

# Oracle® Financial Services

## Know Your Customer Utilities Guide



Release 8.1.2.10.0

G42130-02

August 2025

ORACLE®

Copyright © 1994, 2005, Oracle and/or its affiliates.

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, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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 about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

# Contents

## 1 About This Guide

---

1.1	Who Should Use This Guide	1
1.2	How this Guide is Organized	1
1.3	Where to Find More Information	1
1.4	Conventions Used in This Guide	2

## 2 Overview

---

2.1	Prerequisites	1
2.2	Table to JSON Utility	1
2.3	JSON to Table Utility	2

## 3 Tables and Table Elements

---

3.1	Table to JSON	1
3.1.1	fcc_tpg_table_json_mapping	1
3.1.2	fcc_tpg_table_json_query	3
3.1.3	fcc_tpg_table_json_query_params	4
3.2	JSON to Table	5
3.2.1	fcc_ob_json_table_config	5
3.2.2	fcc_ob_json_table_map	5

## 4 Viewing JSON

---

## 5 Mapping IDs and REST URLs for Utilities

---

## A OFSAA Support

---

## B Send Us Your Comments

---

# Document Control

**Table    Document Control**

Revision Number	Revision Date	Change Log
8.1.2.10.0	August 2025	There are no updates to this document in this release.
8.1.2.9.0	February 2025	There are no updates to this document in this release.
8.1.2.8.0	August 2024	There are no updates to this document in this release.
8.1.2.7.0	February 2024	There are no updates to this document in this release.
8.1.2.6.0	October 2023	There are no updates to this document in this release.
8.1.2.5.0	June 2023	There are no updates to this document in this release.
8.1.2.4.0	March 2023	There are no updates to this document in this release.
8.1.2.3.0	December 2022	There are no updates to this document in this release.
8.1.2.2.0	September 2022	There are no updates to this document in this release.
8.1.2.1.0	June 2022	A new version has been created for the 8.1.2.1.0 release.
8.1.2.0.0	March 2022	A new version has been created for the 8.1.2.0.0 release.
1.0	July 2021	Created the document.

# 1

## About This Guide

This guide details the table and table elements that form a part of the Table to JSON and JSON to Table utilities' web services which can be deployed on any web server.

### Topics:

- [Who Should Use This Guide](#)
- [How this Guide is Organized](#)
- [Where to Find More Information](#)
- [Conventions Used in This Guide](#)

## 1.1 Who Should Use This Guide

This document is intended for all those users who want to understand the utilities used in the Oracle Financial Services Know Your Customer (OFS KYC) onboarding service so that they can integrate with our service from their onboarding systems.

## 1.2 How this Guide is Organized

The OFS KYC Utilities Guide includes the following chapters:

- [Overview](#) provides information on the Table to JSON utility and JSON to Table utility.
- [Tables and Table Elements](#) provides information on the different tables used.
- [Viewing JSON](#) shows a sample JSON.
- [Mapping IDs and REST URLs for Utilities](#) provides information on the mapping IDs and REST URLs used.

## 1.3 Where to Find More Information

For more information about OFS KYC, see the following documents:

- Know Your Customer Administration Guide
- Know Your Customer Risk Assessment Guide
- Data Interface Specification (DIS) Guide
- Data Model Reference (DMR) Guide
- Service Guide
- API Data Elements Guide
- Enterprise Case Management User Guide.

These documents can be found at the following link: [Financial Crime and Compliance Management Behavior Detection Application Pack](#).

To find additional information about how OFS solves real business problems, see our website at [Financial Services Home Page](#).

## 1.4 Conventions Used in This Guide

The following table mentions the conventions used in this guide.

**Table 1-1 Conventions Used**

Conventions Used	Meaning
<i>Italics</i>	<ul style="list-style-type: none"><li>Names of books as references</li><li>Emphasis</li><li>Substitute input values</li></ul>
<b>Bold</b>	<ul style="list-style-type: none"><li>Menu names, field names, options, button names</li><li>Commands typed at a prompt</li><li>User input</li></ul>
Hyperlink	Hyperlink type indicates the links to external websites and internal document links to sections.
Asterisk (*)	Mandatory fields in User Interface
<Variable>	Substitute input value

# 2

## Overview

The following sections discuss the prerequisites and utilities used.

