

PUBLIC

# **Build a Workflow from Scratch with SAP Cloud Platform Workflow Management**

## **DEV163**

Exercise 1 | Workflow  
Archana Shukla / SAP

## TABLE OF CONTENTS

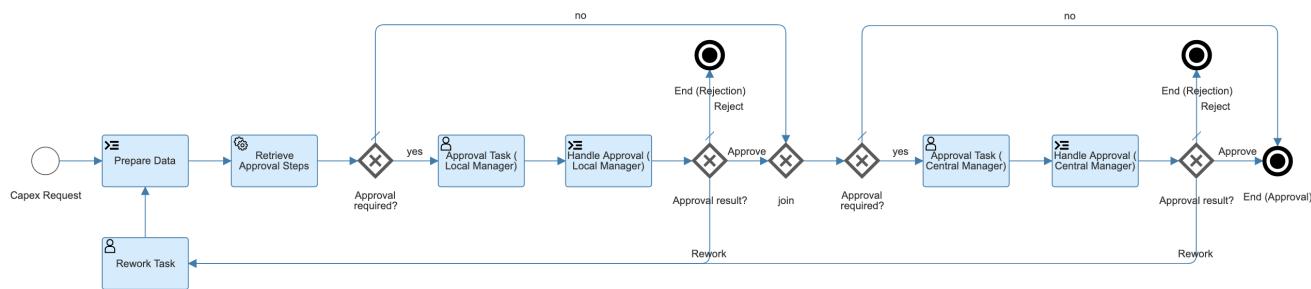
<b>OVERVIEW .....</b>	<b>3</b>
<b>MODEL WORKFLOW USING BUSINESS APPLICATION STUDIO.....</b>	<b>4</b>
<b>BUILD AND DEPLOY WORKFLOW .....</b>	<b>46</b>
<b>EXECUTE AND MONITOR WORKFLOW.....</b>	<b>49</b>
<b>SUMMARY .....</b>	<b>54</b>

## OVERVIEW

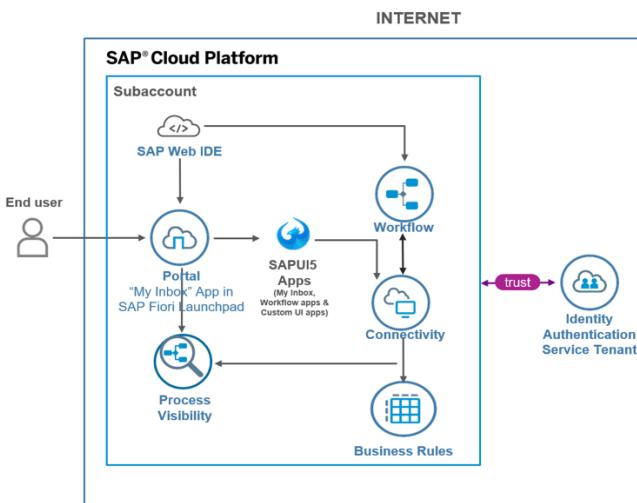
Estimated time: 90 minutes

Capital Expenditure process starts when any company receives a funding request for capital expenditures. These requests are then thoroughly reviewed, evaluated and finally approved or rejected based on the available budgets for the current period. Approval is not straight forward. There are many rules that are involved to determine how many levels of approvals will be there or who will approve it based on the investment details.

