

Oracle® Cloud

Using Human in the Loop in Oracle Integration 3



G45527-01
November 2025



Oracle Cloud Using Human in the Loop in Oracle Integration 3,

G45527-01

Copyright © 2025, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

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	Get Started with Human in the Loop	
	Find More Resources	1
2	Learn About Human in the Loop	
	Overview of Human in the Loop for Agentic AI	1
	Human in the Loop for Agentic AI in Oracle Integration	2
	Prerequisites for Using Human in the Loop	3
3	Model Workflows	
	Create a Workflow	1
	Understand Start and End Events	2
	Add and Configure User Tasks	3
	About Sequence Flows	4
	Manage Data in Workflows	4
	Work with Data Objects	5
	Work with Expressions	6
	Configure Data Associations	12
	Task Exec Variables	13
	Activate a Workflow	13
	Add a Workflow to an Integration	14
4	Design Forms	
	Create a Form	1
	Work with Form Controls	2
	Position Controls on Forms	2
	Configure Basic Controls	3
	Configure Text Input and Area Fields	4
	Configure Simple Text Fields	5
	Configure Buttons	5
	Configure Select Fields	6
	Configure Checklists and Checkboxes	7

Configure Radio Buttons	8
Configure Number Fields	9
Configure Date and Time Fields	10
Configure Email Fields	11
Configure URL Fields	12
Configure Message Fields	12
Configure Links	13
Configure Advanced Controls	14
Configure Currency (Money) Fields	14
Configure Phone Number Fields	16
Include Images	17
Include Videos	17
Place Controls in Panels, Sections, or Tabs	18
Configure Tables	19
Configure Repeatable Sections	20
Configure Train Controls	21
Configure Divider Controls	22
Style Form Controls	22
Styling Properties	22
Bind Form Data with Controls	24
Activate a Form	24

5 Manage Workflows and Forms

Version a Workflow or Form	1
Clone a Workflow or Form	2
Deactivate a Workflow or Form	3
Edit a Workflow or Form	4
Delete a Workflow or Form	4

6 Use Workflows with Agentic AI

Design an Integration to Call a Workflow	1
Create an Agentic AI Tool Using the Integration	4
Complete Assigned Task as a User	5

About This Content

Using Human in the Loop in Oracle Integration 3 describes how to configure and use human in the loop in Oracle Integration.

Audience

Using Human in the Loop in Oracle Integration 3 is intended for users who want to configure and use human in the loop in Oracle Integration.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Resources

For more information, see these Oracle resources:

- Oracle Integration documentation on the Oracle Help Center.
- Oracle Cloud at <http://cloud.oracle.com>.

Conventions

The following text conventions are used in this document.

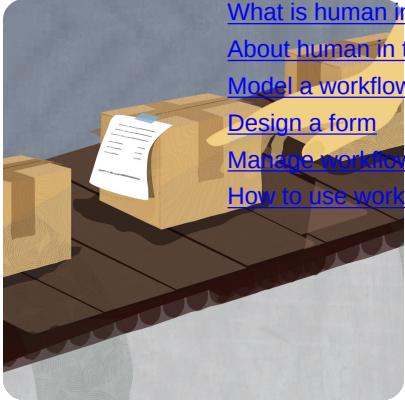
Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type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

1

Get Started with Human in the Loop

Agentic AI can behave in unpredictable ways. When a consistent outcome is important or required, keep a human in the loop to ensure that approvals are managed consistently.

Explore Human in the Loop

Image	Links
	<ul style="list-style-type: none">What is human in the loop?About human in the loop for agentic AIModel a workflowDesign a formManage workflows and formsHow to use workflows with agentic AI?

Find More Resources

Oracle Integration offers many ways to stay connected and grow your expertise.

Resource	Description	Link
Documentation	Find step-by-step instructions, best practices, and more.	Get Started page on the Oracle Help Center
Tutorials	Take a product tour, build an integration from start to finish, or complete a longer self-directed workshop with a LiveLab.	Tours and Tutorials in <i>Using Integrations in Oracle Integration 3</i>
Videos	Learn more about Oracle Integration by watching a short video.	Videos page on the Oracle Help Center
Live demos	Created by the product management team, live demos offer deep dives into new features, use cases, and more. Subscribe to the channel to get notified when a new live demo is available.	Oracle Integration on YouTube
Oracle Integration blog	Learn what's new from the product management team.	Oracle Integration blog
A-Team Blog	Learn from technical solution architects and software engineers.	A-Team Chronicles blog
Cloud Customer Connect	Connect with experts, ask or answer questions, and share your opinions.	Cloud Customer Connect
Newsletter	Oracle Integration sends a quarterly newsletter with product, event, and training information.	Subscribe to the Oracle Integration Newsletter

Resource	Description	Link
Webcasts	Oracle Integration offers the following quarterly webcasts: <ul style="list-style-type: none">• Product Update Webcasts Get product updates and watch live demos.• Customer Success Webcasts Get inspired by success stories and best practices, or present your own wins to the community.	Integration Resources blog post: Watch past webcasts and learn about upcoming webcasts Subscribe to the Oracle Integration Newsletter : Learn about upcoming webcasts by subscribing to the newsletter
Oracle Partners	Use the Oracle PartnerNetwork to find a partner to support your business goals. Connect to the Oracle network to view this page.	Oracle Partner Finder

2

Learn About Human in the Loop

Human in the loop places people at the center of automation solutions powered by agentic AI, ensuring that human expertise and oversight guide decision-making. In the context of agentic AI, learn what human in the loop means, how it is implemented within Oracle Integration, and the prerequisites for using it.

Topics:

- [Overview of Human in the Loop for Agentic AI](#)
Get an overview of human in the loop, including how keeping humans involved boosts trust, accuracy, and accountability when AI agents perform business tasks.
- [Human in the Loop for Agentic AI in Oracle Integration](#)
Discover how Oracle Integration leverages human in the loop to let users monitor, approve, or guide agentic AI, ensuring better control over critical business operations.
- [Prerequisites for Using Human in the Loop](#)
Learn about the prerequisites needed to successfully use the human in the loop feature in Oracle Integration.

Overview of Human in the Loop for Agentic AI

Keeping a human in the loop in an automation solution with agentic AI combines the efficiency of AI agents with human oversight and contextual understanding.

Agentic AI is powered by large language models (LLM) and autonomous AI agents that handle multi-step tasks, make decisions, and interact with other tools or APIs. AI excels at automating repetitive tasks and streamlining business processes. However, the complexity and unpredictability of real world scenarios mean that even the most sophisticated AI agent can encounter uncertainty, ambiguous requirements, and high-stakes decisions that are beyond its capabilities. This is where human oversight becomes essential.

Human in the loop refers to incorporating human supervision and oversight at critical and strategic points in otherwise automated solutions with agentic AI. Allowing humans to review, approve, and provide guidance when agentic AI encounters risks, ambiguity, or bottlenecks helps organizations achieve safer, more reliable, and consistent outcomes. Just like cruise control on a car, AI can handle the driving, but a human remains ready to take the wheel the moment conditions demand it.

Benefits of Keeping a Human in the Loop

- **Trust and Accountability:** Users are more likely to trust AI-assisted outcomes when human judgment plays a role in critical decisions.
- **Enhanced Accuracy and Safety:** Human review helps catch errors and ensures quality, especially in sensitive domains like healthcare or finance.
- **Handling Complex Scenarios:** Humans provide guidance when AI hits a bottleneck, such as unclear, missing, or fast-changing rules or data.

Examples of Human in the Loop with Agentic AI

- **Financial services:** AI agents flag a suspicious transaction, and a human analyst reviews it before blocking the payment or reporting it.
- **Healthcare:** An AI tool helps sort patient cases, but sends high-risk or unclear cases to a clinician for review.
- **Customer service:** A chatbot answers common questions, but passes unusual or sensitive issues for review by a human.

Human in the Loop for Agentic AI in Oracle Integration

In Oracle Integration, human in the loop is designed as an approval orchestration tool to obtain human approval, feedback and oversight in agentic AI powered automation solutions. Human in the loop supports agentic AI by involving humans at critical decision points.

Common Use Cases

- Before an AI agent takes important actions, like creating a high stake purchase order, a human reviews and approves the action, ensuring oversight and accuracy.
- If the AI agent is uncertain about the next step, it can request help from a human instead of stopping the process.
- When an error occurs, such as a failure in the tool used by the AI agent, a human can step in to provide feedback and help resolve the issue, allowing the process to continue.

The human in the loop feature in Oracle Integration, allows you to:

- [Model Workflows](#)
Model approval workflows by assigning specific tasks (such as review or approval) to designated users.
- [Design Forms](#)
Design user friendly forms that assigned users can access in order to perform required actions within those tasks.

Steps to Keep a Human in the Loop with Agentic AI in Oracle Integration

In Oracle Integration, to keep a human in the loop in agentic AI automated solutions use the following approach as general guideline.

	Step	More information
1	Create a project	To create human approval workflows and forms, you must create a project in Oracle Integration. To create a project, see Create or Import a Project in Using Integrations in Oracle Integration 3 .
2	Create a form	Create a form for users to review, approve or take appropriate actions on assigned tasks. Forms are the interface that facilitates human interactions in the workflow. See Design Forms .
3	Create a workflow	Create a workflow, which is the container for other key components such as user tasks, events and gateways. See Create a Workflow .

	Step	More information
4	Add and configure a user task	Add a user task to the workflow and configure its properties including the user to assign it to. See Add and Configure User Tasks .
5	Manage data	Manage how data flows in workflows with data objects and data associations. See Manage Data in Workflows
6	Activate the workflow and form	Activate the workflow and form to use in an integration. See Activate a Workflow and Activate a Form .
8	Create an integration	Create an integration to call the human in the loop workflow. See Design an Integration to Call a Workflow .
9	Create an agentic AI tool	Use the integration created with the workflow to create an agentic AI tool. See Create an Agentic AI Tool Using the Integration .
10	Build an AI agent	Build an agent that uses the agentic AI tool. See Build the AI Agent in <i>Using Agentic AI in Oracle Integration 3</i> .
11	Run and test the AI agent	Run and test the AI agent. When you run the AI agent, your workflow gets invoked and tasks gets assigned to users. See Run and Test the AI Agent in <i>Using Agentic AI in Oracle Integration 3</i> .
12	Complete assigned user task	As a end user who is assigned the user task, access and complete the task in Process Automation Workspace. See Complete Assigned Task as a User .