### Topics:

- [Prerequisites](#)
- [Table to JSON Utility](#)
- [JSON to Table Utility](#)

## 2.1 Prerequisites

JNDI data source configured for the ATOMIC schema. After you deploy the web service, replace the

##JNDI\_NAME## placeholder with the data source name in the following files:

- connection.properties file under the WEB-INF/classes directory
- Web.xml file under the WEB-INF/classes directory

## 2.2 Table to JSON Utility

The Table to JSON utility creates a JSON according to the configurations specified by a user. The user can define the JSON structure and the JSON fields to be constructed, and indicate where each field data is available. Once the field values are provided, the utility creates the JSON according to the specified configurations and picks the values against each field defined during configuration.

### Note

The utility can support only two levels of the array in the JSON. To see an example, see [Viewing JSON](#). Examples are also provided in the [Oracle Financial Services Know Your Customer Service Guide](#).

The Table to JSON utility is used in KYC to perform the following tasks:

- Prepare the inputs of the third-party verifications to be provided as a part of the onboarding service. The JSONs can vary based on data such as country, data sources, and customer type. Ready-to-use mappings have been created to use this JSON.
- Prepare the input JSON for the onboarding service. This JSON is then provided as a response for KYC Onboarding.
- Create risk assessments.
- Prepare the input for Customer Screening (CS).
- Prepare the input of the internal watch list.

The Table to JSON utility is used in Enterprise Case Management (ECM) to perform the following tasks:

- Generate the JSON response for the Common Gateway service.
- Generate the JSON response, which will be available in the KYC setup.

**Note**

The Table to JSON utility must only be used in conjunction with the Financial Crime and Compliance Management (FCCM) application packs. Usage with any external applications is not allowed.

## 2.3 JSON to Table Utility

The JSON to Table utility allows users to define where each field's data is available. These values are captured in the database in a tabular format. This utility captures the information of individual fields of JSON as per the configuration.

The JSON to Table utility is used in KYC to:

- Save the JSON response.
- Display the JSON response on your User Interface.

**Note**

The Table to JSON utility must only be used in conjunction with the Financial Crime and Compliance Management (FCCM) application packs. Usage with any external applications is not allowed.



# 3

## Tables and Table Elements

As part of the general configuration, some tables need to be updated to get the JSONs constructed by the utility. Currently, these configurations must be defined by the user in the tables. These tables and their respective columns are explained in the following sections.

### Topics:

- [Table to JSON](#)
- [JSON to Table](#)

### 3.1 Table to JSON

The following tables must be configured so that the Table to JSON utility can construct the JSON:

- [fcc\\_tpg\\_table\\_json\\_mapping](#)
- [fcc\\_tpg\\_table\\_json\\_query](#)
- [fcc\\_tpg\\_table\\_json\\_query\\_params](#)

#### 3.1.1 fcc\_tpg\_table\_json\_mapping

This table allows you to configure the JSON constructed by the utility and is the first table that needs to be configured. Use this table to perform the following tasks:

- Configure the structure of the JSON
- Indicate whether the value of the JSON is from a column in the table or a user input
- Configure the value of the JSON field
- Configure the data type of the field or array of the JSON. To view the table values, run the following query:

```
select * from fcc_tpg_table_json_mapping where mapping_id = 'parent_mapping_id value';
```

The following table describes the values as follows.

**Table 3-1 fcc\_tpg\_table\_json\_mapping Table**

Value Name	Description
Mapping ID	Mapping IDs are used in queries to construct a JSON. Each table has a mapping ID assigned to it and is case-sensitive. For example, TEST_ARRAY_ENH1. Mapping IDs are unique for every JSON which is constructed.

