick()
```

Input Parameters: NA

Return Type: void

purchaseOrder_ForceComplete

Allows the user to click the **Force Complete** button on the Purchase Order header page.

This method should be called only after the purchaseOrder_searchAndSelectById method is called.

Example:

```
purchaseOrder_ForceComplete()
```

Input Parameters: NA

Return Type: void

purchaseOrder_ForceCompleteConfirm

Allows the user to click the **OK** button on the **Force Complete** confirmation dialogue box.

This method should be called only after the purchaseOrder_ForceComplete method is called.

Example:

```
purchaseOrder_ForceCompleteConfirm()
```

Input Parameters: NA

Return Type: void

purchaseOrder_ForceCompleteCancel

Allows the user to click **Cancel** button on the **Force Complete** confirmation dialogue box.

This method should be called only after the purchaseOrder_ForceComplete method is called.

Example:

```
purchaseOrder_ForceCompleteCancel()
```

Input Parameters: NA

Return Type: void

Financial Transaction**financialTransaction_searchAndSelectById**

Searches for the Financial Transaction record based on the ID provided. If found, opens the Financial Transaction record.

Example:

```
financialTransaction_searchAndSelectById(String strFTId)
```

Input Parameters: String strFTId

Return Type: void

financialTransaction_ClickEdit

Allows the user to click **Edit** button on the Financial Transaction page.

This method should be called only after the financialTransaction_searchAndSelectById method is called.

Example:

```
financialTransaction_ClickEdit()
```

Input Parameters: NA

Return Type: void

financialTransaction_ChangeAccDate

Allows the user to set the date in the Financial transaction maintenance UI to the value provided as input.

This method should be called only after the financialTransaction_ClickEdit method is called.

Example:

```
financialTransaction_ChangeAccDate(String date)
```

Input Parameters: String date

Return Type: void

financialTransaction_ChangeCostCenter

Allows the user to set the Cost Center associated to each row in the financial transaction maintenance UI to the value provided as input value, row number, and column number.

This method should be called only after the `financialTransaction_ClickEdit` method is called.

Example:

```
financialTransaction_ChangeCostCenter (String strRow, String
strColumn, String costcenter)
```

Input Parameters: String strRow, String strColumn, String costcenter
Return Type: void

financialTransaction_ChangeExpenseCode

Allows the user to set the Expense Code associated to each row in the financial transaction maintenance UI to the value provided as input value, row number and column number.

This method should be called only after the `financialTransaction_ClickEdit` method is called.

Example:

```
financialTransaction_ChangeExpenseCode (String strRow, String
strColumn, String expenseCode)
```

Input Parameters: String strRow, String strColumn, String expenseCode)
Return Type: void

financialTransaction_SaveEditedChanges

Allows the user to click the **Save** button in the financial transaction maintenance UI.

This method should be called only after the `financialTransaction_ClickEdit` method is called.

Example:

```
financialTransaction_SaveEditedChanges ()
```

Input Parameters: NA
Return Type: void

financialTransaction_CancelEditedChanges

Allows the user to click the **Cancel** button in the financial transaction maintenance UI.

This method should be called only after the `financialTransaction_ClickEdit` method is called.

Example:

```
financialTransaction_CancelEditedChanges ()
```

Input Parameters: NA
Return Type: void

Scheduling**Scheduler_searchAndSelectById**

Searches for the Crew shift based on the ID provided. If found, opens the Crew Shift record.

On success it opens the Crew Shift record. On failure, it is not selected and the flow fails.

Example:

```
Scheduler_searchAndSelectById (String strCrewShiftId)
```

Input Parameters: String strCrewShiftId
Return Type: void

Scheduler_ClickSchedule

Opens the Schedule record in Crew Shift record. On success, the schedule of the crew shift opens successfully. On failure, the shift does not open.

The Crew Shift record should be available.

Example:

```
Scheduler_ClickSchedule()
```

Input Parameters: NA
Return Type: void

Scheduler_LaborReqSearch_ByBacklogWork_Select

Opens the Backlog work selection of the Labor requirements search. On success, the Backlog work is opened for editing. On failure, it is not.

The Crew Shift record and Schedule record should be available.

Example:

```
Scheduler_LaborReqSearch_ByBacklogWork_Select()
```

Input Parameters: NA
Return Type: void

Scheduler_LaborReqSearch_BacklogWork_Craft

Sets the craft in Backlog work. On success, the craft is successfully set. On failure, it is not set.

The Crew Shift record and Backlog work is selected in the Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_BacklogWork_Craft(String craftCode)
```

Input Parameters: String craftCode
Return Type: void

Scheduler_LaborReqSearch_FirstWorkPriority

Sets the first work priority in Backlog work. On success, the first work priority is successfully set. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_FirstWorkPriority(String workPriority)
```

Input Parameters: String workPriority
Return Type: void

Scheduler_LaborReqSearch_SecondWorkPriority

Sets the second work priority in Backlog work. On success, the second work priority is successfully set. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_SecondWorkPriority(String workPriority)
```

Input Parameters: String workPriority

Return Type: void

Scheduler_LaborReqSearch_Search

Sets the backlog group in Backlog work. On success, the backlog group is successfully set. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_Search(String searchString)
```

Input Parameters: String searchString

Return Type: void

Scheduler_LaborReqSearch_ServiceClass

Sets the service class in Backlog work. On success, the service class is successfully set. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_ServiceClass(String serviceClass)
```

Input Parameters: String serviceClass

Return Type: void

Scheduler_LaborReqSearch_CheckOutageOnly

Set the **Check outage only** check box in the Backlog work. On success, the check box is set successfully. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_CheckOutageOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborReqSearch_OutageType

Sets the outage type in Backlog work. On success, the outage type is successfully set. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_OutageType(String outageType)
```

Input Parameters: String outageType

Return Type: void

Scheduler_LaborReqSearch_CheckUnscheduledOnly

Sets the **Check unscheduled only** check box in Backlog work. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record should be available and Backlog work is selected in Labor requirements search.

Example:

```
Scheduler_LaborReqSearch_CheckUnscheduledOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborReqSearch_Search

Searches for the labor requirements. On success, the labor requirements record is displayed. On failure, the record is not displayed.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborReqSearch_Search()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborReqSearch_Schedule_CrewShiftId

Selects the Crew Shift ID based on the input provided. On success, the Crew Shift ID is successfully selected. On failure, it is not selected.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborReqSearch_Schedule_CrewShiftId(String crewShiftId)
```

Input Parameters: String crewShiftId

Return Type: void

Scheduler_LaborReqSearch_Schedule_ClickUnscheduledOnly

Sets the **Check unscheduled only** check box in Backlog work. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborReqSearch_Schedule_ClickUnscheduledOnly(String crewShiftId)
```

Input Parameters: String crewShiftId

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_CrewId

Select the Crew ID on the **Work order info** drop down in the labor requirements search record. On success, the Crew ID is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info is selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_CrewId(String CrewId)
```

Input Parameters: String CrewId

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_WorkType

Select the work type on the **Work order info** drop down in the labor requirements search record. On success, the work type is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_WorkType(String WorkType)
```

Input Parameters: String WorkType

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_FirstWorkPriority

Sets the first work priority in Work Order Info. On success, the first work priority is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_FirstWorkPriority(String  
workPriority)
```

Input Parameters: String workPriority

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_SecondWorkPriority

Sets the second work priority in Work Order Info. On success, the second work priority is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_SecondWorkPriority(String  
workPriority)
```

Input Parameters: String workPriority

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_WorkClass

Sets the Work Class in Work Order Info. On success, the Work Class is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_WorkClass(String workClass)
```

Input Parameters: String workClass

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_WorkCategory

Sets the Work category in Work Order Info. On success, the Work category is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_WorkCategory(String
workCategory)
```

Input Parameters: String workCategory

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_Description

Sets the description in Work order info. On success, the description is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_Description(String description)
```

Input Parameters: String description

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_WorkOrder

Sets the Work order in Work order info. On success, the Work order is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_WorkOrder(String workOrderId)
```

Input Parameters: String workOrderId

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_ClickOutageOnly

Sets the **Check outage** only check box in Work order info. On success, the Work order info is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_ClickOutageOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_OutageType

Sets the outage type in work order info. On success, the outage type is successfully set. On failure, it is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_OutageType(String outageType)
```

Input Parameters: String outageType

Return Type: void

Scheduler_LaborReqSearch_WorkOrderInfo_ClickUnscheduledOnly

Sets the **Check unscheduled only** check box in work order info. On success, the check box is set successfully. On failure, the check box is not set.

The Crew Shift record should be available and work order info should be selected in the labor requirements search.

Example:

```
Scheduler_LaborReqSearch_WorkOrderInfo_ClickUnscheduledOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborReqSearch_ByActicityInfo

Select the activity information from the **Labor Requirements Search** drop down. On success, the activity information record is displayed. On failure, the information is not displayed.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborReqSearch_ByActicityInfo()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByActivityInfo_setCraft

Sets the craft in Search by activity info zone. On success, the craft is set successfully. On failure, it is not set.

The Crew Shift record should be available and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_setCraft(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_selectWorkPriorityFrom

Sets the select work priority from in Search by activity info zone. On success, the select work priority from is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectWorkPriorityFrom(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_selectWorkPriorityTo

Sets the select work priority to in Search by activity info zone. On success, the 'select work priority to' is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectWorkPriorityTo(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_selectServiceArea

Sets the select service area in Search by activity info zone. On success, the service area is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectServiceArea(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_setLocationID

Sets the location ID in Search by activity info zone. On success, the location ID is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_setLocationID(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_selectActivityType

Sets the activity type in Search by activity info zone. On success, the activity type is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectActivityType(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_selectActivityStatus

Sets the activity status in Search by activity info zone. On success, the activity status is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectActivityStatus(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_selectCostCenter

Sets the cost center in Search by activity info zone. On success, the cost center is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectCostCenter (String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByActivityInfo_setActivityId

Set the activity ID in Search by activity info zone. On success, the activity ID is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_setActivityId (String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByActivityInfo_selectBackLogGrp

Sets the backlog group in Search by activity info zone. On success, the backlog group is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectBackLogGrp (String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByActivityInfo_selectServiceClass

Sets the service class in Search by activity info zone. On success, the service class is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectServiceClass (String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByActivityInfo_selectHeldforParts

Sets the select held for parts in Search by activity info zone. On success, the select held for parts is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectHeldforParts (String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByActivityInfo_checkOutageOnly

Sets the **Check outage only** check box in the Search by activity info. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_checkOutageOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByActivityInfo_selectOutageType

Sets the outage type in activity info. On success, the outage type is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectOutageType(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByActivityInfo_CheckUnscheduledOnly

Sets the **Check unscheduled only** check box in activity info. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_CheckUnscheduledOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByActivityInfo_selectOwningOrg

Selects the owning organization. On success, the owning organization is successfully selected. On failure, it is not selected.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_SearchByActivityInfo_selectOwningOrg(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_LaborRequirementSearch_clickSearch

Searches for the labor requirements record. On success the labor requirements will be displayed. On failure, they are not displayed.

The Crew Shift record and Search by activity zone should be available.