Prerequisites for Using Human in the Loop

You can use human in the loop in Oracle Integration only if you have enabled Process Automation.

To use human in the loop in Oracle Integration:

- Enable Process Automation for your Oracle Integration instance.
- Assign the predefined Process Automation roles *ServiceDeveloper* or *ServiceAdministrator* to the required users or groups so that they can access the human in the loop feature on your instance.

For detailed instructions to enable Process Automation, see Enable Process Automation with Oracle Integration 3 in *Administering Oracle Cloud Infrastructure Process Automation*.

3

Model Workflows

Workflows introduce human review, feedback and input at specific points in business process automation. These workflows are primarily focused on approval orchestration, involving tasks that users must perform manually.

Common examples of human in the loop workflows:

- **Purchase order approvals:** Purchase order requests that require procurement manager approvals.
- **Leave request approvals:** Employee leave requests needing HR and management review.
- **Expense report approvals:** Expense claims that must be reviewed by finance and supervisors.
- **Contract approvals:** Contract details routed through legal, finance and executive teams for review and sign off.

How to Model Workflows?


Workflows are graphical representations of the events, user tasks and sequence flows that define approval orchestrations. You can model workflows using:

- **Events** that start and end workflows.
- **User Tasks** that require a human to perform an action.
- **Sequence Flows** that connect events and user tasks.

Topics:

- [Create a Workflow](#)
- [Understand Start and End Events](#)
- [Add and Configure User Tasks](#)
- [About Sequence Flows](#)
- [Manage Data in Workflows](#)
- [Activate a Workflow](#)
- [Add a Workflow to an Integration](#)

Create a Workflow

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Human in the loop** .
3. In the **Workflows** section, click **Add**.

- Enter the following details for the workflow:

Field	Description
Name	Enter a name for the workflow.
Identifier	This field is automatically populated with a unique identifier based on the name value. You can manually change this value, if needed.
Version	Use the default value. You can update the version number, if needed.
Description	Provide additional information about the workflow.

- Click **Create**.
A workflow with a start and end event automatically appears in the canvas. You can start designing the workflow according to your business needs by adding user task activities.



Understand Start and End Events

The start and end events of a workflow define how a workflow starts and ends.



A **start event** defines the start of the workflow. It allows you to specify input arguments that triggers the workflow so that the workflow starts executing its defined sequence of activities. A workflow contains a single start event.

An **End event** marks the completion of a workflow. The end event can be configured to define the callback request and specify the output argument.

Configure the Start Event

- Select the **Start event** in your workflow.
- Open the Properties pane using one of the following options:
 - Click **Properties**  on the right tool bar.
 - Select the start event, click  and then click **Open Properties**.
- Enter the name of the start event in the **Name** field.
Tip: To edit the start event name on the canvas, simply double-click the event name and enter your changes.
- Provide additional information in the **Description** field.
- In the **Input Payload Definition** section, drag and drop a JSON payload or enter it inline in the editor.
- Close the Properties pane to save the changes.

Configure the End Event

- Select the **End event** in your workflow.
- Open the Properties pane using one of the following options:
 - Click **Properties**  on the right tool bar.
 - Select the end event, click  and then click **Open Properties**.
- Enter the name of the end event in the **Name** field.
Tip: To edit the end event name on the canvas, simply double-click the event name and enter your changes.

4. Provide additional information in the **Description** field.
5. Drag and drop a JSON payload or enter it inline in the editor to configure the output payload of the callback request.
Typically in the context of agentic AI, the callback request is sent to AI agent, based on which the AI agent performs the necessary action.
6. Close the Properties pane to save the changes.

Add and Configure User Tasks

User tasks are activities in your workflow which require human interaction.

In agentic AI use cases that require human oversight and review, human intervention (human in the loop) is facilitated through user tasks, which are configured and assigned to the appropriate users.


Common User Tasks


- **Review and validation**
Humans review outputs generated by agentic AI systems.
Example: Confirming the accuracy of an OCR-extracted invoice before processing payment.
- **Exception handling**
Humans intervene when agentic AI encounters unexpected scenarios, errors, or ambiguous cases it can't resolve.
Example: Handling a customer support ticket that the chatbot couldn't answer.
- **Approval and authorization**
Certain processes require human approval as a control step for compliance, security, or accuracy, even when initiated by agentic AI.
Example: Approving a financial transaction above a certain threshold initiated by agentic AI.
- **Decision making**
Where agentic AI provides recommendations, humans make the final decision.
Example: Reviewing agentic AI generated candidate shortlists and selecting applicants to interview.


Add a User Task

1. Open a workflow.
2. Select the **User Task** element under **Human** in the right pane.
3. Drag and drop the user task onto the workflow canvas. Position it in the workflow where you want it to appear.

Configure User Task Properties

1. Select the **User Task** element in your workflow.
2. Open the Properties pane using one of the following options:
 - a. Click **Properties**  on the right tool bar.

- b. Select the user task, click  and then click **Open Properties**.
 3. In the Properties pane, enter the following details.
 - a. In the **Name** field, enter a name for the task.

Tip: To edit a user task name on the canvas, simply double-click the task name and enter your changes.
 - b. In the **Description** field, provide additional information about the task.
 - c. In the **Assignees** section, click **Assign to**  in order to assign a user to the task.
 - d. In the **End User Display** section, enter the following:

Field	Description
Title	Enter the title of the task that will get displayed to users to whom the task is assigned. Note that this is a required field.
Summary	Optionally, provide a description for the task.
UI	Select a form to associate with the task. Only active forms are displayed and available for selection. For details on how to design and build forms, see Design Forms . After designing the form, activate it. See Activate a Form . This is a required field.

4. Close the Properties pane to save the configurations.


About Sequence Flows

A sequence flow connects the events and user tasks in your workflow. It defines the order or sequence that work is performed within a workflow.

Incoming sequence flows are the sequence flows that lead into an event or user task.

Outgoing sequence flows are the sequence flows that determine the path out of an event or user task. User tasks contain both incoming and outgoing sequence flows. Start events can only contain outgoing sequence flows. End events can only contain incoming sequence flows.

To add sequence flows:

1. Select the start event or user task where you want to create the outgoing sequence flow.
2. Click  and keep the mouse depressed.
3. Move the cursor to the event or user task that you want to connect to, and then release the click.

Manage Data in Workflows

Human approval workflows require some type of data. For example, a workflow for purchase orders needs information such as the customer's name, contact details, order number, items to be approved, and payment details. To manage data in your workflows, create data objects to

store data, learn how to perform calculations on data objects with expressions, and configure data associations in events and user tasks to define how data flows in and out of them.

Topic:

- [Work with Data Objects](#)
- [Work with Expressions](#)
- [Configure Data Associations](#)

Work with Data Objects

Data objects are variables used to define the type of information used by your workflow. They are also used to store the value of this information.

Data Object Types

You can create and use data objects of the following types.

Type	Description
String	Represents a sequence of characters. For example: <code>This text is a string.</code>
Number	Represents whole numbers and numbers with fractional or decimal components. For example: <code>-5.75, 0.0, 100.25</code>
Boolean	Represents the logical values <code>true</code> or <code>false</code> .
Date	Stores year, month, and day information, typically in a <code>YYYY-MM-DD</code> format.
Time	Represents a specific time expressed as <code>hour:minute:second</code> .
DateTime	Represents a specific date and time expressed as: <code>year-month-day hour:minute:second</code> . For example: <code>2014-12-14 13:20:28</code>
Duration	Represents a duration of time expressed as a number in years, months, days, hours, minutes, and seconds. For example: <code>1d3h30m</code> .
Integer	Represents an integer. For example: <code>23, -10, 0</code> .

Create a Data Object

1. Open a workflow.
2. In the vertical toolbar on the right side of the workflow canvas, click **Data**.
3. The Data pane appears and the following sections are displayed - Workflow Data, Input and Output.
 - **Workflow Data** lists all the data objects used in the workflow.
 - **Input** lists the data objects for receiving data into the workflow in the start event.
 - **Output** lists the data objects for data that goes out of the workflow in the end event.
4. In the Data pane, click **Create Data Object**. The Create Data Object pane appears.
5. In the **Name** field, enter a name for your data object or use the default name.
6. Under **Data Type**, select **Simple** and then choose a data type from the drop-down menu.
7. Click **Create**.

Editing and Deleting Data Objects


- After editing a data object, you must ensure that all references to it are still valid. For example, if you change a data object type from an integer to a string, you must verify that all of the expressions that use the data object still function correctly. If they don't function correctly, you won't be able to activate the workflow due to validation errors.
- After deleting a data object, you must ensure that all references to it are removed. This includes any data associations and expressions that use the data object. If you don't remove references to the deleted data object, you won't be able to activate the workflow due to validation errors.

Work with Expressions

Use expressions to evaluate and perform calculations on data stored in data objects.

You can create expressions when configuring data associations and when configuring properties for user tasks.

Create Expressions

- To create an expression, click  from the user task's properties field. This enables the expression mode and you can use the inline expression builder which gives auto-complete options for you to create and build your expression. You can use an expression to dynamically determine the task title, summary, due date, and assignees.
- To create an expression while configuring data associations, enter the expression directly in the input or output fields of the data association editor. You can use the inline expression builder which gives auto-complete options for you to create and build your expression.