**Table 3-1 (Cont.) fcc\_tpg\_table\_json\_mapping Table**

Value Name	Description
Key	A unique key is assigned for every mapping ID which is generated.
Parent Field	Parent of the key that is generated in the JSON. The parent field of the first layer of keys must be kept as a parent in this field. For the second layer of keys, the parent field is the field in the first layer of keys.
Field	This field captures the exact label of the field name generated in the JSON.
Field Type	This field captures the field type. The types can be ELEMENT, OBJECT, or ARRAY
Field Data Type	This field captures the field data type of the JSON. The default is STRING. Other acceptable values are STRING, NUMBER, and BOOLEAN.
Array Data Type	This field captures the array data type. The default is OBJECT (JSON object). Other acceptable values are OBJECT, STRING, NUMBER, and BOOLEAN.
Data Source	The data source has to be either TABLE or REQUEST. The data source is TABLE if the value comes from a table or expression. The data source is REQUEST if the value comes from the request parameters. For information on the request parameters, see the Oracle Financial Services Know Your Customer Service Guide. <a href="#">Oracle Financial Services Know Your Customer Service Guide</a>
Table Name	This field captures the name of each table that contains the JSON field values. This has to be only in the schema where the web services are deployed. If the field's value is not available in a table, then this will be blank. If the value is static, then this value can be provided in the Expression column. The table name must have the physical table name as defined in the database.
Column Name	This field captures the physical name of the column corresponding to the table, which contains the field values of the JSON.
Expression	SQL expression that generates the field value. It overrides the Table Name and Column Name fields. It must be as is from the SQL query that generates the value.  For example, in the case of <i>SELECT SampleValue FROM DUAL</i> , the expression field would be SampleValue. In the case of <i>SELECT UPPER(TABLE_NAME.COLUMN_NAME) from CUST</i> , the expression field would be <i>UPPER(TABLE_NAME.COLUMN_NAME)</i> .

**Table 3-1 (Cont.) fcc\_tpg\_table\_json\_mapping Table**

Value Name	Description
Key Source	Provide the key source value as REQUEST to make a particular JSON field value dynamic. This means that the key is substituted by the value of the request parameters in the URL.
Related Mapping ID	Provides the mapping ID for the child array element inside the parent array. Each child array value in Field has a corresponding Related Mapping ID. The default of Related Mapping ID is null.
Primary Key	Provides the primary key for the parent array element. For each parent array value in Parent Field, there is a corresponding Mapping ID. The default is null.

### 3.1.2 fcc\_tpg\_table\_json\_query

This table contains the FROM and WHERE clauses of the queries used to construct the JSON. This table has to be updated for each mapping to reflect the FROM and WHERE clauses.

To view the table values, run the following query:

```
select * from fcc_tpg_table_json_query where mapping_id = 'parent_mapping_id
value'
```

The following describes the values as follows.

**Table 3-2 fcc\_tpg\_table\_json\_query Table**

Value Name	Description
Mapping ID	This field captures the mapping IDs which are case-sensitive. This mapping ID has to be the same as the previous table mapping ID.
Parent Field	Parent of the key that is generated in the JSON. The parent field of the first layer of keys must be kept as a parent in this field.
From Clause	FROM clause of the query that provides the data for the JSON. It should be defined at the parent level whenever we need to define a mapping ID.
Where Clause	WHERE clause of the query that provides the data for the JSON. There should only be one record being returned for the OBJECT and ELEMENT types. ARRAYS may have one or more records, and each of those records will become a separate ARRAY element. It should be defined at the parent level, and every time, we need to pass request parameters for the URL and replace the question mark.

**Table 3-2 (Cont.) fcc\_tpg\_table\_json\_query Table**

Value Name	Description
Foreign Key	Provides the foreign key for the child array element. Each child array value in the Field has a corresponding Related Mapping ID. You must define the foreign key for that child array. The default is null.

### 3.1.3 fcc\_tpg\_table\_json\_query\_params

This table contains the request parameters used by the queries in the fcc\_tpg\_table\_json\_query table.

To view the table values, run the following query:

```
select * from fcc_tpg_table_json_query_param where mapping_id =
'parent_mapping_id value';
```

The following table describes the values as follows:

**Table 3-3 fcc\_tpg\_table\_json\_query\_params Table**

Value Name	Description
Mapping ID	Mapping IDs are used in queries to construct a JSON. Each table has a mapping ID assigned to it and is case-sensitive. For example, TEST_ARRAY_ENH1. Mapping IDs are unique for every JSON which is constructed.
Parent Field	Parent of the key that is generated in the JSON. The parent field of the first layer of keys must be kept as a parent in this field.
Parameter Order	Order of the parameter within the WHERE clause of the previous table. The ? values will replace values in the WHERE clause in this order. For example, if the WHERE clause is where ob_cust_seq_id = ? and request_id = ?, the first ? is replaced by the value of the parameter with order 1.
Parameter Name	Name of the query parameter that will be passed through the URL.
Parameter Source	Source of the parameter. REQUEST is the only supported value now.