Example:

```
Scheduler_LaborRequirementSearch_clickSearch()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborRequirementSearchResults_SlectAllCraft

Selects all craft in labor requirement search results. On success, all the crafts will be selected. On failure, they are not selected.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborRequirementSearchResults_SlectAllCraft()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborRequirementSearchResults_SelectSingleCraft

Selects the single craft in labor requirement search results. On success, single craft will be selected. On failure, it is not selected.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborRequirementSearchResults_SelectSingleCraft(String strRow)
```

Input Parameters: String strRow

Return Type: void

Scheduler_LaborRequirementSearchResults_SelectMultipleCrafts

Enables to select multiple crafts.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborRequirementSearchResults_SelectMultipleCrafts(String strRow1, String strRow2, String strRow3, String strRow4, String strRow5, String strRow6)
```

Input Parameters: String strRow1, String strRow2, String strRow3, String strRow4, String strRow5, String strRow6

Return Type: void

Scheduler_LaborRequirementSearchResults_clickApply

Displays all the search results.

Example:

```
Scheduler_LaborRequirementSearchResults_clickApply()
```

Input Parameters: NA

Return Type: void

Scheduler_LaborReqSearch_ByLocationInfo

Displays the location information. On success, the location information record is displayed. On failure, the information record is not displayed.

The Crew Shift record should be available.

Example:

```
Scheduler_LaborReqSearch_ByLocationInfo()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByLocationInfo_SelectLocType

Selects the location type in the location information. On success, the location type is selected successfully. On failure, it is not selected.

The Crew Shift record and Location Information zone should be available.

Example:

```
Scheduler_SearchByLocationInfo_SelectLocType(String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByLocationInfo_setAddress

Set the address in the location information. On success, the address is set successfully. On failure, the address is not set.

The Crew Shift record and Location Information zone should be available.

Example:

```
Scheduler_SearchByLocationInfo_setAddress(String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByLocationInfo_setCity

Set the city in the location information. On success, the city is set successfully. On failure, it is not set.

The Crew Shift record should be available and Location Information zone is displayed.

Example:

```
Scheduler_SearchByLocationInfo_setCity(String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByLocationInfo_setPostal

Sets the postal in the location information. On success, the postal is set successfully. On failure, it is not set.

The Crew Shift record and Location Information zone should be available.

Example:

```
Scheduler_SearchByLocationInfo_setPostal(String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByLocationInfo_setDescription

Set the description in the location information. On success, the description is set successfully. On failure, it is not set.

The Crew Shift record and Location Information zone should be available.

Example:

```
Scheduler_SearchByLocationInfo_setDescription(String strValue)
```

```
Input Parameters: String strValue
```

```
Return Type: void
```

Scheduler_SearchByLocationInfo_setCraft

Set the craft in the location information. On success, the craft is set successfully. On failure, the it is not set.

The Crew Shift record and Location Information zone should be available.

Example:

```
Scheduler_SearchByLocationInfo_setCraft (String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLocationInfo_selectServiceArea

Select the service area in the location information. On success, the service area is set successfully. On failure, it is not set.

The Crew Shift record should be available and Location information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_selectServiceArea (String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLocationInfo_selectBackLogGrp

Selects the backlog group in the location information. On success, the backlog group is set successfully. On failure, it is not set.

The Crew Shift record should be available and Location information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_selectBackLogGrp (String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLocationInfo_setLocationID

Select the location ID in the location information. On success, the location ID is set successfully. On failure, it is not set.

The Crew Shift record should be available and Location Information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_setLocationID (String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLocationInfo_checkOutageOnly

Sets the **Check outage only** check box in the Search by Location Info. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record should be available and Location information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_checkOutageOnly ()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByLocationInfo_selectOutageType

Sets the outage type in location info. On success, the outage type is successfully set. On failure, it is not set.

The Crew Shift record should be available and Location Information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_selectOutageType(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLocationInfo_CheckUnscheduledOnly

Sets the **Check unscheduled only** check box in location info. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record should be available and Location information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_CheckUnscheduledOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByLocationInfo_selectOwningOrg

This function selects the owning organization. On success, the owning organization is successfully selected. On failure, it is not selected.

The Crew Shift record should be available and Location information zone should be displayed.

Example:

```
Scheduler_SearchByLocationInfo_selectOwningOrg(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLaborReqInfo_setCraft

Sets the craft in the labor requirement information. On success, the craft is set successfully. On failure, it is not set.

The Crew Shift record should be available and labor requirement information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_setCraft(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLaborReqInfo_setDescription

Set the description in the labor requirement information. On success, the description is set successfully. On failure, it is not set.

The Crew Shift record should be available and labor requirement information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_setDescription(String strValue)
```

Input Parameters: String strValue
Return Type: void

Scheduler_SearchByLaborReqInfo_selectServiceArea

Select the service area in the labor requirement information. On success, the service area is set successfully. On failure, it is not set.

The Crew Shift record should be available and labor requirement information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_selectServiceArea(String strValue)
```

Input Parameters: String strValue
Return Type: void

Scheduler_SearchByLaborReqInfo_setLaborReqID

Sets the labor requirement ID.

The Crew Shift record should be available and labor requirement information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_setLaborReqID(String strValue)
```

Input Parameters: String strValue
Return Type: void

Scheduler_SearchByLaborReqInfo_selectServiceClass

Sets the service class in labor requirement information zone. On success, the service class is successfully set. On failure, it is not set.

The Crew Shift record should be available and labor requirement information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_selectServiceClass(String strValue)
```

Input Parameters: String strValue
Return Type: void

Scheduler_SearchByLaborReqInfo_selectHeldforParts

Sets the Select Held for Parts in the Labor Requirement Information zone. On success, the Select Held for Parts is successfully set. On failure, it is not set.

The Crew Shift record should be available and labor requirement information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_selectHeldforParts(String strValue)
```

Input Parameters: String strValue
Return Type: void

Scheduler_SearchByLaborReqInfo_checkOutageOnly

Sets the **Check outage only** check box in the Labor Requirement information. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record should be available and Labor Requirement Information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_checkOutageOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByLaborReqInfo_selectOutageType

Sets the outage type in the Labor Requirement Information zone. On success, the outage type is successfully set. On failure, it is not set.

The Crew Shift record should be available and Labor Requirement Information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_selectOutageType(String strValue)
```

Input Parameters: String strValue

Return Type: void

Scheduler_SearchByLaborReqInfo_CheckUnscheduledOnly

Sets the **Check unscheduled only** check box in labor requirement information. On success, the check box is successfully set. On failure, it is not set.