Key Points to Note About Expressions

- Create expressions with data objects, operators, and functions. As soon as you put the cursor in the expression field, you get a list of suggested objects using which you can build the expression.
- As you choose the object, related suggestions for the next level are displayed below the field. The suggestions are context sensitive. For example, suggestions displayed for a **Title** field will be different from that of a **Due Date** field.
- Use Ctrl + Space to get more tabs and suggestions. A search field also displays that you can use to search data objects under the various tabs such as **Process** or **Activity**.
- The type that is expected to be entered into the expression field is auto-suggested at the top right of the field.
- If an expression is invalid, for example, if you type a number in a string field, an error message indicating the details of the error is displayed.
- If the configured expression is too long, you can expand the field by using Ctrl + Enter keys.

Simple Expressions

Simple expressions are defined using a basic expression language and support. Generally simple expressions perform their calculations based on the data objects in your workflow. You can write expressions and conditions using the value of the data objects, but you can't explicitly modify the value within the data object.

Here are some examples of expressions using operators:

- `totalAmount - discount`
- `activationCount > 3`
- `unitsSold <= 1200`

Operator Precedence

Operator precedence defines the order in which the compiler evaluates operators. You can change operator precedence in an expression by using parentheses.

- Addition, Subtraction
- Multiplication, Division, Remainder
- Plus, Minus
- Less than, Greater than, Less than or equal to, Greater than or equal to
- Equals, Not equals
- Not
- Conditional And
- Conditional Or

The following sections lists Operators and Functions per type.

Unary

Operator	Name	Description
+	Plus	Has no effect on the value of the numeric operand. Use it to explicitly indicate that a certain value is positive.
-	Minus	Negates an arithmetic expression.
!	Not	Logical complement operator. Negates the value of a Boolean expression.

Equality and Relational

Operator	Name	Description
= or ==	Equal to	Returns true if the first operand is equal to the second operand
!=	Not equal to	Returns true if the first operand isn't equal to the second operand
>	Greater than	Returns true if the first operand is greater than the second operand
>=	Greater than or equal to	Returns true if the first operand is greater than or equal to the second operand
<	Less than	Returns true if the first operand is less than the second operand
<=	Less than or equal to	Returns true if the first operand is less than or equal to the second operand

Conditional

Operator	Name	Description
and	Conditional And	Returns true if both operands evaluate to true
or	Conditional Or	Returns true if either operand evaluates to true

String

Function/ Operator	Description	Usage Expression	Usage Result
+	String concatenation	"pine" + "apple"	"pineapple"
==	Equals	"apples" == "apples"	true
!=	Not equals	"apples" != "oranges"	true
>	Greater than	"word" > "work"	false
>=	Greater than or equals	"work" >= "work"	true
<	Less than	"word" < "work"	true
<=	Less than or equals	"work" <= "work"	true
contains	Returns true if the first argument string contains the second argument string; otherwise returns false	"caramel".contains("ram")	true
endsWith	Returns true if the first argument string ends with the second argument string; otherwise returns false	"immutable".endsWith("table")	true
length	Returns the number of characters in a string	"house".length()	5
lowerCase	Returns a string with all the characters in the argument converted to lower-case representation	"Example".lowerCase()	"example"
startsWith	Returns true if the first argument string starts with the second argument string, otherwise returns false	"caramel".startsWith("car")	true
substring	Returns the substring of the first argument starting at the position specified in the second argument and continuing to the end of the string	"care".substring(2)	"are"
substring	Returns the substring of the first argument starting at the position specified in the second argument with length specified in the third argument	"care".substring(1,3)	"car"
upperCase	Returns a string with all the characters in the argument converted to upper-case representation	"Example".upperCase()	"EXAMPLE"
replaceAll	Replaces each substring of this string that matches the given pattern with the given replacement.	"care".replaceAll("e", "t")	"cart"
indexOf	Returns the index of the first occurrence of the specified substring, or -1 if there is no such occurrence.	"care".indexOf("a")	2

Numeric

The following table includes operators for both `Integer` and `Number`.

Operator	Description	Usage Expression	Usage Result
+	Addition	2 + 8	10
-	Subtraction	7 - 4	3
*	Multiplication	3 * 4	12
/	Division	3 / 2	1.5
%	Remainder	3 % 2	1
==	Equals	12 == 13	false
!=	Not equals	12 != 13	true
>	Greater than	15 > 16	false
>=	Greater than or equals	15 >= 15	true
<	Less than	12 < 10	false
<=	Less than or equals	12 <= 12	true
abs	Returns the absolute value of a number	abs(- 6)	6

In addition to the above, the following specific operators are also available for `Number`.

Operator	Description	Usage Expression	Usage Result
floor	Returns the largest (closest to positive infinity) number that isn't greater than the argument and is an integer	floor(5.60)	5
ceil	Returns the smallest (closest to negative infinity) number that isn't less than the argument and is an integer	ceil(5.60)	6
round	Returns the number that is closest to the argument and is an integer	round(5.60)	6

Date and Time

Date and Time types include `Date`, `Time`, and `DateTime`. All of them have the following operators.

Operator	Description
+	Addition (valid only when the second argument is a duration)
-	Subtraction (valid only when the second argument is a duration)
==	Equals
!=	Not equals
>	Greater than
>=	Greater than or equals
<	Less than
<=	Less than or equals
format	Returns the formatted string of date-time using the provided format picture

In addition, `Date`, `Time`, and `DateTime` have specific operators as listed under each of them.

Date

Operator	Description
<code>year</code>	Returns a number representing the year component of the date-time argument.
<code>month</code>	Returns a number representing the month component of the date-time argument.
<code>day</code>	Returns a number representing the day component of the date-time argument.

Time

Operator	Description
<code>hours</code>	Returns a number between 0 and 23, both inclusive, representing the hours component of the date-time argument.
<code>minutes</code>	Returns a number between 0 and 59, both inclusive, representing the minutes component of the date-time argument.
<code>seconds</code>	Returns a number between 0 and 59, both inclusive, representing the seconds component of the date-time argument.

DateTime

Operator	Description
<code>year</code>	Returns a number representing the year component of the date-time argument.
<code>month</code>	Returns a number representing the month component of the date-time argument.
<code>day</code>	Returns a number representing the day component of the date-time argument.
<code>hours</code>	Returns a number between 0 and 23, both inclusive, representing the hours component of the date-time argument.
<code>minutes</code>	Returns a number between 0 and 59, both inclusive, representing the minutes component of the date-time argument.
<code>seconds</code>	Returns a number between 0 and 59, both inclusive, representing the seconds component of the date-time argument.
<code>timezone</code>	Returns an interval value, representing the time offset from UTC.

Operator	Description
toTimezone	<p>Returns the date-time expressed in the time offset corresponding to the timezone ID provided.</p> <p>You have the following options for specifying a timezone ID:</p> <ul style="list-style-type: none"> Provide a fixed offset from UTC/Greenwich, such as "-07:00". <p>For example:</p> <ul style="list-style-type: none"> <code>dateTimeDO.toTimezone("-07:00")</code> returns 2002-11-30T17:20:00-07:00 when <code>dateTimeDO</code> is 2002-12-01T01:20:00+01:00 Specify a geographical region, which is an area where a specific set of rules for finding the offset from UTC/Greenwich apply. A geographical region is usually represented in the format "{area}/{city}", such as "Europe/Amsterdam" or "America/New_York". <p>For example:</p> <ul style="list-style-type: none"> <code>dateTimeDO.toTimezone("America/Los_Angeles")</code> returns 2002-11-30T16:20:00-08:00 when <code>dateTimeDO</code> is 2002-12-01T01:20:00+01:00

Note

You must include double quotation marks around the timezone ID value.

For more information about timezone IDs, see [Class ZoneId](#) in the Java Platform documentation.

Boolean

Operator	Description	Usage Expression	Usage Result
==	Equals	<code>true == true</code>	true
!=	Not Equals	<code>true != false</code>	true
and	Conditional — And	<code>true and false</code>	false
or	Conditional — Or	<code>true or false</code>	true
not	Logical complement operator, inverts the value of a Boolean expression.	<code>not true</code>	false

Duration

Operator	Description
==	Equals
!=	Not equals
>	Greater than
>=	Greater than or equals
<	Less than
<=	Less than or equals


Configure Data Associations

Data association refers to the flow of data within a workflow. Use the data association editor to define input and output for workflow elements that need them.

A data association involves a source and a target, where the source provides a value or an expression to be assigned to the target. A user task, for example, needs both input and output data association.

- On the input side, it needs data input into the user task (referred to as its payload).
- On the output side, after the task has completed, it needs output from the task to data objects, to store results for use elsewhere.

1. Open a workflow.

2. Select the user task, click  and then click **Open Data Association**.

The data association editor appears. Use its center pane as a canvas for mapping associations, by dragging and dropping elements from the left and right panes. Any data associations already configured display below the center pane. Use the empty pair of fields to create a new association.

3. In the data association editor, click the **Input** tab to define data input into the user task, and the **Output** tab to define data output resulting from the user task.

- The **Input** tab is selected by default. The left pane displays source objects (**Data Objects**) in an expandable tree. The right pane displays the payload, or entry parameters the activity needs to perform its function.
- Click the **Output** tab and notice how the left and right panes changes. Now the left pane displays the user task's payload, and the right pane displays the variables available in the workflow.

4. In the **Input** tab, begin creating a new association by dragging an input object from the **Data Objects** tree and dropping it in the input field titled *New Association*.

You can also begin typing and select from auto-complete options that appear using the inline expression builder. To specify attributes within objects, enter a . (period) and select from the list that displays. See [Work with Expressions](#).

5. Complete the association by dragging an output object from the payload tree and dropping it in the output field. You can also begin typing and select from auto-complete options that appear using the inline expression builder.

If the data association is valid, the data association icon turns green. The icon turns red for an invalid association.

6. If needed, create additional associations and reorder them.

Drag and drop an association to move it up or down in the associations list. Associations are executed in the order in which you position them. For example, if multiple associations assign a value to the same object, the last assigned value is used.

7. Click the **Output** tab, and create output data associations. You can think of the output as though the activity has just finished, and likely contains results that you'd like to store for use elsewhere in the workflow.