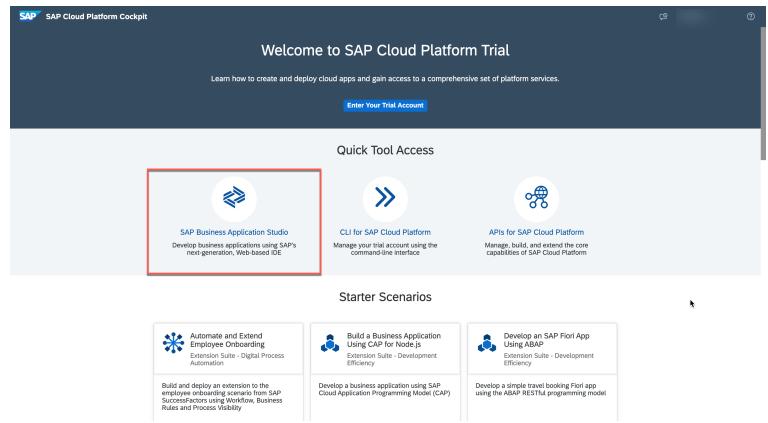
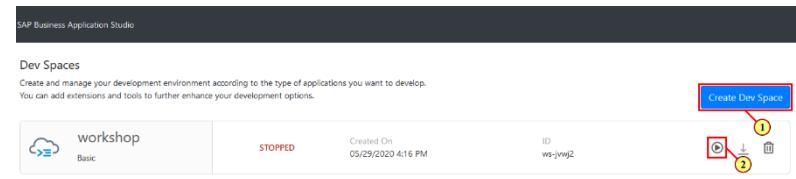
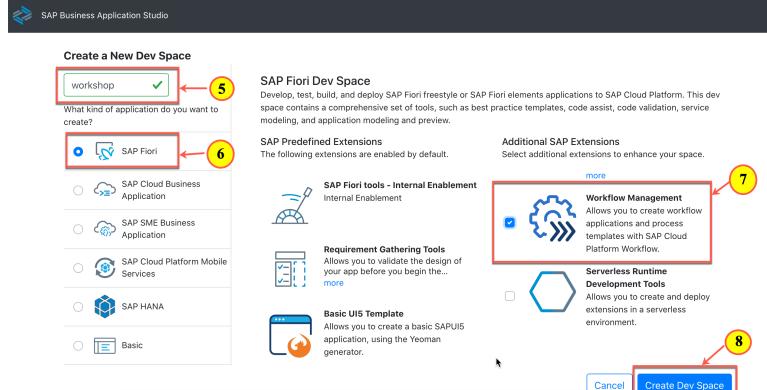
There are many challenges with capital expenditure process and companies are always looking for managing investment decisions in a transparent way. Automating capital expenditure approval process enables customer to accelerate investment decisions and to bring the agility into every step and decision such that the corporate policies are followed while taking the investment decisions.

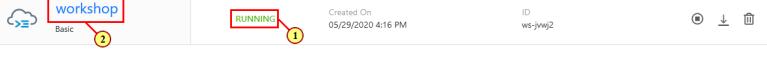
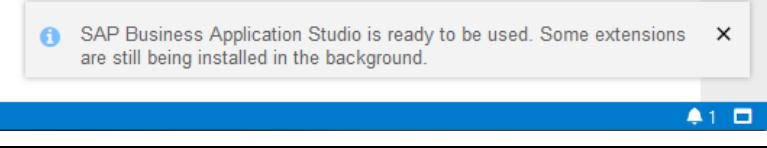
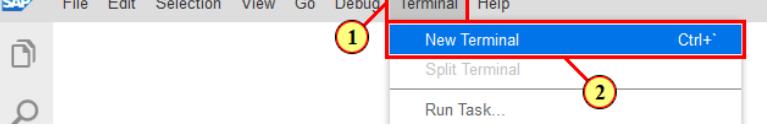
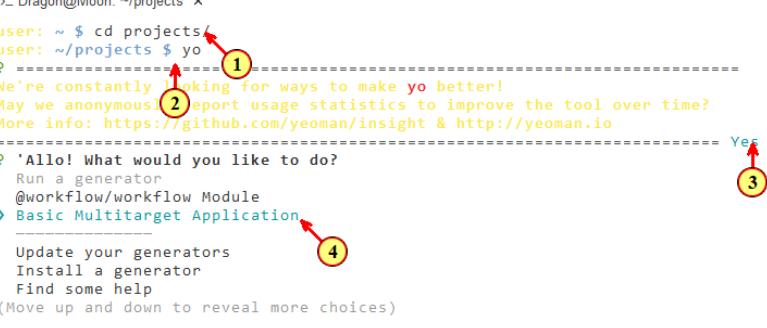
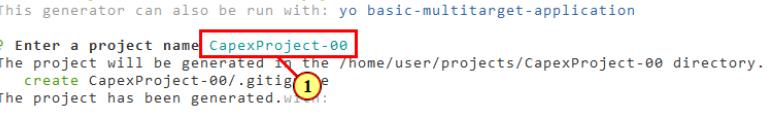