After the configurations are done, enter the URL for the JSON in the POSTMAN client based on the following format:

```
http://domain:port/ TabletoJSONService/createtabletojson?mappingId= " "
```

## 3.2 JSON to Table

The following tables are used in this utility:

- [fcc\\_ob\\_json\\_table\\_config](#)
- [fcc\\_ob\\_json\\_table\\_map](#)

### 3.2.1 fcc\_ob\_json\_table\_config

This table defines the structure of the input JSON, which needs to be persisted in the KYC tables.

To view the table values, run the following query:

```
select * from fcc_tpg_json_table_config where  
parent_mapping_id='parent_mapping_id value' order by order_used;
```

The following table describes the values as follows.

**Table 3-4 fcc\_ob\_json\_table\_config Table**

Value Name	Description
PARENT_MAPPING_ID	Parent mapping ID of the JSON. The parent_mapping_id value can be one of the following: <ul style="list-style-type: none"><li>• SCORING_RESPONSE</li><li>• CS_WLS_RESPONSE</li><li>• INT_WLS_RESPONSE</li><li>• TRULIOO_ENTITY_RESPONSE</li><li>• TRULIOO_RESPONSE.</li></ul>
JSON_ELEMENT	This field captures the JSON ID value for each JSON element. This value should match the element names in the JSON.
PARENT_JSON_ELEMENT	This field captures the name of the parent JSON element. If there is no parent element, (null) is displayed by default.
DATA_TYPE	This field captures the logical data type of the JSON element. The default is STRING. Other Data types are String array, Array, or JSON.
ORDER_USED	This field provides the order of the JSON elements which need to be captured among the complete JSON structure. The order of the first element to be captured is 1. The order of each subsequent element increases by 1 whenever a particular JSON element is captured.

### 3.2.2 fcc\_ob\_json\_table\_map

This table contains the mappings between the configured JSON elements and the corresponding table elements.

To view the table values, run the following query:

```
select * from fcc_tpg_json_table_map where parent_mapping_id='<
parent_mapping_id value>' order by order_used;
```

The following table describes the values as follows.

**Table 3-5 fcc\_ob\_json\_table\_map Table**

Value Name	Description
MAPPING_ID	Child mapping ID of the parent mapping ID. For example, if the corresponding parent mapping ID is mapped to three different tables, then three unique mapping IDs will be used.
PARENT_MAPPING_ID	Mapping ID which groups all the configurations and the corresponding mappings for the JSON. This must be the same in both config and map tables. The parent_mapping_id value can be one of the following: <ul style="list-style-type: none"> <li>• SCORING_RESPONSE</li> <li>• CS_WLS_RESPONSE</li> <li>• INT_WLS_RESPONSE</li> <li>• IDV_ENTITY_RESPONSE</li> <li>• IDV_RESPONSE.</li> </ul>
JSON_ELEMENT	The JSON elements configured in the config table. Only captures JSON elements of type String or String array.  If a JSON element of type JSON or array is defined in the config table, then that JSON element must not be displayed in the map table.
TARGET_TABLE	The table into which data is persisted.
TARGET_COLUMN	The column into which data is persisted.
EXPRESSION	Values can be the service name, system date, or another query expression. If you do not know the actual expression value, provide the value NOTNULL.
ORDER_USED	The same order used for a particular JSON element in the config table. Only captures JSON elements of type string.

# 4

## Viewing JSON

The following steps show how to view the JSON in different utilities.

To view the JSON for the Table to JSON utility, follow these steps:

1. Open the POSTMAN client.
2. In the **Builder** tab, select the POST method.
3. Enter a URL in the method field. An example is given below. *http://domain:port/TabletoJSONService/createtabletojson?mappingId= ' '*
4. Click **Send**.