8. Click **Apply** to save the data associations.

Task Exec Variables

Task exec variables can be mapped in output data associations to store task execution details in your workflow.


When configuring output data associations for a user task, the following task exec variables are supported:


Variable	Description	Type
taskExec.taskId	Task number	String
taskExec.title	Title	String
taskExec.approverDisplayName	First name and last name of latest approver.	String
taskExec.summary	Summary	String
taskOutcome	Outcome of a task. For example, APPROVE or REJECT.	String
taskExec.approvers	Details of each approver. Use when multiple approvers act on a task.	Array of objects, contained fields.
taskExec.approvedDate	Date of task action. For example, date of approval.	DateTime
taskExec.approver	System identifier of latest approver.	String
taskExec.createdDate	Created Date	DateTime
taskExec.actionDate	Date on which action is taken. For example, the date when approval is taken for a task.	DateTime
taskExec.assignedDate	Assigned Date	DateTime
taskExec.priority	Specify the priorities as: <ul style="list-style-type: none"> • 1 for high • 2 for medium • 3 for low 	int
taskExec.fromUser	Creator or user who reassigned the task.	Object (Identity Type). Contains the following fields: <ul style="list-style-type: none"> • id • name • display name • type
taskExec.approverName	Username of the latest approver.	String
taskExec.dueDate	Due date	DateTime

Activate a Workflow

After modeling a workflow, activate it to use it in an integration.

Prerequisite: Before activating a workflow, review and fix all errors in the workflow.

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Human in the loop** .

3. In the Workspace section, ensure that the workflow you want to activate is in the **Configured** state.
4. Point to the Configured workflow, click , and then select **Activate**.

A Confirmation pop-up appears, and the state changes to **Activation in progress**. Within a minute or two, the workflow's state changes to **Active**.

Add a Workflow to an Integration




After activating a workflow, you can add it to an integration and use it for business automation that needs human review and approval.


Although the primary use case of human in the loop in Oracle Integration is with agentic AI, human in the loop can also be used for automation that don't use agentic AI. The steps described here are for calling a workflow in an integration that is not used by agentic AI.

Prerequisites

- You must create an integration in your project. To create an integration, see [Create an Integration in *Using Integrations in Oracle Integration 3*](#).
- The integration must be in the **Configured** state.

To add a workflow to an integration:

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Integration**.
3. In the **Integrations** section, point to the required integration, click , and click **Edit**.
4. In the integration designer, add a human in the loop action to the integration.
 - a. Click **Actions**  to open the Action panel.
 - b. From the Call section on the panel, drag the **Human in the loop** action and drop it onto the  sign that appears when you hover over a connection arrow between elements.

Alternatively, hover over a connection arrow, click the  sign that appears, and select the **Human in the loop** action from the dialog box.
5. In the Configuration Wizard that appears, configure the human in the loop action.
 - a. On the Configure Basic Info page, provide a unique name and description in the following fields.
 - **What do you want to call your endpoint?**
 - **What does this endpoint do?**
 - b. Click **Continue**.
 - c. On the Configuration page, complete the following.
 - i. From the **Process** drop-down, select the workflow that the integration calls.
 - ii. From the **Version** drop-down, select the appropriate version of the workflow selected.

iii. Click **Continue**.

d. On the Summary page, review the data and click **Finish**.

You've successfully added a human in the loop action to the integration. Now, a corresponding map action appears on the canvas.

To continue configuring your integration, see the following:

- Map Data in *Using the Oracle Mapper with Oracle Integration 3*
- Create Application Integrations in *Using Integrations in Oracle Integration 3*

To call a workflow in an integration that can be used as a tool in agentic AI, see [Design an Integration to Call a Workflow](#).

4

Design Forms

A form is a user interface with structured fields where users can enter, select, or edit required data as part of an approval process that involves human review. These forms are designed specifically to be used by human in the loop workflows.

Common examples of human in the loop forms:


- Purchase order approval forms
- Employee leave request forms
- Expense report approval forms
- Contract approval forms

Topics:

- [Create a Form](#)
- [Work with Form Controls](#)
- [Activate a Form](#)

Create a Form

To create a new form:

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Human in the loop** .
3. In the **Forms** section, click **Add**.
4. Enter the following details for the form:

Field	Description
Name	Enter a name for the form.
Identifier	This field is automatically populated with a unique identifier based on the name value. You can manually change this value, if needed.
Version	Use the default value. You can update the version number, if needed.
Description	Provide additional information about the form.

5. Click **Create**.

The form canvas appears. You can start designing your form by adding controls from the Basic and Advanced palettes on the right of the canvas.

Work with Form Controls

Form controls are interactive elements, such as text boxes, checkboxes, and checklists, that make up the fields of a form. They are the building blocks used to design forms that allow users to enter, select, or modify data.

Learn more about form controls and how you can use them to design forms:

- [Position Controls on Forms](#)
- [Configure Basic Controls](#)
- [Configure Advanced Controls](#)
- [Style Form Controls](#)
- [Bind Form Data with Controls](#)

Position Controls on Forms

You have many options for positioning controls in your form.

Rows and Columns

You can drop and position controls onto a form in rows and columns. Rows and columns are defined based on where you want to drop your control.

- Each row has a drop target on the top and bottom.
- Each column has a drop target on the left and right.

Each time you drop a control onto the canvas, the control is placed in a new row. If you drop a control beside another control, then that control is added to the same row but in a new column. The column size is based on the number of controls in a row.

The row is based on a 12-column grid system. For example, if the row has four controls, then each uses three columns. When more than 12 controls are added to a row, the remaining controls are displayed below the table.

Panels, Sections, and Tabs

You can group different controls inside a single control by using panels, sections, and tabs.

A **Panel** control allows you to group multiple controls inside it. Drag and drop controls inside a panel and position them as you would in the canvas.

A **Section** control is similar to a panel, but is collapsible. You can use the section control to create groups of controls that users can expand and collapse. To show the section expanded when the form loads, select **Expand** in the General tab. To show the section collapsed when the form loads, deselect **Expand**. You can drag and drop any other controls inside a section control, including panels and other section controls.

A **Tab** control allows you to group controls in different tabs, to create a tabbed view in your web form. By default, when you drag a tab control onto your web form, one tab is created. To add another tab, click **Add tab**. The new tab is added to the right of the existing one. To rearrange the tab order, drag a tab to the right or left of another tab. In small devices, such as mobile phones, tabs are shown as accordions to accommodate the content in the space. You can drag and drop other controls, such as panels, sections, and tables, into an individual tab. When you select a tab, only the controls in the currently selected tab can be seen.

Note

If you delete a panel, a section, or a tab, all the controls grouped inside the panel, section, or tab also get deleted.

See [Place Controls in Panels, Sections, or Tabs](#).

Tables and Repeatable Sections

Use **Table** controls to conserve space in a form. The table control allows you to arrange the form controls in a grid pattern. You can add columns into the table and add one control per column. When you delete a column in the table, all the controls in the column get deleted.

If auto binding is selected when you drop a table onto the canvas, then a new entry in the data definition is generated. That attribute is an array of type object. Every time a control is dropped into a column, an attribute inside that list is added.

If auto binding isn't selected, you can select an attribute from the data definition. The controls inside a table can be bound to any attribute in the form data definition. When rendering the form, the data is fetched and shown for that control. However, modifying the value in one row doesn't update every row. Also, when submitting the form, the last control bound to that attribute is the one updating the payload.

Use **Repeatable Section** controls to display multiple copies of a set of controls in a form. Repeatable sections are useful when you want to allow users to enter the same type of input information multiple times in the form; for example, a phone number.

Unlike a table, which is structured, a repeatable section is a free-form container. You can add new controls above, below, or between existing controls. You can surround an individual control with a repeatable section control, or it can include an entire section control. The section control included in the repeatable section must contain all of the form controls that must be repeated.

See [Configure Tables](#) and [Configure Repeatable Sections](#).

Configure Basic Controls

The Basic palette contains a set of basic controls that you can drag onto the canvas and build a simple form.

When you select a control on the canvas, the **General** and **Styling** tabs displays in the **Properties** pane so you can configure settings specific to the selected control.

Topics:

- [Configure Text Input and Area Fields](#)
- [Configure Simple Text Fields](#)
- [Configure Buttons](#)
- [Configure Select Fields](#)
- [Configure Checklists and Checkboxes](#)
- [Configure Radio Buttons](#)
- [Configure Number Fields](#)
- [Configure Date and Time Fields](#)

- [Configure Email Fields](#)
- [Configure URL Fields](#)
- [Configure Message Fields](#)
- [Configure Links](#)

Configure Text Input and Area Fields

A text input control allows users to enter short, single-line text, such as name or email. A text area control allows users to enter longer, multi-line text entries, such as comments and descriptions.

To configure a text input or a text area control:

1. From the Basic palette, drag an **Input text** or a **Text Area** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Placeholder	The default text that will appear in the control until any text is entered by the user.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Min Length	Defines the minimum number of characters a user must enter into the control.
Max Length	Defines the maximum number of characters a user can enter into the control.
Rows	Defines the number of text rows that will be visible to a user. A scroll bar appears automatically when the number of text rows entered by the user is greater than the value specified. This option is only available for text area controls.
Pattern	Allows you to define custom validations on the type of text a user enters into the control. Enter a pattern using regular expressions. When you specify a pattern, you can also specify a message in the Pattern Validation Message field that will display if the validation fails. This option is only available for text input controls.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Autocomplete	Select this property to display autocompletion entries when users begin to enter text into the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.

Field	Description
Hide	Select this property to hide the control.
Password	Select this property to create a secure text field that masks the characters entered into it.
Automatic Height	Select this property to automatically adjust the height of the control to fit content. This option is available only for text area controls.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Simple Text Fields

Use a simple text control to specify a label without binding for a form field.

To configure a simple text control:

- From the Basic palette, drag the **Simple Text** control onto the canvas.
- Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Default Text	The default text that will display to users when the form loads.
Hide	Select this property to hide the control.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Buttons

Use a button control to add a button, such as a Submit button, to your form.