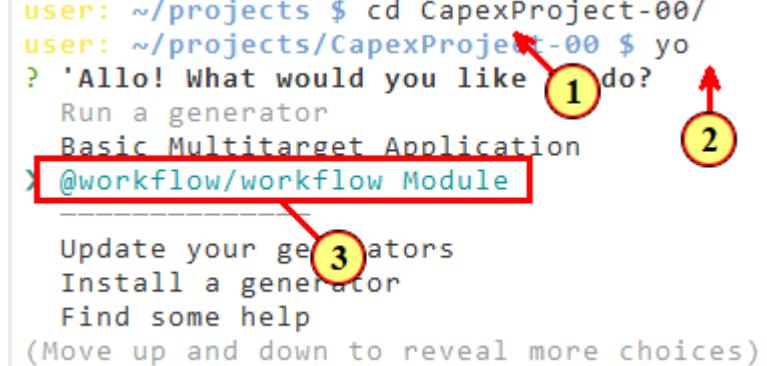
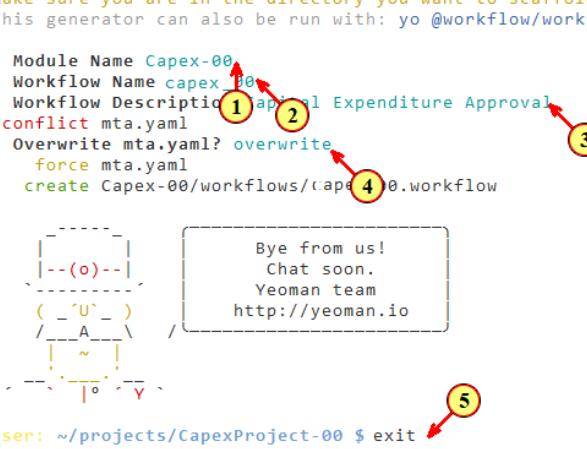
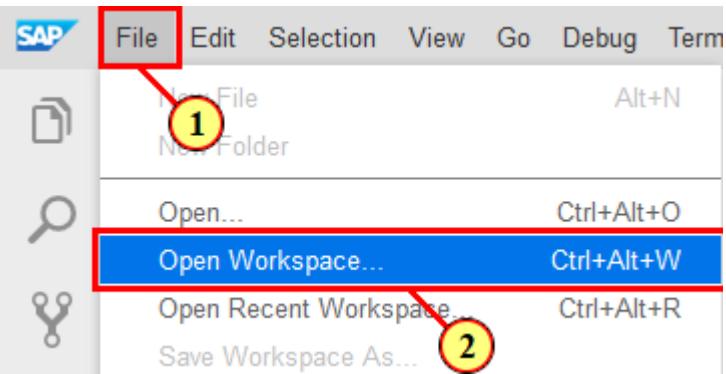
The below solution diagram explains how customer can automate CAPEX scenario using SAP Cloud Platform.