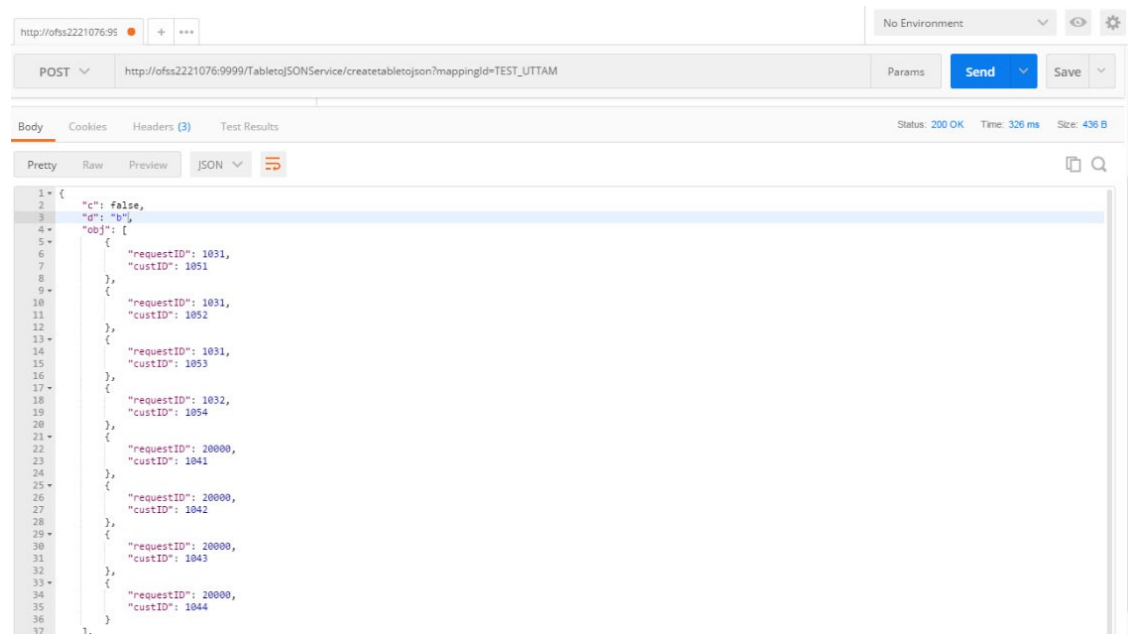
The JSON appears in the Request Body editor.

To view the JSON for the JSON to Table utility, follow these steps:

1. Open the POSTMAN client.
2. In the **Builder** tab, select the **POST** method.
3. Enter a URL in the method field. An example is given below. *http://domain:port/JSONToTablePersistenceUtility/persistJSON?mappingID=' '&requestid=' '&OB\_CUST\_SEQ\_ID=' ' '*
4. Click **Send**.

The JSON appears in the Request Body editor.

**Figure 4-1 Sample JSON**



If there are any errors, they are displayed in the Body editor after you send the request.



# 5

## Mapping IDs and REST URLs for Utilities

This chapter provides the required mapping IDs and REST URLs for the utilities.

For information on the JSON preparation, see the Input preparation, Hitting the individual services, or Output capture section in the [Oracle Financial Services Know Your Customer Service Guide](#).

The following table provides information on the mapping IDs, REST URLs, and the applicable utilities.

**Table 5-1 Mapping IDs and REST URLs for Utilities**

Mapping ID	Description	URL	Utility
CMMN_GATEWAY_INP UT	This mapping ID defines the structure of the JSON to be formed for the common gateway. It holds the definition of each data element and the value to be picked from the table for input creation.	http:// #deployed_server#:#po rt#/ TabletoJSONService/ createtabletojson? mappingId=CMMN_GA TEWAY_INPUT&reques tId={OB_REQUEST_ID &customerCounter={ OB_CUST_COUNT}	Table to JSON
CREATE_ASSMNT	This mapping ID defines the structure of the JSON to be formed for creating an assessment. It holds the definition of each data element and the value to be picked from the table for input creation.	http:// #deployed_server#:#po rt#/ TabletoJSONService/ createtabletojson? mappingId=CREATE_A SSMNT&requestId={OB _REQUEST_ID}	Table to JSON
CS_INPUT_IND	This mapping ID defines the structure of the JSON to be formed for customer screening where the customer type is individual. It holds the definition of each data element and the value to be picked from the table.	http:// #deployed_server#:#po rt #/ CommonGatewayServi ce/createtabletojson? mappingId=CS_INPUT_ IND&customerId={CUS TOMER_ID}&r equestId={REQUEST_I D}	Table to JSON