To configure a button control:

- From the Basic palette, drag the **Button** control onto the canvas.
- Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the control name that displays to a user.
Disabled	Select this property to display the control as inactive.
Hide	Select this property to hide the control.
Auto Focus	Select this property to automatically select the button when the form loads.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Select Fields

Use a select control to add a drop-down list to your form.

To configure a select (drop-down list) control:

1. From the Basic palette, drag the **Select** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Placeholder	The default text that will display until the user clicks the control.
Hint	Hint text that will display when a user clicks the control.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Multiple	Determines whether users can select multiple values in the control. If this field is disabled, users can select only one option in the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Hide	Select this property to hide the control.
Sorted	Select this property to display the control options sorted alphabetically.
Options Source	Specify if the control's options should come from static values you enter or from a list of values data attributes. <ul style="list-style-type: none"> • Static: Specify choices using Options Names and Options Values fields. Use Options Names to specify the label to display for an option and use Options Values to specify an internal value for an option. • From Data: In the Options List field, select a list of values options source from the data definitions available in the form. If you selected a list of complex elements, then, in the Label Binding field, specify a data attribute that will display as the label and in the Value Binding field, specify a data attribute that will be the value.
Default Value	If you selected Static in the Options Source field, then, specify a default option in this field. If you selected From Data in the Options Source field, then select either the first or the last value as the default value.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Checklists and Checkboxes

Use a checklist control to add a list of options to your form. A checklist allows users to select one or more options. Use a checkbox control to your form to allow users to specify a true or false value.

Note that a check box control is set to false by default. You can edit the **Default Value** property on the **Properties** tab to change the default value.

To configure a check list or a check box control:

1. From the Basic palette, drag the **Checklist** or the **Checkbox** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the control name that displays to the user. For check box controls, if you select the Inline HTML check box below the Label field, the value that you enter in the Label field is treated as an inline HTML.
Binding	Defines a link between the control and a data attribute.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Inline	Specify the layout of the options defined for the checklist control. If enabled, the layout changes from vertical to horizontal.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Hide	Select this property to hide the control.
Sorted	Select this property to display the options sorted alphabetically for the checklist control.
Options Source	Specify if the checklist control's options should come from static values you enter or from a list of values data attributes. <ul style="list-style-type: none"> • Static: Specify choices using Options Names and Options Values fields. Use Options Names to specify the label to display for an option and use Options Values to specify an internal value for an option. • From Data: In the Options List field, select a list of values options source from the data definitions available in the web form. If you selected a list of complex elements, then, in the Label Binding field, specify a data attribute that will display as the label and in the Value Binding field, specify a data attribute that will be the value.
Default Value	If you selected Static in the Options Source field for a checklist control, then specify a default option in this field. If you selected From Data in the Options Source field for a checklist control, then select either the first or the last value as the default value. For a checkbox control, select either True or False as the default value.
Auto Focus	Select this property to automatically focus on the checklist control or a specific option in the checklist control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Radio Buttons

Add mutually exclusive radio buttons to your web form. Radio buttons allow users to select an option from a set of available options.

To configure a radio button control:

1. From the Basic palette, drag the **Radio Button** control onto the canvas.
2. Select the control and edit its properties on the Properties pane's **General** tab.

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Hide	Select this property to hide the control.
Inline	Specify the layout of the options defined for the control. If selected, the layout changes from vertical to horizontal.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's required.
Sorted	Select this property to display the control options sorted alphabetically.
Options Source	Specify if the control's options should come from static values you enter or from a list of values data attributes. <ul style="list-style-type: none"> • Static: Specify choices using Options Names and Options Values fields. Use Options Names to specify the label to display for an option and use Options Values to specify an internal value for an option. • From Data: In the Options List field, select a list of values options source from the data definitions available in the web form. If you selected a list of complex elements, then, in the Label Binding field, specify a data attribute that will display as the label and in the Value Binding field, specify a data attribute that will be the value.
Default Value	If you selected Static in the Options Source field, then, specify a default option in this field. If you selected From Data or Connector in the Options Source field, then, select either the first or the last value as the default value.
Auto Focus	Select this property to automatically focus on the radio button control or a specific option in the radio button control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Number Fields

Add number controls to your form to allow users to enter decimal numbers.

To configure a number control:

1. From the Basic palette, drag and drop a **Number** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Default Value	Sets a value to display to users when the form loads.
Hint	Hint text that will display to users when a user clicks the control.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Max	Sets a maximum value that users can enter into the control.
Min	Sets a minimum value that users can enter into the control
Show up/down buttons	Select this property to display the up and down arrows used to increment or decrement the control value.
Step	This property appears if you selected the Show up/down buttons property. Specifies a value based on which the number will increase or decrease when users increment or decrement the value with the up or down arrow. By default the step value is set to 1. Step must be always greater than 0 and you can't enter a value lower than 1.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Autocomplete	Select this property to display autocompletion entries when users begin to enter text into the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.
Hide	Select this property to hide the control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Date and Time Fields

A date time control allow users to enter date and time together. To allow users to enter date and time separately enter a date control and a time control.

To configure a Date Time control:

1. From the Basic palette, drag and drop a **Date Time** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Placeholder	Hint text that describes the expected value. This hint text will display in the control before users enter a value.
Help	Help text to display to users when they hover over or click the control's help icon.
Default Value	Sets the default date and time that appears in the control when the form loads and the current control value is empty.
Max Time	Sets a maximum date and time users can enter in the control.
Min Time	Sets a minimum date and time users can enter in the control.
Max Date	Sets the latest date users can select.
Min Date	Sets the earliest date users can select.
Format	Specifies a date format for the control. You can select from the available date formats such as yyyy-MM-dd, MM/dd/yyyy, dd-MM-yyyy and so on.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Hide	Select this property to hide the control.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.
Prevent Free Text	Select this property to prevent users from entering free text through the keyboard. When selected, users can enter a date only through the date-selection widget.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Note

When a user enters data into a Date Time control, the timezone information of the user is saved (in UTC). When other users access this data, the date-time information is displayed in timezones specific to them. This timezone conversion capability is available only in the Date Time control and not in the separate Date and Time controls.

Configure Email Fields

Add an email control to allow users to enter a valid email address to a form.

1. From the Basic palette, drag and drop an email control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Default Value	Specifies an email to display to users when the form loads and the current control value is empty. You must ensure that you enter a valid email format in the field, otherwise you will get a message indicating that the email format entered isn't valid.
Placeholder	Hint text that describes the expected email format. This hint text will display in the control before users enter a value.
Hint	Useful hint text that displays to users when they select the control.
Help	Help text to display to users when they hover over or click the control's help icon.
Max Length	Sets the maximum number of characters users can enter before the @ symbol for the email.
Min Length	Sets the minimum number of characters users can enter before the @ symbol for the email.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Autocomplete	Select this property to display autocompletion entries when users begin to enter text into the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.
Hide	Select this property to hide the control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure URL Fields

Add a URL control to allow users to enter a web address URL to your form.

1. From the Basic palette, drag and drop a **URL** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Default Value	Sets a value to display to users when the form loads and the current control value is empty. You must ensure that you enter a correct web address URL format in the field, otherwise you will get a message indicating that the URL format entered isn't valid.
Placeholder	Hint text that describes the expected value. This hint text will display in the control before users enter a value.
Hint	Useful hint text to display to users when they select the control.
Help	Help text to display to users when they hover over or click the control's help icon.
Max Length	Specifies the maximum number of characters users can enter into the control.
Min Length	Specifies the minimum number of characters users must enter into the control.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Autocomplete	Select this property to display autocompletion entries when users begin to enter text into the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.
Hide	Select this property to hide the control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Message Fields

Use a message control to allow users to enter a simple message to your form.

1. From the Basic palette, drag and drop a **Message** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Binding	Defines a link between the control and a data attribute.
Default Text	The default text message that will appear to users when the form loads. You can use inline HTML tags to format the default text. Before using, you have to select Inline HTML from the Type drop-down list.
Type	Sets the style and format in which the message displays. For example, as a bold heading or as a paragraph text. Note that predefined message types such as Error, Info, Success, and Warning are available for selection. These message types have their unique style and format, so that the user can easily identify if the message is an error, info, success, or warning. Select Inline HTML if you want to use inline HTML tags to format the default text message that displays when the form loads.
Hide	Select this property to hide the control.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Links

Use the link control to insert a URL into a form. You can specify the value for the link URL or configure the URL value to change dynamically based on the payload. For example, a URL link that contains an order item could change based on the order ID, which could be in the payload.

- From the Basic palette, drag and drop a **Link** control onto the canvas.
- Select the control and edit its properties on the Properties pane's **General** tab.

Field	Description
Name	Defines an internal name for the control. It's an internal identifier that you'll use to identify the control.
Label	Specifies the control name that displays to a user.
Static	Select this option to use the value in the Default Label field as the control name when the form loads. This is the default selection.
Dynamic	Select this option to assign the control name dynamically when the form loads. If you select this option, you must create a data attribute in the Data pane and link it to the control using the Label Binding field. When the form loads, the value of the data attribute is fetched from the payload and assigned as the control name.

Note

If the attribute value is not available in the payload, then the value in the **Default Label** field is used as the control name.

Label Binding	This property appears if you selected a dynamic label. Defines a link between the control's label and a data attribute. Specify an attribute for this field by selecting an option from the auto-complete list or entering a valid binding.
---------------	--

Field	Description
Default Label	Sets a label to display to users when the form loads. The value in this field is used as the control name in the following contexts: <ul style="list-style-type: none"> When you select the Static option in the Label field. When you select Dynamic option in the Label field but the binding value is not available.
Value Binding	Defines a link between the control and a data attribute. Data attribute bound to the control, either automatically when Auto Binding is enabled or manually using autocompletion.
Default Value	Sets a value to display to users when the form loads and the current control value is empty.
Open	Specifies whether you want the link to open in the current tab or in a new tab.
Anchor	Select this property to enable linking to specific controls in the current form. When you select this property, the Default Value field changes to a drop-down menu that lists names of all controls present in the form. Select a control name to which you want to link. All Basic Palette controls, Money, Phone, Image, and Video controls, if present in the current form, appear in the Default Value field. However, controls within repeatable sections and tables do not appear in the drop-down list.
Hide	Select this check box to hide the control. For example, you might hide a control by default, but configure another control that when selected triggers an event that displays the hidden control.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Advanced Controls

The Advanced palette contains a set of advanced controls that you can use in your forms for handling complex data.