The Crew Shift record should be available and Labor Requirement Information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_CheckUnscheduledOnly()
```

Input Parameters: NA

Return Type: void

Scheduler_SearchByLaborReqInfo_selectOwningOrg

Selects the owning organization. On success, the owning organization is successfully selected. On failure, it is not selected.

The Crew Shift record should be available and Labor Requirement Information zone should be displayed.

Example:

```
Scheduler_SearchByLaborReqInfo_selectOwningOrg(String strValue)
```

Input Parameters: String strValue

Return Type: void

Material Disposition

MaterialDisposition_searchAndSelectByActivityId

Allows to search and select the material disposition record based on the activity ID provided as input.

Example:

```
MaterialDisposition_searchAndSelectByActivityId(String strActivityId)
```

Input Parameters: String strActivityId

Return Type: void

MaterialDisposition_CompleteExchange

Allows the user to click **Complete Exchange** on the **Material Disposition** page.

This method should be called only after the MaterialDisposition_searchAndSelectByActivityId method is called.

Example:

```
MaterialDisposition_CompleteExchange ()
```

Input Parameters: NA

Return Type: void

MaterialDisposition_ReturnAsset

Allows the user to click **Return** on the specified row in the **Assets** zone on the **Material Disposition** page.

This method should be called only after the MaterialDisposition_searchAndSelectByActivityId method is called.

Example:

```
MaterialDisposition_ReturnAsset(String strRow)
```

Input Parameters: String strRow

Return Type: void

MaterialDisposition_MaterialReturn_slectStoreRoom

Allows the user to select the store room from **Storage Location** drop down in the **Material Return** page based on the value provided as input.

This method should be called only after the MaterialDisposition_ReturnAsset method is called.

Example:

```
MaterialDisposition_MaterialReturn_slectStoreRoom (String strValue)
```

Input Parameters: String strValue

Return Type: void

MaterialDisposition_MaterialReturn_setCostCenter

Allows the user to set the cost center in the **Material Return** page based on the value provided as input.

This method should be called only after the MaterialDisposition_ReturnAsset method is called.

Example:

```
MaterialDisposition_MaterialReturn_setCostCenter (String strValue)
```

Input Parameters: String strValue
Return Type: void

MaterialDisposition_MaterialReturn_setPercentage

Allows the user to set the cost center in the **Material Return** page based on the value provided as input.

This method should be called only after the `MaterialDisposition_ReturnAsset` method is called.

Example:

```
MaterialDisposition_MaterialReturn_setPercentage (String strValue)
```

Input Parameters: String strValue
Return Type: void

MaterialDisposition_MaterialReturn_setProjectId

Allows the user to set the project Id in the **Material Return** page based on the value provided as input.

This method should be called only after the `MaterialDisposition_ReturnAsset` method is called.

Example:

```
MaterialDisposition_MaterialReturn_setProjectId (String strValue)
```

Input Parameters: String strValue
Return Type: void

MaterialDisposition_MaterialReturn_ClickOnSave

Allows the user to click **Save** in the **Material Return** page.

This method should be called only after the `MaterialDisposition_ReturnAsset` method is called.

Example:

```
MaterialDisposition_MaterialReturn_ClickOnSave ()
```

Input Parameters: NA
Return Type: void

MaterialDisposition_MaterialReturn_ClickOnCancel

Allows the user to click **Cancel** in the **Material Return** page.

This method should be called only after the `MaterialDisposition_ReturnAsset` method is called.

Example:

```
MaterialDisposition_MaterialReturn_ClickOnCancel ()
```

Input Parameters: NA
Return Type: void

Material Return

MaterialReturn_openReturnLines_navigateTo

Allows the user to click **Open Material Return** line info string in the **Open Return Lines** zone of the **Material Return Header** based on the row number provided.

This method should be called only after the `MaterialDisposition_MaterialReturn_ClickOnSave` method is called.

Example:

```
MaterialReturn_openReturnLines_navigateTo(String strRow)
```

Input Parameters: String strRow

Return Type: void

MaterialReturnLine_VerifyRepairedFlag

Allows the user to verify the repaired filed value on the **Material Return Line** page is same as the value that is provided as input.

This method should be called only after the MaterialReturn_openReturnLines_navigateTo method is called.

Example:

```
MaterialReturnLine_VerifyRepairedFlag(String strExpectedValue)
```

Input Parameters: String strExpectedValue

Return Type: void

Chapter 3

Sample Work Flows

This chapter describes the Oracle Utilities Work and Asset Management sample flows that illustrate common use cases for Oracle Utilities Work and Asset Management. It also explains the procedure to execute these sample flows.

The chapter includes the following sections:

- [Sample Flows](#)
- [Executing Sample Flows](#)

Sample Flows

The sample flows delivered as part of Oracle Functional Testing Advanced Pack for Oracle Utilities Oracle Utilities Work and Asset Management demonstrate how flows can be created for Web services based testing and for a combination of Web services and UI based testing using the same framework.

These flows are designed to run using the demo data, there by giving the user, the ability to deploy Oracle Functional Testing Advanced Pack for Oracle Utilities for Oracle Utilities Work and Asset Management and execute the sanity flows immediately. The flows perform a part of the basic sanity testing required to certify that the Oracle Utilities Work and Asset Management environment has been setup appropriately.

This section includes the following sample work flows:

- [W1-ReorderReview_Inventory Flow](#)
- [W1-ReplicationOfAssets-ODM Flow](#)
- [W1-Material Disposition Flow](#)
- [W1-Work Demand Flow](#)

W1-ReorderReview_Inventory Flow

The W1-ReorderReview_Inventory flow includes the creation and completion life cycle of reorder review and inventory adjustment for inventory.

The following table lists the tasks created and their respective components in Oracle Utilities Work and Asset Management.

Task	WAM Component
Stock Item (Add/Update)	W1-StockItem
Store room Inventory Management (Add)	W1-StoreroomInventorymgmt

Task	WAM Component
Store room Inventory Management (Add)	W1-StoreroomInventorymgmt
Vendor (Add)	W1-Vendor
Vendor Location (Add)	W1-VendorLocation
SID Inventory (Add/Update)	W1-SIDInventory
Regular Inventory Adjustment Price/Quantity (Add)	W1-InvAdjRegularInventory

Inventory adjustment on SID Inventory brings down the inventory count below the reorder point. Execute the W1-SID batch using the Oracle Utilities Application Framework (OUAF) component F1-BatchJobSubmission to generate the Reorder Review record.

Approve the reorder review record using the W1-ReorderReview component. Execute the W1-BSRR batch using the Core component F1-BatchSubmission to process the reorder review record further to create a PR/PO as per user configuration. Read the Reorder Review Log using the W1-ReadLogEntry component to verify the correct document PR/PO is created.

The test execution report is sent to the configured email IDs using the Core component F1-SendResultsMail.

W1-ReplicationOfAssets-ODM Flow

The W1-ReplicationOfAssets-ODM flow includes the creation of an asset and uses it as a template asset to replicate multiple assets using the replication activity.

The following table lists the tasks created and their respective components in Oracle Utilities Work and Asset Management.

Task	WAM Component
Storage Location (Add)	W1-StorageLocation
Smart Meter(Add)	W2-SmartMeter
Basic Work Order (Add)	W1-BasicWorkOrder (Add)
Asset Replication Activity (Add/Update)	W1-AssetReplicationActivity

Create a smart meter asset in Inreceipt disposition on a out-of-service storage location. Create a basic work order and use it and smart meter asset as temple to create the asset replication activity. Complete the replication activity to replicate the required number of assets.

The test execution report is sent to the configured email IDs using the Core component F1-SendResultsMail.

W1-Material Disposition Flow

The Material Disposition flow includes the creation and completion life cycle of work request, work order, activity along with the crew shift, and material request.

The following table lists the tasks created and their respective components in Oracle Utilities Work and Asset Management.

Task	WAM Component
Stock Item (Add/Update)	W1-StockItem
Store room Inventory Management (Add)	W1-StoreroomInventorymgmt
SID Inventory (Add/Update)	W1-SIDInventory
General Asset Location (Add)	W1-GeneralAssetLocation
Work Request (Add/Update)	W1-WorkRequest
Work Order (Add/Update)	W1-WorkOrder
Tracked General Specification (Add)	W1-TrackedGeneralSepec
Tracked General Asset (Add/Update)	W1-TrackedGeneralAsset
Local Activity Core WAM (Add/Update)	W1-LocalActivityCoreWAM
Regular Inventory Adjustment Price/Quantity (Add)	W1-InvAdjRegularInventory
Activity Material Requirement (Add)	W1-ActivityMaterialRequirement
Activity Labor Requirement (Add)	W1-ActivityLaborRequirement
Employee (Add)	W1-Employee
Craft (Add)	W1-Craft
Crew with Craft (Add)	W1-CrewWithCraft
Daily Schedule (Add)	W1-DailySchedule
Material request header (Add/Update)	W1-MaterialRequestHeader
Activity Material Requirement (Add)	W1-ActivityMaterialRequirement

Work request is created with association of the Location and Approval profile. Once the Work request is approved a Work Order is created with attributes from Work request. Add the activity and associate with the Work Order. Once the activity is associated and the resource planning is done, work order should be approved and crew shift will be assigned to it.

When the Material is issued using the component W1-MaterialRequestLineIssueAll, from the Storeroom for the activity, a material disposition record will be created. The faulty asset is replaced with the new asset issued from the storeroom using the component W1-MaterialDispCompleteExchange and the repair Work order will be created for the faulty asset. Test execution report will be sent to the configured email IDs using the Core component F1-SendResultsMail.

W1-Work Demand Flow

The work demand flow includes the creation and completion life cycle of work request, work order, and activity, along with the stock transfer.

The following table lists the tasks created and their respective components in Oracle Utilities Work and Asset Management.

Task	WAM Component
Stock Item (Add/Update)	W1-StockItem
Store room Inventory Management (Add)	W1-StoreroomInventorymgmt
SID Inventory (Add/Update)	W1-SIDInventory
General Asset Location (Add)	W1-GeneralAssetLocation
Work Request (Add/Update)	W1-WorkRequest
Work Order (Add/Update)	W1-WorkOrder
Tracked General Specification (Add)	W1-TrackedGeneralSepec
Tracked General Asset (Add/Update)	W1-TrackedGeneralAsset
Local Activity Core WAM (Add/Update)	W1-LocalActivityCoreWAM
Regular Inventory Adjustment Price/Quantity (Add)	W1-InvAdjRegularInventory
Activity Material Requirement (Add)	W1-ActivityMaterialRequirement
Activity Labor Requirement (Add)	W1-ActivityLaborRequirement
Employee (Add)	W1-Employee
Craft (Add)	W1-Craft
Crew with Craft (Add)	W1-CrewWithCraft
Daily Schedule (Add)	W1-DailySchedule
Material request header (Add/Update)	W1-MaterialRequestHeader
Activity Material Requirement (Add)	W1-ActivityMaterialRequirement
Stock Transfer header (Add/Update)	W1-StockTransferHeader

The work request is created with association of the Assets and Approval profile. Once the Work request is approved a Work Order is created with attributes from Work request. Add the activity and associate with the Work Order. Once the activity is associated and the resource planning is done, work order should be approved and demand will be placed against the Storeroom.

The storeroom doesn't have enough resources so the demand will be placed on another storeroom through Stock Transfer process. Stock transfer process transfer the stock items to the required storeroom using the component W1-StkTranOpenReqLineIssueAll and then the material will be issued from the storeroom to the crew for completion of activity.

Once the Activity is completed, timesheet will be created and completed for the labor and posted against Work Order/Activity after running the W1-TMST batch using the Oracle Utilities Application Framework component F1-BatchJobSubmission. Test execution report will be sent to the configured mail ids using the Core component F1-SendResultsMail.

Executing Sample Flows

This section describes the procedure to setup sample flows and execute them.

- [Pre-requisites](#)
- [Setting Up Sample Flows](#)

Pre-requisites

To execute the sample flow, ensure the following pre-requisites are met:

- Oracle Utilities Work and Asset Management v2.1.1.0 is up and running with the demo data pack.
- OpenScript is installed in the local machine. See the *Oracle Functional Testing Advanced Pack for Oracle Utilities Installation and Administration Guide* for the version details.
- Oracle Utilities Work and Asset Management is installed and repository/directory is setup in the local machine appropriately. See the *Oracle Functional Testing Advanced Pack for Oracle Utilities Installation and Administration Guide* for more details.

Setting Up Sample Flows

To setup a sample flow, follow these steps:

1. Login to Oracle Utilities Work and Asset Management.
2. Import the Inbound Web services into the Oracle Utilities Work and Asset Management application where the scenarios need to be executed.

See the **Importing Inbound Web Services** section in *Oracle Functional Testing Advanced Pack for Oracle Utilities User's Guide* for steps to import the Inbound Web services

3. Navigate to **Admin > B > Bundle Import > Add**.
4. Enter the **External Reference**, **Detailed Description**, and **Bundle Details** from the IWS Bundle Export Dump.
5. Click **Save**, and then click **Apply Bundle**.
6. Navigate to **Admin > I > Inbound Webservice Deployment**, and then click **Deploy**.
7. Launch OpenScript in the local machine and perform the following steps:
 - a. Navigate to **View > OpenScript Preferences**.
 - b. In the left tree, select **OpenScript**. In the sub tree, select **Playback**, and then select **Error Recovery**.
 - c. Click **SetAll** and select **Report Error and Continue**.
 - d. Click **Apply**, and then click **Close**.
7. Configure the **configuration.properties** file as follows:
 - a. Provide the application URL for the parameter:

```
gStrApplicationURL = http\://<%serverName%>\:<%portNumber%>/
ouaf
```

- b. Provide the additional path required for webservice/inbound service URL:

```
gStrApplicationXAIServerPath=/<%AppendThisToAbove
gStrApplicationURL %>/<%AppendThisToAbove gStrApplicationURL
%>/
```

- c. Provide an environment name for display in the results email:
`gStrEnvironmentName= <%testEnvironmentName%>`
- d. Provide the application login user ID:
`gStrApplicationUserName= <%UserName%>`
- e. Provide the application login password:
`gStrApplicationUserPassword= <%password%>`
- f. Provide the SMTP email server and e-mail ID:
`gStrSMTP_HOST_NAME=<%SMTP ServerName%>`
`gStrSMTP_PORT=<%PortNumber%>`
`gStrTO_EMAIL_RECIPIENTS=<%e-mail Id%>`
- g. Provide the application database details as below:
`gStrApplicationDBConnectionString =<%jdbc Connectionstring%>`
eg: `jdbc\:oracle\:thin\:@<%DBserverName%>\:<%port%>\:<%DBSID%>`
`gStrApplicationDBUsername=<%DBUserID%>`
`gStrApplicationDBPassword=<%DBPassword%>`
- h. Provide the full directory path of Oracle Application Testing Suite repository directories in the local machine.
`gStrOutputFilePath=<%LogFilePath%>`
Example: `C:\OATSOUWAM_DEMO\OUTSP\Log\`
`gStrXSDFiles=<%XSD Folder path%>`
Example: `C:\OATSOUWAM_DEMO\OUTSP\Log\`
9. Copy the following function libraries (scripts from the installer) into the **outsp-function-libs** folder in the Oracle Application Testing Suite repository directory.
- WAM
 - OUWAMUILIB
 - Core
 - OUTSPCORELIB
 - WSCOMMONLIB
 - WSVALIDATELIB
10. Copy all the .jar files provided in the installer into the **genericJars** folder in the Oracle Application Testing Suite repository directory.

Appendix A

Inbound Web Services

The Oracle Utilities Work and Asset Management components are developed using Web services method, and these components require Inbound Web Services to be defined in the application.

For instructions to create, import, or search an Inbound Web Service, see the **Setting Up Inbound Web Services** appendix in *Oracle Functional Testing Advanced Pack for Oracle Utilities User's Guide*.

List of Inbound Web Services

The list of Inbound Web Services provided to use with the delivered components and flows is as follows:

- ATW1AcceptLineCraft
- ATW1AcceptLineMaterial
- ATW1AccountingCalendar
- ATW1ActivityComplInboundComm
- ATW1ActivityCompletionEventType
- ATW1ActivityEquipRequirement
- ATW1ActivityLaborRequirement
- ATW1ActivityMaterialRequirement
- ATW1ActivityOtherRequirement
- ATW1AdHocActivityCore
- ATW1AmtBasedAppvlProf
- ATW1ApprovalRequest
- ATW1AssessmentClass
- ATW1AssessmentGroup
- ATW1AssetAgeBktConfig
- ATW1AssetCompCfgRpt
- ATW1AssetCompCfgWO
- ATW1AssetCompConfig
- ATW1AssetCompToAssetCfgRpt
- ATW1AssetCompToAssetCfgWO

- ATW1AssetCompToAssetCompCfgRpt
- ATW1AssetCompToAssetCompCfgWO
- ATW1AssetCompToAssetCompConfig
- ATW1AssetCompToAssetConfig
- ATW1AssetCompletionEventType
- ATW1AssetConditionScoreBuckets
- ATW1AssetCriteriaActGenerator
- ATW1AssetCriticalityBuckets
- ATW1AssetDocument
- ATW1AssetDposCmpEvtType
- ATW1AssetInstallAgeBktConfig
- ATW1AssetInstoreAgeBktConfig
- ATW1AssetListActGenerator
- ATW1AssetLoc
- ATW1AssetLocSpecTemplWorkOrder
- ATW1AssetLocationSpecificTemplateActivity
- ATW1AssetReplicationActivity
- ATW1AssetReplicationComplEvent
- ATW1AssetSearchMasterConfig
- ATW1AssetToAssetCfgRpt
- ATW1AssetToAssetCfgWO
- ATW1AssetToAssetCompCfgRpt
- ATW1AssetToAssetCompCfgWO
- ATW1AssetToAssetCompConfig
- ATW1AssetType
- ATW1AssetTypeAverageCost
- ATW1Assetloctype
- ATW1AssettoAssetconfig