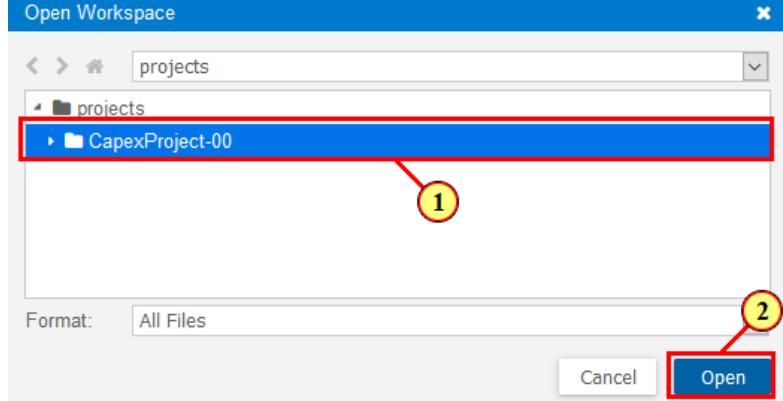
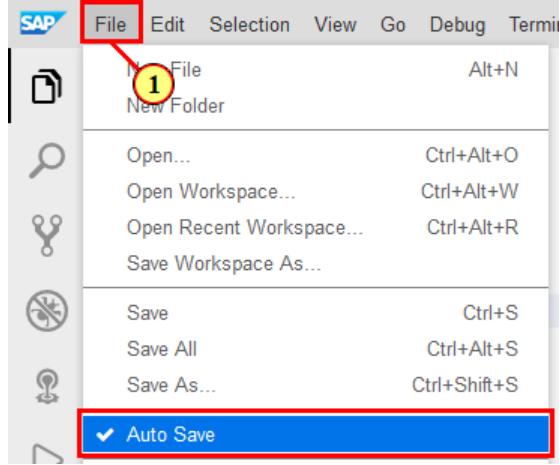
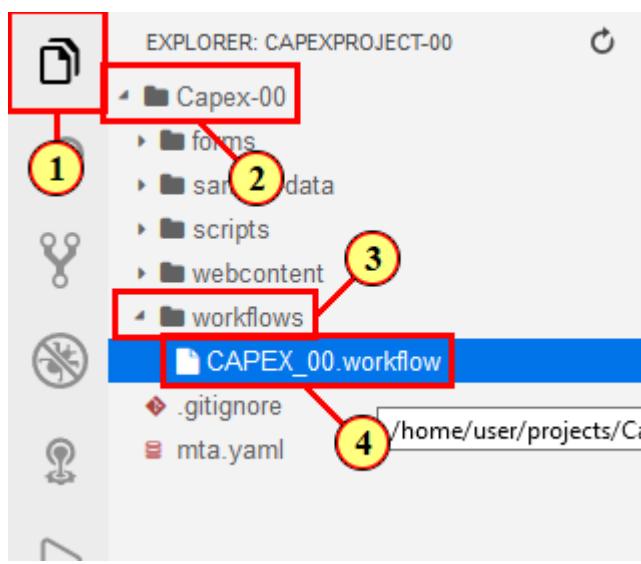


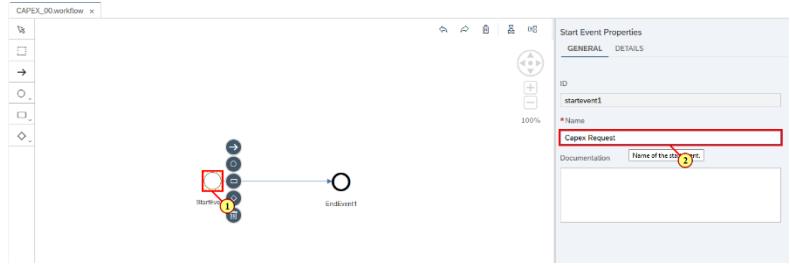
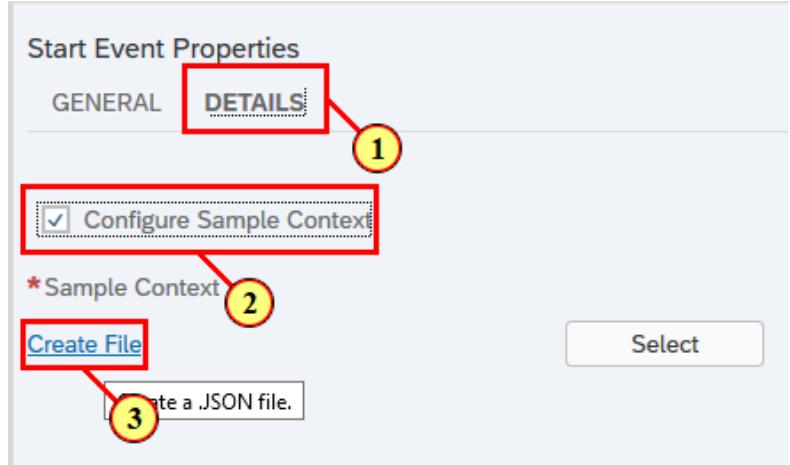
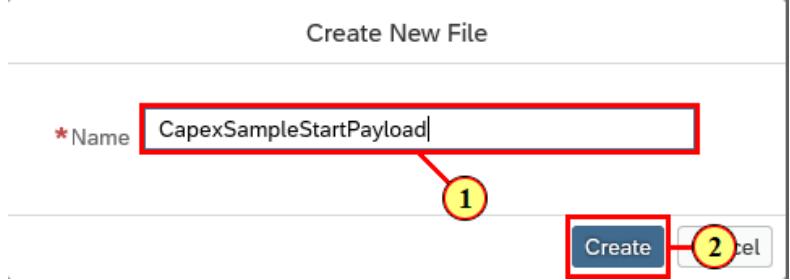
## MODEL WORKFLOW USING BUSINESS APPLICATION STUDIO