When you select a control on the canvas, the **General** and **Styling** tabs displays in the Properties pane so you can configure settings specific to the selected control.

Topics:

- [Configure Currency \(Money\) Fields](#)
- [Configure Phone Number Fields](#)
- [Include Images](#)
- [Include Videos](#)
- [Place Controls in Panels, Sections, or Tabs](#)
- [Configure Tables](#)
- [Configure Repeatable Sections](#)
- [Configure Train Controls](#)
- [Configure Divider Controls](#)


Configure Currency (Money) Fields

Add a money control to your form to allow users to enter money amounts (USD, EUR, JPY, GBP, and INR). By default, the currency type is set to **USD**. The user can change the currency

type using the **Currency** property on the **General** tab. The corresponding currency symbol displays next to the currency amount. Users can use commas or decimal point to enter different amounts into the control, such as £30,700.00. If users don't specify commas or decimal point, the form automatically displays these symbols. The form rounds all currency amounts to two decimal places.

To configure a money control:

1. From the Advanced palette, drag the **Money** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Currency	Allows you to change the currency type.
Default Value	The default value that will appear in the control until any currency value is entered by the user. If the user entered currency value is removed in the control, the value specified in this field reappears.
Hint	Hint text that will display to users when a user clicks into the control.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Min	Specify the minimum amount that users need to enter into the control.
Max	Specify the maximum amount that users can enter into the control.
Show up/down buttons	Select this property to display the up and down arrow buttons used to increment or decrement the control value.
Step	Specify a step value based on which the amount will be incremented or decremented correspondingly when a user increments or decrements the amount in the control. By default, the step value in this field is set to 1. For example, if the step value specified is 3 and the initial amount is \$ 0.00, then, when a user increments the amount in the control for the first time, the amount is updated to \$ 3.00. When the user increments the amount again for the second time, the amount is updated to \$ 6.00 and so on.
<div style="border: 1px solid #ccc; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Note</p> <p>This property is displayed only when you select the Show increment/decrement buttons check box.</p> </div>	
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Autocomplete	Select this property to display autocompletion entries when users begin to enter text into the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.

Field	Description
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.
Hide	Select this property to hide the control.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Phone Number Fields

Add a phone control to your form to allow users to enter phone numbers in international or US formats. The phone control uses the US format by default and displays the expected phone number pattern (xxx-xxx-xxxx). You can change the format using the **Format** property on the **General** tab.

To configure a phone control:

- From the Advanced palette, drag the **Phone** control onto the canvas.
- Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Format	Specify a phone number format. By default, the US format is selected.
Default Value	Specify the default value that will appear in the control when the form loads.
Placeholder	Value that will appear in the control until any value is entered by the user. If value is removed from the control, the value specified in this field reappears.
Hint	Hint text that will display to users when a user clicks into the control.
Help	Help text that will appear for the control. If you specify help text, a help icon appears next to the name of the control and displays the help text when a user hovers over it or clicks it.
Max Length	Defines the maximum number of characters a user can enter into the control.
Min Length	Defines the minimum number of characters a user must enter into the control. Leave this field blank if you don't want to define the minimum length.
Required	Select this property to specify that users must provide a value in order to successfully submit the form. An asterisk displays to indicate that it's a required field.
Autocomplete	Select this property to display autocompletion entries when users begin to enter text into the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Auto Focus	Select this property to automatically focus on this control when the form loads. If multiple controls are set to Auto Focus, the focus goes to the first control in the form.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.

Field	Description
Hide	Select this property if you want to hide this control when the form loads.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Include Images

Use an image control to include an image in your form. Under image source properties, you can identify the image using an absolute or a relative image URL or a Base64 format string.

To configure an image control:

- From the Advanced palette, drag the **Image** control onto the canvas.
- Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
From URL	To specify either an absolute or relative URL, select this option and enter the image URL in the field below it.
From Base64	To specify an image converted to base64 format, select this option, then identify the image's format in the Image Format field and its binding in the Base64 Binding field. Base64 images can be viewed in preview mode only.
Alternative text	Enter text to display if the image can't be loaded.
Hide	Select this property to hide the control.

- Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Include Videos

Use a video control to add a video, such as a YouTube video, to your form. You can specify full video URLs, embedded URLs, or shortened URLs using the **Source Url** property on the **General** tab. You can optionally loop the video or specify to automatically start playing the video when loaded.

To configure a video control:

- From the Advanced palette, drag the **Video** control onto the canvas.
- Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the name that displays to users.
Binding	Defines a link between the control and a data attribute.
Source Url	Specify a valid source URL for the video.

Field	Description
Allow Fullscreen	Set this property to allow users to play the video in full screen mode. By default, this property is enabled.
Loop	Set this property to loop the video continuously.
Auto Play	Set this property to automatically start playing the video when the form loads.
Show Controls	Specify whether to display play or pause controls for the video. By default, this property is enabled.
Hide	Select this property to hide the control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Place Controls in Panels, Sections, or Tabs

You can add panels, sections, and tabs to your form and use them to group multiple controls under a single control.

To configure a panel, section, or tab control:

1. From the Advanced palette, drag and drop a **Panel**, **Section**, or **Tab** control onto the canvas.
 - A text *Drop Elements to this panel!* indicates that you can drop controls into the panel.
 - A text *Drop Elements to this section!* indicates that you can drop controls into the section.
 - A text *Drop Elements to this tab!* indicates that you can drop controls into the tab. By default, the tab control displays one tab (Tab1). Click **Add tab** to insert additional tabs into the control.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	The title of the panel, section, or tab that the user will see in the form.
Description	For a panel control, provides additional information or instructions for the user.
Type	For a section control, sets the style and format in which the section label displays.
Hide	Select this property to hide the control.
Read Only	Set this property to make the (panel, section, or tab) control read-only. Note that when this property is set, all the controls inside a panel, section, or tab control become read-only (irrespective of the individual property of each of the controls). The user can view but is not allowed to edit the control.
Lazy Loading	For a section control, implements lazy loading for the controls inside it. The lazy loading controls are collapsed by default.
Expanded	For a section control, specifies if the section control is expanded when the form loads. By default, this field is checked.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).
4. Drag and drop individual controls from the Basic Palette or the Advanced Palette into the panel, section, or tab control.

See [Position Controls on Forms](#).

5. Configure general and styling properties for the controls inside your panel, section, or tab control.

Note

If you delete the panel, section, or tab control, all the controls grouped under it get deleted.

Configure Tables

Use table controls to group multiple controls in a grid pattern into your form.

To configure a table control:

1. From the Advanced palette, drag and drop a table control onto the canvas.
By default the table control contains one column.
2. Select the control and edit its properties on the Properties pane's **General** tab.

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	The table title that users will see in the form.
Binding	Defines a link between the control and a data attribute.
Empty State Message	Specify the message that will display when the table is empty.
Max Rows	Specify the maximum number of rows that the table can include. By default, this value is set to 200.
Min Rows	Specify the minimum number of rows that the table can include.
Users can Add/Remove Rows	Allows users to add or remove rows to the table.
Multiple Selection	Allows users to select multiple rows of the table.
Keep Header Visible	If selected, the table header will scroll, so that it is always visible.
Hide	Select this property to hide the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.
Columns	In the Columns field, click Add to add columns. Edit the labels of each column in their respective label fields. Click the Delete icon to delete a column.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#). You can specify column width in several ways:
 - Leave the **Automatic column size** checkbox selected so the table's columns automatically resize to fit the device on which the form is viewed.
 - Deselect the **Automatic column size** checkbox and specify column sizes for different device sizes by entering a number from 1 to 12 in the column size fields that display for different devices.
 - In the **Table Columns Width** fields, specify an absolute width for each column such as 2in (inches), 5cm (centimeters), 100px (pixels), or 25% (percent). If the column widths

together exceed the table width, a scroll bar displays, unless columns are defined with percentages. If percentages are specified but exceed 100%, column widths are displayed proportionally across the table.

4. Drag and drop individual controls from the Basic or Advanced Palette into the columns. Each column can have one control.
5. Configure general and styling properties for the controls inside your table control.

Note

If you delete the table control, all the controls grouped under it get deleted.

Configure Repeatable Sections

Use repeatable section controls to display multiple copies of a set of controls in your form.

To configure a repeatable section control:

1. From the Advanced palette, drag and drop a repeatable section control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the control title that displays to a user.
Binding	Defines a link between the control and a data attribute.
Label Binding	Allows you to define a dynamic label. To do this, you can use a data attribute listed in the Data definition pane.
Max Rows	Specify the maximum number of rows that the control can include. By default this value is set to 200.
Min Rows	Specify the minimum number of rows that the control can include.
Users can Add/Remove Rows	Allows users to add or remove rows to the repeatable section.
Multiple Selection	Allows users to select multiple rows of the repeatable section.
Hide	Select this property to hide the control.
Disabled	Select this property to display the control as inactive (grayed out) to users.
Read Only	Select this property to display the control's content (if any) without allowing users to edit it.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).
4. Drag and drop individual controls from the Basic or Advanced Palette into the repeatable section control.
5. Configure general and styling properties for the controls inside your repeatable section control.

Note

If you delete the repeatable section control, all the controls grouped under it get deleted.

Configure Train Controls

If you have a multistep form activity that users should approach in a particular sequence, then add a train control to guide them through this activity. The train control displays the number of steps a user must complete before submitting the form, and it also indicates the user's current place within a multistep activity.

In a multistep form, you can represent each step using a train stop. Users can navigate between steps by clicking on the train stops; the current stop of the user is highlighted in blue and the visited stops bear a check mark.