- ATW1BCAccessList
- ATW1BCCraftLine
- ATW1BCEquipmentLine
- ATW1BCHeader
- ATW1BCHeaderIntegration
- ATW1BCOtherLine
- ATW1BCStockItemLine
- ATW1BIExtractParameters
- ATW1BOMGenStockItem
- ATW1BasicActivityType

- ATW1BasicTest
- ATW1BasicWorkOrder
- ATW1BillableRate
- ATW1Business
- ATW1BusinessUnit
- ATW1Buyer
- ATW1CalendarAnniversary
- ATW1CalendarInterval
- ATW1CalendarWarranty
- ATW1CalendarWarrantyTerm
- ATW1CapabilityType
- ATW1Circle
- ATW1Color
- ATW1CommodityCategory
- ATW1CommodityName
- ATW1CommodityType
- ATW1CommonShiftPlanTemplate
- ATW1CommunicationComponentSpec
- ATW1CommunicationRelaySpec
- ATW1CompnAssetListActGenerator
- ATW1CompnCriteriaActGenerator
- ATW1ComponentType
- ATW1ComponentTypeAverageCost
- ATW1ConfigType
- ATW1ContactType
- ATW1CostCategory
- ATW1CostCenter
- ATW1CostCenterExpenseCode
- ATW1Course
- ATW1Craft
- ATW1CraftPOLine
- ATW1CreActivityCmpEvt
- ATW1CreActivityPayCmpEvt
- ATW1CreAssetAttachCmpEvt
- ATW1CreAssetCmpEvt
- ATW1CreAssetDocumentCmpEvt
- ATW1CreAssetPayCmpEvt
- ATW1CreComActivityInboundComm

- ATW1CreLocationDocumentCmpEvt
- ATW1CreNodeAttachCmpEvt
- ATW1CreServiceHistoryAttCmpEvt
- ATW1CreServiceHistoryPayCmpEvt
- ATW1CreSvcHistCmpEvt
- ATW1CreWorkReqDocumentCmpEvt
- ATW1CreWorkRequestPayCmpEvt
- ATW1CreateActivityInboundComm
- ATW1Crew
- ATW1CrewShiftActivitySchedules
- ATW1CrewShiftTemplate
- ATW1CrewType
- ATW1CrewWithCraft
- ATW1DailySchedule
- ATW1DailyScheduleType
- ATW1DataAreaCmpEvtType
- ATW1DistributionCode
- ATW1DocAct
- ATW1DocAttachment
- ATW1DocCmplEvt
- ATW1DocCmplEvtType
- ATW1DocUrl
- ATW1DocumentType
- ATW1Employee
- ATW1EmployeeType
- ATW1EquipForTemplateAct
- ATW1Equipment
- ATW1EquipmentPOLine
- ATW1EquipmentPRLine
- ATW1Expedite
- ATW1ExpenseCode
- ATW1FOB
- ATW1Factor
- ATW1FactorValue
- ATW1FailureComponent
- ATW1FailureMode
- ATW1FailureProfile
- ATW1FailureProfileFailureComp

- ATW1FailureProfileFailureMode
- ATW1FailureProfileFailureRepair
- ATW1FailureProfileFailureType
- ATW1FailureRepair
- ATW1FailureServiceHistory
- ATW1FailureType
- ATW1FinancialMasterConfig
- ATW1FinancialTransaction
- ATW1FirmwareSpecification
- ATW1FirmwareVersion
- ATW1Function
- ATW1FunctionType
- ATW1GaugeReadWithThresholdMeasType
- ATW1GeneralAssetLocation
- ATW1GeneralAssetLocationType
- ATW1GeneralCommunicationType
- ATW1GeneralCompletionEventType
- ATW1GeneralTemplateActivity
- ATW1GeneralTemplateWorkOrder
- ATW1IOSvcFleetAsset
- ATW1IOSvcITAsset
- ATW1IOSvcITComponent
- ATW1IOSvcInfrastructureAsset
- ATW1InOutSvcAsset
- ATW1InOutSvcComponent
- ATW1InOutOfServiceGeneralAsset
- ATW1InOutOfServiceGeneralComponent
- ATW1IndependentPartBOM
- ATW1InventoryAdjustmentIT
- ATW1InventoryAdjustmentLM
- ATW1InventoryAdjustmentRI
- ATW1InventoryMgmtMasterConfig
- ATW1InvoiceHeader
- ATW1InvoiceLine
- ATW1LaborEarningType
- ATW1LaborForTemplateAct
- ATW1Leave
- ATW1LeaveType

- ATW1Line
- ATW1LinearAssetLocation
- ATW1LinearAssetLocationType
- ATW1LocalActivityCore
- ATW1LocalActivityCoreWAM
- ATW1LocationDocument
- ATW1MDMAsset
- ATW1MTBFMTTRBuckets
- ATW1MaintenanceSchedule
- ATW1ManualMeterSpecification
- ATW1Manufacturer
- ATW1MassUpd
- ATW1MaterialRequestHeader
- ATW1MaterialRequestLine
- ATW1MaterialRequestTemplate
- ATW1MaterialReturnHeader
- ATW1MeasQuickEntryActivityType
- ATW1MeasurementUOM
- ATW1MeterReadWithRollOverMeasType
- ATW1MigrPlan
- ATW1MigrTransactionImport
- ATW1MigrationAssistantConfig
- ATW1MsrmtQuickEntryActivity
- ATW1MsrmtQuickEntryCmplEvt
- ATW1MsrmtQuickEntryCmplEvtType
- ATW1MsrmtTrackedGeneralAsset