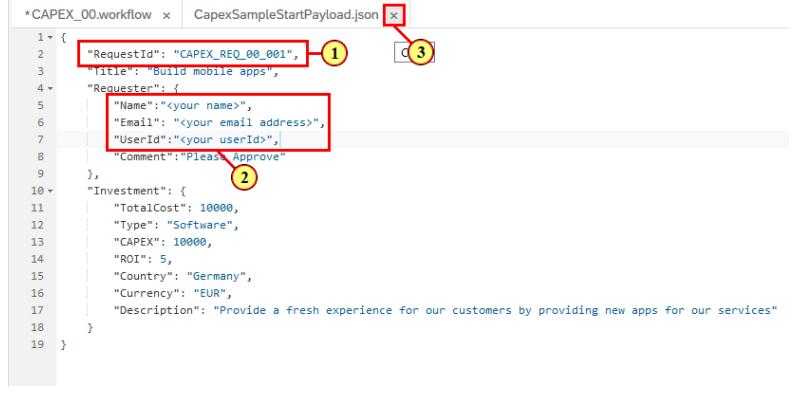
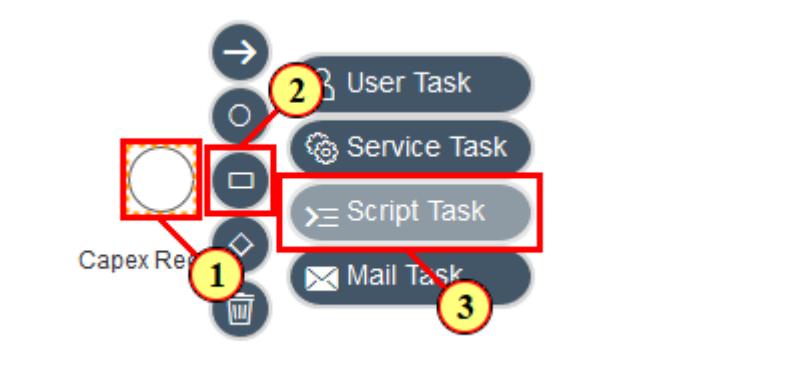
Explanation	Screenshot
<p>1. Logon to <b>SAP Cloud Platform Trial Home Page</b> with your trial username and password.</p> <p><a href="https://cockpit.hanatrial.ondemand.com/cockpit/#/home/trial">https://cockpit.hanatrial.ondemand.com/cockpit/#/home/trial</a></p> <p>2. Choose <b>SAP Business Application Studio</b>.</p>	
<p>Each SAP Business Application Studio Development Space is a virtual machine hosting your project files, securely separated from other spaces.</p> <p>3. If you have no dev space, click the <b>Create Dev Space</b> and follow instructions in step 5.</p> <p>4. Else select <b>PLAY</b> to start the space if you see the status as <b>STOPPED</b>.</p> <p>Note: Idle dev space will be shut down after some time. This is indicated by a yellow bar in the bottom of the screen with the text "Offline". In this case, return to this Dev Space selection screen from your bookmark and restart it.</p>	
<p>5. Enter <b>workshop</b> as the new <b>Dev Space name</b>.</p> <p>6. Ensure <b>Application Type</b> is selected as <b>SAP Fiori</b>.</p> <p>7. Select the extension <b>Workflow Management</b>. Note: This extension will enable the workflow modelling features.</p> <p>8. Select <b>Create Dev Space</b>.</p>	