To configure a Train control:

1. From the Advanced palette, drag the **Train** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you will use to identify the control.
Label	Specifies the control title that displays to a user.
Binding	Defines a link between the control and a data attribute.
Hide	Select this property to hide the control.
Stretch	Select this property to allow the control to expand and occupy the full width of its container. If selected, all the stops will fit into a single view and users don't have to scroll to view the stops.
Sorted	Select this property to display the control options in alphabetical order.
Options Source	Specify if the control's options should come from static values you enter or from a list of values data attributes. <ul style="list-style-type: none"> • Static: Specify choices using Options Names and Options Values fields. Use Options Names to specify the label to display for an option and use Options Values to specify an internal value for an option. • From Data: In the Options List field, select a list of values options source from the data definitions available in the web form. If you selected a list of complex elements, then, in the Label Binding field, specify a data attribute that will display as the label and in the Value Binding field, specify a data attribute that will be the value.
Default Value	If you selected Static , then specify a default option in this field. If you selected From Data , then select either the first or the last value as the default value.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).

Configure Divider Controls

Use a divider control to separate related content within your form or to create logical breaks in your form's layout. For example, you might want to break up your form into sections or separate a related group of controls from the rest in the form.

To configure a divider control:

1. From the Advanced palette, drag the **Divider** control onto the canvas.
2. Select the control and use the **General** tab in the Properties pane to configure the following:

Field	Description
Name	An internal identifier that you use to identify the control.
Hide	Select this property to hide the control.

3. Use the **Styling** tab in the Properties pane to modify the control's appearance as needed. See [Styling Properties](#).
 - To specify the color of the divider control, use the **Stroke Color** property.
 - To specify the width of the divider control, use the **Stroke Width** property. You can enter standard values such as 0.05in, 0.2em, or 5px.

Style Form Controls

Styling allows you to customize the appearance of individual controls in your form. Use styling options to define display-specific properties for the currently selected control.

Change the Style of a Form Control

To change the styling of a control:

1. Select the control you want to style.

The Properties pane changes and displays the **General** and **Styling** tabs.
2. Click the **Styling** tab.

This displays various styling options specific to your selected control.
3. Choose and set the desired styling properties. See [Styling Properties](#).

Styling Properties

You can update a control's appearance by configuring its styling properties on the **Styling** tab in the Properties pane. The **Styling** tab provides display-specific properties for the currently selected control.

The following table lists all available properties on the **Styling** tab in alphabetical order.

Note

Styling properties are control-specific. Not all of the properties listed below are available for every control.

Property	Description
Automatic column size	<p>Calculates the column size for the control based on the amount of visible controls in the row. Automatic column size is selected by default.</p> <p>Note: In small devices like a phone, each control is displayed in one row when automatic column size is enabled.</p> <p>You can specify absolute column sizes for different device sizes. To do this, deselect Automatic column size and enter a number from 1 to 12 in the four available options: Small, Medium, Large, and Extra large column sizes.</p>
Background Color	Specifies the background color of the content area in a control.
Border (Color, Width, Style, Radius)	<p>Determines the appearance of the border in the content area of your control.</p> <ul style="list-style-type: none"> • Border Color: Defines the color of the border. • Border Width: Defines the width of the border. Use standard values such as 1in, 5em, or 20px. • Border Style: Defines the style of the border - Solid, Dotted, or Dashed. • Border Radius: Defines the value of rounded border corners. Use standard values such as 1in, 5em, or 20px.
Color	Specifies the color of the text in the content area of a control.
Control Alignment	Specifies the alignment (left, right, or center) of a control in the form.
Font Size	Defines the font size of the text in the content area of a control. The available values are x-small, small, normal, large, and x-large.
Height	Sets an absolute height for the control. Use standard values such as 1in, 5em, or 20px.
Label Color	Specifies the color of a control's label.
Label Size	<p>Specifies the size of a control's label. The available values are x-small, small, normal, large, and x-large.</p> <p>Note: For a Panel control, using the Standard theme overrides the value in the Label Size property.</p>
Reset Inline Styles to Default	Discards all styling selections made on the Styling tab and restore settings to their default values.
Text Alignment	Specifies the alignment (left, right, or center) of the content in a control. This property applies to controls where the user can type in text such as the Input Text control.
Theme	<p>Uses a CSS to define the format of a control. This property applies to Panel and Section controls.</p> <p>For a Panel control:</p> <ul style="list-style-type: none"> • Standard: Automatically increases the size of the control's label to 24 pixels (font size) and makes the label bold (font weight). The Standard theme overrides the value in the Label Size property. • None: Applies no formatting to the control. None is the default value. <p>For a Section control:</p> <ul style="list-style-type: none"> • Indent: Automatically indents the section. With the Indent theme, you can easily nest sections within sections. • None: Applies no formatting to the control. None is the default value.
Width	Sets an absolute width for the control. Use standard values such as 5in, 20px, or 5%.

Bind Form Data with Controls

In forms, data and controls are decoupled. You can define your controls and data independently and then connect the controls to the data using the binding property.

How Auto Binding Works?

If you open the Properties pane of a form, you'll notice that the **Auto Binding** property is selected by default. When the auto binding property is enabled, data attributes are automatically created and are linked to the controls as you drag and drop controls from the palette onto the canvas. The data attributes are listed in the Data pane.

Note

The name of the binding in the **Binding** field is the same as the name of the control to which it is linked when you use auto-binding.

Create Data Attributes Manually and Link Them to Controls

If you disable the **Auto Binding** field, then you must create data attributes in the Data pane and link them to every unbound control using the **Binding** field in the control's Properties pane.

1. In the form's Data pane, click **Add**.
2. In the Create Attribute window, enter a name.

Note

An attribute name can only start with a lower case letter and can only contain letters, digits, and underscores.


3. Under **Data Type**, select **Simple** and then choose a data type from the drop-down menu.
4. Click **Create**.
5. Drag and drop a control from the palette onto the form canvas.
6. Select the control and click the **General** tab in the Properties pane.
7. In the **Binding** field, specify the attribute that you created in the Data pane.


The attribute is successfully linked to the control.

Activate a Form

After designing a form, activate it to use in a user task in a human in the loop workflow.

Prerequisite: Before activating a form, review and fix all errors in the form.

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Human in the loop** .

3. In the Forms section, ensure that the form you want to activate is in the **Configured** state.
 4. Point to the Configured form, click , and then select **Activate**.
- The form is successfully activated and its status changes to **Active**.

5

Manage Workflows and Forms

Version, clone, edit or delete your workflows and forms to keep your workspace up-to-date and organized.

Topics:


- [Version a Workflow or Form](#)
- [Clone a Workflow or Form](#)
- [Deactivate a Workflow or Form](#)
- [Edit a Workflow or Form](#)
- [Delete a Workflow or Form](#)

Version a Workflow or Form

You can version a workflow or form for various purposes.

- **Change tracking:** Allows maintaining a history of changes to your workflow or form to review previous versions or revert when necessary.
- **Support for multiple scenarios:** Enables using different versions to address varying business automation scenarios.
- **Incremental development:** Facilitates updating your workflow or form incrementally as your business requirements evolve over time.

Version a Workflow

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Human in the loop** .
3. In the Workflow section, point to the workflow you want to version, click **⋮**, and then select **Create new version**.
4. In the Create new version panel:
 - a. Update the **Version** field with an appropriate number, and add a description if necessary.
 - b. Click **Version**.



A new version of the workflow is created and listed in the Workflow section.

Version a Form

1. Open a project.
 - a. In the navigation pane, select **Projects**.

Edit a Workflow or Form

You can edit a workflow or form at any time when it's in the **Draft** or **Configured** state. However, a workflow or form in the **Active** state is view-only. To edit an active workflow or form you must either deactivate or version it.

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Human in the loop** .
3. In the Workflow or Form section, point to the workflow or form you want to edit, click  and then select **Edit**.



For a workflow, the workflow designer opens where you can edit the workflow as required.

For a form, the form designer opens where you can edit the form as required.

Delete a Workflow or Form

If you don't need a workflow or form anymore, you can delete it from your project.

You may require to delete a decision model in the following scenarios:

- **End of lifecycle:** Delete workflows or forms that are outdated and no longer relevant.
 - **Project organization and cleanup:** Clear out unused workflows or forms to maintain a clean and efficient project workspace.
 - **Presence of duplicate or incorrect workflows or forms:** Delete incorrectly setup or redundant workflows and forms to optimize management of these artifacts.
1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
 2. In the left toolbar, select **Human in the loop** .
 3. In the Workflow or Form section, point to the workflow or form you want to delete, click , and then select **Delete**.
 4. In the Delete dialog, click **Delete**.

Note

- Deleting a form that is currently used in one or more workflows may result in workflow errors.
- Deleting a workflow that is currently used in one or more integrations may result in runtime errors.

6

Use Workflows with Agentic AI

When you need to add human oversight at any point in an agentic AI automation solution, use human in the loop workflows.

Before you start, review the prerequisites for using human in the loop in Oracle Integration. See [Prerequisites for Using Human in the Loop](#).

Check out the overall guidelines to use human in the loop for agentic AI in Oracle Integration. See [Human in the Loop for Agentic AI in Oracle Integration](#). Then, dive deep and learn how to design integrations that call workflows, how to create agentic AI tools with those integrations, and finally how to complete assigned human tasks.

Topics:

- [Design an Integration to Call a Workflow](#)
- [Create an Agentic AI Tool Using the Integration](#)
- [Complete Assigned Task as a User](#)

Design an Integration to Call a Workflow

To use a human in the loop workflow with Agentic AI, you must create an integration that calls the workflow.


Prerequisites:

- Create a project. See Create or Import a Project in *Using Integrations in Oracle Integration 3*.
- Create a human in the loop workflow. There must be at least one user task in the workflow with an associated form and a user assigned to the task. The workflow and form must be in **Active** state. For details, see the following:
 - [Model Workflows](#)
 - [Add and Configure User Tasks](#)
 - [Design Forms](#)
- Create a REST trigger connection with the REST adapter. The REST connection must be configured with the **Trigger** role and use **OAuth 2.0 Or Basic Authentication** as the security policy. See Create a Connection in *Using the REST Adapter with Oracle Integration 3*.