- ATW1MsrmtQuickEntryCmplEvtTyp
- ATW1NoAppvlRequiredProf
- ATW1NodeCriteriaActGenerator
- ATW1NonAmtBasedAppvlProf
- ATW1NumberOfCyclesOverdueBucket
- ATW1NumericGauge
- ATW1NumericMeter
- ATW1ODCEquipment
- ATW1OperationalReadingCmpEvt
- ATW1Organization
- ATW1OrganizationType
- ATW1Other

- ATW1OtherDirectCharges
- ATW1OtherDirectChargesOther
- ATW1OtherForTemplateAct
- ATW1OtherResourcePOLine
- ATW1OutboundActivityCore
- ATW1OutboundActivityType
- ATW1OvertimeType
- ATW1PIHeader
- ATW1POHeaderDocuments
- ATW1POHeaderIntegration
- ATW1POHeaderNotes
- ATW1POLineDocuments
- ATW1POLineIntegration
- ATW1POLineNotes
- ATW1PRCrLine
- ATW1PRCreBC
- ATW1PREqLine
- ATW1PRHeader
- ATW1PRORLine
- ATW1PRSILine
- ATW1PaymentTerm
- ATW1Person
- ATW1PersonWAM
- ATW1PhysicalInventoryCountLine
- ATW1PhysicalInventoryCountMaster
- ATW1PhysicalInventoryHeader
- ATW1Planner
- ATW1PlantAssetLocation
- ATW1PlantAssetLocationType
- ATW1Project
- ATW1ProjectCategory
- ATW1PurchaseCommodity
- ATW1PurchaseOrderHeader
- ATW1PurchaseRequisition
- ATW1PurchasingMasterConfig
- ATW1Question
- ATW1QuestionnaireSvcHist
- ATW1QuestionnaireSvcHistType

- ATW1RCMCalendarAnniversary
- ATW1RCMCalendarInterval
- ATW1RCMRuntimeAnniversary
- ATW1RCMRuntimeInterval
- ATW1RCMTrigPlndFailureInfoLITE
- ATW1RETCEVT
- ATW1ReadLogEntry
- ATW1ReadWithNoEvaluateMeasType
- ATW1ReceiptHeader
- ATW1ReceiptLine
- ATW1ReceiptLineCraft
- ATW1ReorderReview
- ATW1ReplicationComplEventTypes
- ATW1ResourceMgmtMasterConfig
- ATW1ResourceUOM
- ATW1RetrieveComplEvents
- ATW1ReturnLine
- ATW1RuntimeAnniversary
- ATW1RuntimeInterval
- ATW1RuntimeWarranty
- ATW1RuntimeWarrantyTerm
- ATW1SIDDirectPurchase
- ATW1SIDExpensedAtReceipt
- ATW1SIDInventory
- ATW1SIDSpecificDirectPurchase
- ATW1SIDSpecificLot
- ATW1SchedulePlan
- ATW1SchedulePlanType
- ATW1ScheduledActivityType
- ATW1ScheduledActivityTypeWAM
- ATW1ServHistFromActTypeCmplEvtType
- ATW1ServiceArea
- ATW1ServiceClass
- ATW1ServiceHistTypeCmpEvtType
- ATW1ServiceHistory
- ATW1ServiceHistoryType
- ATW1ServicePoint
- ATW1ShiftPlanTemplate

- ATW1SingleAssetCfgRpt
- ATW1SingleAssetCfgWO
- ATW1SingleAssetConfig
- ATW1SmartMeterSpecification
- ATW1SmartTest
- ATW1StandardNote
- ATW1StockItem
- ATW1StockItemPOLine
- ATW1StockTransaction
- ATW1StockTransferHeader
- ATW1StockTransferIssueLine
- ATW1StockTransferReceiptHeader
- ATW1StockTransferReceiptLine
- ATW1StockTransferRequestLine
- ATW1StorageLocation
- ATW1StorageLocationType
- ATW1StoreroomInventoryMgmt
- ATW1SubscriptionShiftPlanTemp
- ATW1SynchronizationAddContact
- ATW1TaxRateSchedule
- ATW1TemplateActivityLaborReq
- ATW1TemplateActivityMaterialReq
- ATW1TimePeriod
- ATW1Timesheet
- ATW1TimesheetDetail
- ATW1TrackedFleetSpecification
- ATW1TrackedGeneralAsset
- ATW1TrackedGeneralComponent
- ATW1TrackedGeneralSpecification
- ATW1TrackedITAsset
- ATW1TrackedITComponent
- ATW1TrackedITSpecification
- ATW1TrackedInfrastructureSpec
- ATW1TrnAssetCmpEvent
- ATW1UpdAssetPayCmpEvt
- ATW1UpdNodePayCmpEvt
- ATW1User
- ATW1Vehicle

- ATW1Vendor
- ATW1VendorLocation
- ATW1WAMBIExtractParameters
- ATW1WAssetAgeBuckets
- ATW1WAssetInstallAgeBuckets
- ATW1WorkCalendar
- ATW1WorkManagementMasterConfig
- ATW1WorkOrder
- ATW1WorkPriorityBuckets
- ATW1WorkReqComplEvtType
- ATW1WorkRequest
- ATW1WorkRequestComplEvt
- ATW2CommunicationComponent
- ATW2CommunicationRelay
- ATW2FirmwareVersion
- ATW2ManualMeter
- ATW2ManualMeterSpecification
- ATW2ServicePoint
- ATW2SmartMeter
- ATW2SmartMeterSpecification
- ATW1READASSETLOCID
- ATW1READCONTACT
- ATW1READSRVHSTID