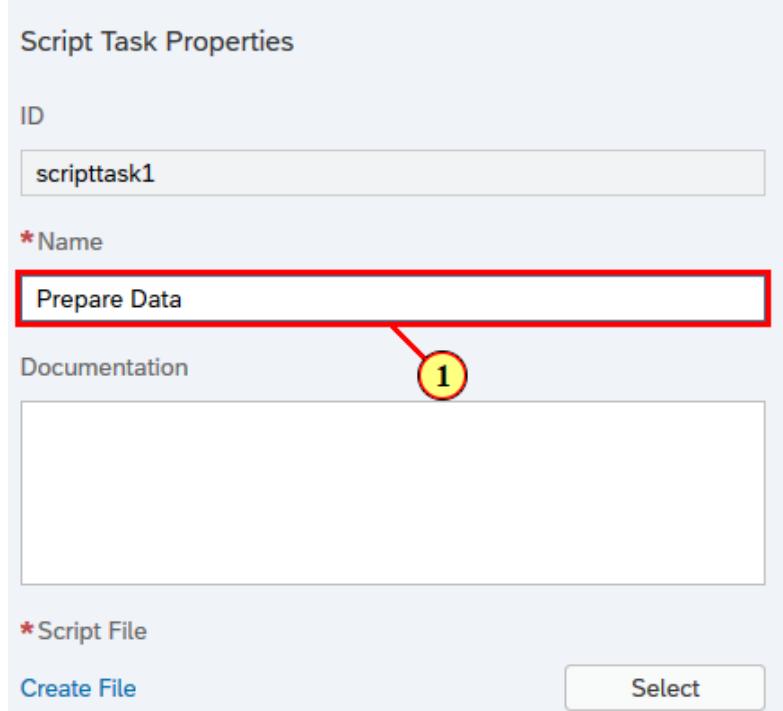
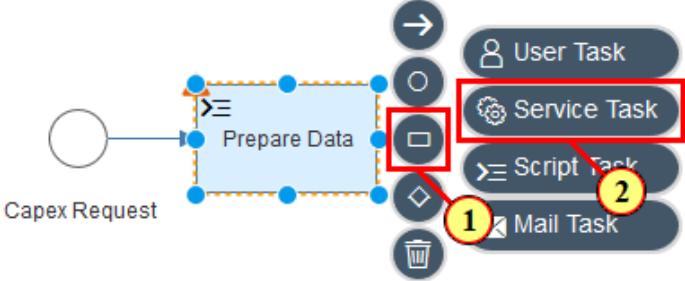
Explanation	Screenshot
<p>9. Wait until the virtual machine providing this dev space is in status <b>RUNNING</b>.</p> <p>10. Open the <b>SAP Business Application Studio</b> by clicking on the dev space name.</p>	
<p>Note: SAP Business Application Studio is a web application that loads the features of the extensions. There might be a short delay for all features to be operational when first accessed.</p>	
<p>11. From the menu, select <b>Terminal</b>.</p> <p>12. Select <b>New Terminal</b> to open a Linux Terminal window.</p>	
<p>13. Enter <b>cd projects/</b> in the <b>Terminal</b> window to go into the folder where you are going to create the workflow module. Note: This project folder is already created.</p> <p>14. Enter <b>yo</b> in the terminal window to start Yeoman.</p> <p>Note: Yeoman is a tool for quickly setting up the basic structure of projects.</p> <p>15. If asked, choose whether you want to enable telemetric for Yeoman.</p> <p>16. Select <b>Basic Multitarget Application</b> by using the up/down arrow keys and press <b>Enter</b>.</p>	
<p>17. Enter <b>CapexProject-00</b> or any name of your choice as <b>Project name</b>.</p>	