Before you create the integration, keep in mind the following:

- The integration must be of *asynchronous* type, that is, the integration's endpoint does not receive a response.
- The integration must always start with a REST trigger connection.
- The REST trigger connection must be configured with the following:
 - POST verb
 - JSON request payload

Design an Integration to Call a Workflow

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. Create an integration.
 - a. In the left toolbar, select **Integrations** .
 - b. In the **Integrations** section, click **Add** (if no integrations have been designed) or+ (if one or more integrations have been designed).

The Add integration panel appears.
 - c. Click **Create** and then select **Application**.
 - d. Enter a name for the integration and provide additional descriptive information, if needed.
 - e. Click **Create**.



The integration canvas opens.
3. Configure the REST trigger connection that starts the integration.
 - a. Click  and in the resulting dialog select the REST trigger connection that you have created before.


The Configure Basic Info page displays.
 - b. Enter a meaningful name in the **What do you want to call your endpoint?** field and click **Continue**.
 - c. In the Configure Resource Configuration page, enter the following details and click **Continue**.
 - i. **Provide an operation name:** Enter the operation name.
 - ii. **What does this operation do?:** Provide a brief description of the purpose of the operation.
 - iii. **What is the endpoint's relative resource URI?:** Enter the endpoint's relative path.
 - iv. **What action do you want to perform on the endpoint?:** Select **POST**.
 - v. **Add and review parameters for this endpoint:** Select this option.
 - vi. **Configure a request payload for this endpoint:** Select this option.
 - d. In the Configure Request Parameters page, add a request parameter and select its data type.

Note

This parameter specifies the call back URL which the AI agent uses to get the response/update after the human approval task completes.
 - e. Click **Continue**.
 - f. In the Configure Request page, enter the following and click **Continue**.
 - i. Select the request payload format as JSON.




- ii. Click **<<inline>>** to copy and paste the JSON payload and click **OK** when complete.
 - g. In the Summary page, click **Finish**.
4. Add a **Human in the loop** action to the integration to call a workflow.

- a. Click **Actions**  to open the Action panel.
- b. From the Call section on the panel, drag the **Human in the loop** action and drop it onto the  sign that appears when you hover over a connection arrow between elements.

Alternatively, hover over a connection arrow, click the  sign that appears, and select the **Human in the loop** action from the dialog box.

- c. In the Configuration Wizard that appears, configure the human in the loop action.
 - i. On the Configure Basic Info page, provide a unique name and description in the following fields.
 - **What do you want to call your endpoint?:** Enter a unique name.
 - **What does this endpoint do?:** Provide a description.
 - ii. Click **Continue**.
 - iii. On the Configuration page, complete the following.
 - From the **Process** drop-down, select the workflow that the integration calls.
 - From the **Version** drop-down, select the appropriate version of the workflow selected.
 - iv. Click **Continue**.
 - v. On the Summary page, review the data and click **Finish**.
- d. Click **Continue**.

The human in the loop element is added to the integration.

5. Add a **Map** action before the **Human in the loop** action to pass information from the integration to the human in the loop workflow.
- a. Click **Actions**  to open the Action panel.
 - b. Drag a **Map** action and drop it onto the  sign that appears when you hover over the connection arrow before the human in the loop action.
 - c. Select the map action, click  and select **Edit**. The mapper opens.
 - d. In the Sources list on the left, expand the tree until you find the data element that you need to pass to the workflow.
 - e. In the Target list on the right, expand the tree until you find the input that needs to receive the data.
 - f. Drag the source element to the target element.
- For more information on mapping, see Map Data in *Using the Oracle Mapper with Oracle Integration 3*.
- g. After you've completed mapping elements, test your mapping.
- See Test Your Mappings in *Using the Oracle Mapper with Oracle Integration 3*.


6. Activate the integration.

After you've completed designing the integration, activate it so that you can use it to create an agentic AI tool. See [Activate an Integration in *Using Integrations in Oracle Integration 3*](#).

Create an Agentic AI Tool Using the Integration


After you have designed an integration that calls a workflow, you can use the integration to create an agentic AI tool. AI agents will use this tool to invoke the human approval workflow.

Prerequisite: You have designed an integration that calls a human in the loop workflow. See [Design an Integration to Call a Workflow](#).

1. Open a project.
 - a. In the navigation pane, select **Projects**.
 - b. Select the project name.
2. In the left toolbar, select **Integrations** .
3. In the **Integrations** section, select the integration that you want to use as an agentic AI tool.

Note

The integration must be in **Active** state. If it is not Active, activate the integration. See [Activate an Integration in *Using Integrations in Oracle Integration 3*](#).

4. Click  and select **Create agentic AI tool**.
The Create Tool panel appears.
5. In the Create Tool panel, enter the following details.

Field	Description
Name	By default, the name is automatically populated from the integration name. If required, edit the name. Note that you cannot change the name once the tool has been created.
Identifier	By default, the identifier is automatically populated from the tool name.
Description	By default, automatically populated from the integration description.

6. Click **Create**.
The Tool details page is displayed. The tool lists the description, guidelines, input parameters.
7. If required, enter additional information for the tool and identify which parameters are sent to the LLM.
8. Click **Save**.

In the project workspace, click **Agentic AI**  in the left toolbar. In the **Tools** section view the tool that you successfully created.

An AI agent can use this tool to invoke the human in the loop workflow and thus incorporate human oversight.

To continue working on the AI agent, see the following:


- Build the AI Agent in *Using Agentic AI in Oracle Integration 3*.
- Run and Test the AI Agent in *Using Agentic AI in Oracle Integration 3*.

Complete Assigned Task as a User

When the human in the loop workflow is invoked by an AI agent, the human task in the workflow appears in the assigned user's task list in Process Automation Workspace.

Prerequisite: To complete the task, the assigned user needs access to Process Automation Workspace.



1. Sign in to Process Automation Workspace.
2. Alternatively, you can also go directly to Process Automation Workspace from Oracle Integration.


- a. In the left toolbar of your project workspace, click **Human in the Loop** .
- b. Click the **Observe** tab.

Workflow instances in various states are displayed.

- c. Search for the workflow that was invoked by the AI agent. The workflow should be in **In progress** state.

If there are several workflow instances, you can search for your instance by using one of the following options:

- Click the Search icon  and enter any of the following in the search field: workflow name, task title or created by.
- Click the Filter icon  and narrow down your search by specifying the workflow's status such as **In progress**, date range and time when it was created, the instance name, and so on.

- d. Select the workflow and click **View details** .

The Activity stream panel opens. The user task's status shows In progress indicating that the user hasn't completed the task.

Activity stream [Close]

Purchase Order Request: 4 Desk Chairs & Coffee Machine for San Francisco Office - Total: \$3200. (1.0.1)

Instance id: 848fe2bb-8887-11f0- [Redacted] [Refresh] [Share]

[Open in workspace](#) [External Link Icon]

Wed Sep 3, 11:03:27 IST 2025, triggered by F49F6D213E [Redacted]

Start event - DirectApproval

Wed Sep 3, 11:03:28 IST 2025

Purchase Order Request: 4 Desk Chairs & Coffee Machine for San Francisco Office - Total: \$3200. [In progress]

Human task • User task

Completed a few seconds ago by l1se [Redacted]

If you click the expand arrow on the user task, you can view the details of the task and the input payload associated with it. As the task hasn't been acted upon by the user, the output payload will be empty.

- e. Click the **Open in workspace** URL.
The Process Automation Workspace opens in another browser tab.
- 3. As a user assigned the task, access the task in Workspace.
 - a. In the Workspace navigation menu, select **Tasks**.
The My Tasks page appears.

ORACLE Workspace

Welcome back , l1 service admin!
You have 50 open tasks

My Tasks Team Tasks [View More]

<input type="checkbox"/>	Title	Process name	Reference Id	Created Date	Due Date	Status	Actions
<input type="checkbox"/>	Purchase Order Request	POAPPWF	Not Set	Sep 25, 2025 9:11 PM	Not Set	Assigned	⋮
<input type="checkbox"/>	Purchase Order Request	POAPPWF	Not Set	Sep 25, 2025 7:40 PM	Not Set	Assigned	⋮
<input type="checkbox"/>	Request for PO v1	StockApproval	Request for PO desc	Sep 25, 2025 4:57 PM	4 hours ago	Assigned	⋮

- b. Select the task assigned to you and open it to view the form associated with it.
4. Perform necessary actions on the form to complete the task.
For example: In an approval form, review the content of the form, suggest next steps, and approve or submit the form.
5. Review the Activity stream of the workflow after the task completes.

The screenshot displays the 'Activity stream' for a 'Purchase Order Request: 4 Desk Chairs & Coffee Machine for San Francisco Office - Total: \$3200. (1.0.1)'. The instance ID is 848fe2bb-8887-11f0-... and there is a link to 'Open in workspace'. The stream shows three events:

- Start event - DirectApproval**: Occurred on Wed Sep 3, 11:03:27 IST 2025, triggered by user F49F6D213!.
- Human task - User task**: Completed on Wed Sep 3, 11:03:28 IST 2025. The task title is 'Purchase Order Request: 4 Desk Chairs & Coffee Machine for San Francisco Office - Total: \$3200.'. It is marked as 'Complete' and was completed a few seconds ago by user l1sei.
- End event - DirectApproval**: Occurred on Wed Sep 3, 11:06:19 IST 2025.

Click the expand arrow on the user task to view the task details. In the **Details** tab, you can view the task history, such as who approved the task and when. In the **Payload** tab, you can view both the task input and output payloads now. In the output payload you can see the response that was sent back to the AI agent.

You've successfully completed the task assigned to you. Now, agentic AI will perform the next action based on the callback response received in the task's output payload.