**Table 5-1 (Cont.) Mapping IDs and REST URLs for Utilities**

Mapping ID	Description	URL	Utility
CS_INPUT_NONIND	This mapping ID defines the structure of the JSON to be formed for customer screening where the customer type is entity. It holds the definition of each data element and the value to be picked from the table.	http:// #deployed_server#:#port #/ CommonGatewayService/createtabletojson?mappingId=CS_INPUT_NONIND&customerId={CUSTOMER_ID}&requestId={REQUEST_ID}	Table to JSON
INTRL_WLS_INPUT	This mapping ID defines the structure of the JSON to be formed for the internal watch list. It holds the definition of each data element and its value to be picked from the table for input creation.	http:// #deployed_server#:#port #/ CommonGatewayService/createtabletojson?mappingId=INTRL_WLS_INPUT&requestId={REQUEST_ID}	Table to JSON
OB_RESP_1	This mapping ID defines the structure of the JSON to be formed as a final response after the onboarding customer data has been processed through all the services.	http:// #deployed_server#:#port #/ TabletoJSONService/createtabletojson?mappingId=OB_RESP_1&requestId=1003	Table to JSON
SCORING_INPUT	This mapping ID defines the structure of the JSON to be formed for the scoring service. It holds the definition of each data element and the value to be picked from the table for input creation.	http:// #deployed_server#:#port #/ TabletoJSONService/createtabletojson?mappingId=SCORING_INPUT&requestId={OB_REQUEST_ID}	Table to JSON
SCORING_RESPONSE	This mapping ID defines where the values of the scoring output in the JSON format have to be captured in the KYC OB tables. This must be in the form of a table. Column mappings for each data element of the JSON.	http:// #deployed_server#:#port #/ JSONToTablePersistenceUtility/persistJSON?mappingID=SCORING_RESPONSE&requestId={OB_REQUEST_ID}	Table to JSON

**Table 5-1 (Cont.) Mapping IDs and REST URLs for Utilities**

Mapping ID	Description	URL	Utility
CS_WLS_RESPONSE	This mapping ID defines where the values of the customer screening output in the form of JSON have to be captured in the KYC OB tables. This must be in the form of a table. Column mappings for each data element of the JSON.	http:// #deployed_server#:#port #/ JSONToTablePersistenceUtility/persistJSON?mappingID=CS_WLS_RESPONSE&requestId={OB_REQUEST_ID}&customerId={CUSTOMER_ID}	JSON to table
INT_WLS_RESPONSE	This mapping ID defines where the values of the internal watch list output in the form of JSON have to be captured in the KYC OB tables. This must be in the form of a table. Column mappings for each data element of the JSON.	http:// #deployed_server#:#port #/ JSONToTablePersistenceUtility/persistJSON?mappingID=INT_WLS_RESPONSE&requestId={OB_REQUEST_ID}&customerId={CUSTOMER_ID}	JSON to table
CREATE_ASSMNT	This mapping ID defines the table. Column will be updated based on the JSON formed when you create an assessment.	http:// #deployed_server#:#port #/ JSONToTablePersistenceUtility/persistJSON?mappingID=CREATE_ASSMNT	JSON to table
KYCOB_CASE_RESPONSE	The action taken on cases gives a JSON response. This mapping ID defines in which table. Column value will be updated.	http:// #deployed_server#:#port #/ JSONToTablePersistenceUtility/persistJSON?mappingID=KYCOB_CASE_RESPONSE	JSON to table

# A

## OFSAASupport

Raise a Service Request (SR) in [My Oracle Support \(MOS\)](#) for queries related to the OFSAA applications.

# B

## Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site which has all the revised/recently released documents.