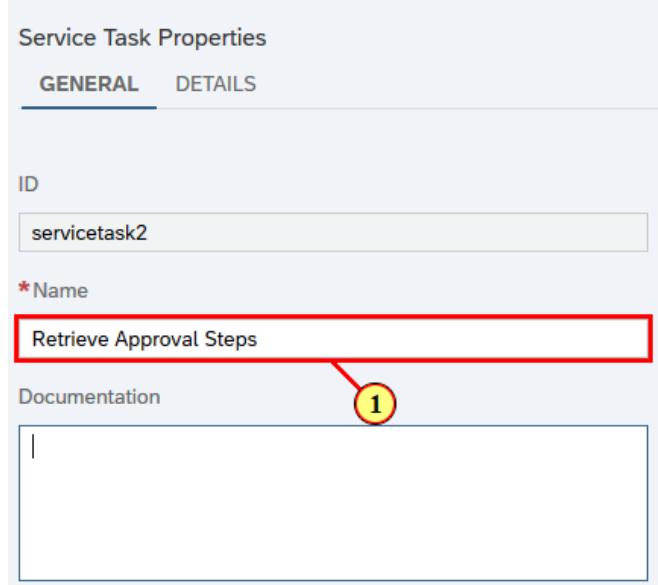
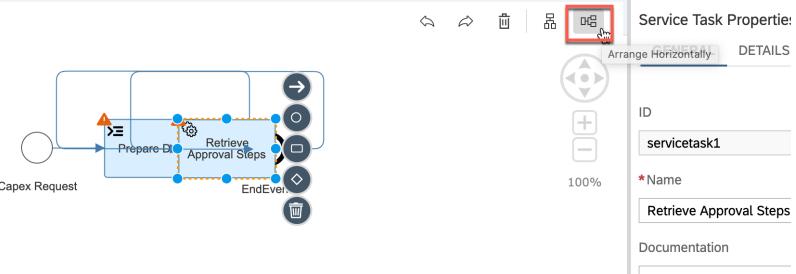
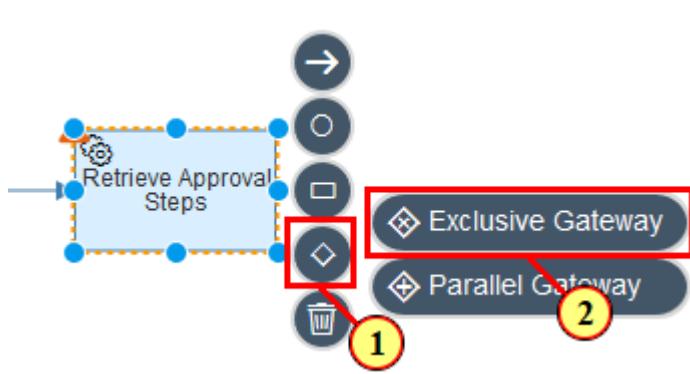
Explanation	Screenshot
<p>18. Change into the newly created directory by entering <b>cd CapexProject-00</b>. You can also use <b>TAB</b> for auto-completion, i.e. entering <b>cd Cap&lt;TAB&gt;</b> will be sufficient.</p> <p>19. Enter <b>yo</b> in the window to start Yeoman again.</p> <p>20. Select <b>Workflow Module</b> by using the up/down arrow keys and press <b>Enter</b>.</p>	 <pre>user: ~/projects \$ cd CapexProject-00/ user: ~/projects/CapexProject-00 \$ yo ? 'Allo! What would you like to do?   Run a generator   Basic Multitarget Application   X @workflow/workflow Module -----</pre> <p>Update your generators Install a generator Find some help (Move up and down to reveal more choices)</p>
<p>21. Enter <b>Capex-00</b> as <b>Module Name</b>.</p> <p>22. Enter <b>Capex_00</b> as <b>Workflow Name</b>.</p> <p>23. Enter <b>Capital Expenditure Approval</b> as <b>Workflow Description</b>.</p> <p>24. Yeoman will warn you that there already is an <b>mta.yaml</b> file. Type <b>Y</b> to overwrite it.</p> <p>25. You can close the <b>Terminal</b> by typing <b>exit</b> or clicking on the X in the window header.</p>	 <pre>Make sure you are in the directory you want to scaffold into. This generator can also be run with: yo @workflow/workflow-module  ? Module Name Capex-00 ? Workflow Name capex_00 ? Workflow Description Capital Expenditure Approval conflict mta.yaml ? Overwrite mta.yaml? overwrite   force mta.yaml create Capex-00/workflows/capex_00.workflow  Bye from us!   Chat soon,   Yeoman team   http://yeoman.io  user: ~/projects/CapexProject-00 \$ exit</pre>
<p>26. From the menu, select <b>File</b>.</p> <p>27. Select <b>Open Workspace....</b></p>	 <p>SAP</p> <p>File Edit Selection View Go Debug Term</p> <p>New File... Alt+N</p> <p>New Folder</p> <p>Open... Ctrl+Alt+O</p> <p>Open Workspace... Ctrl+Alt+W</p> <p>Open Recent Workspace</p> <p>Save Workspace As... Ctrl+Alt+R</p>

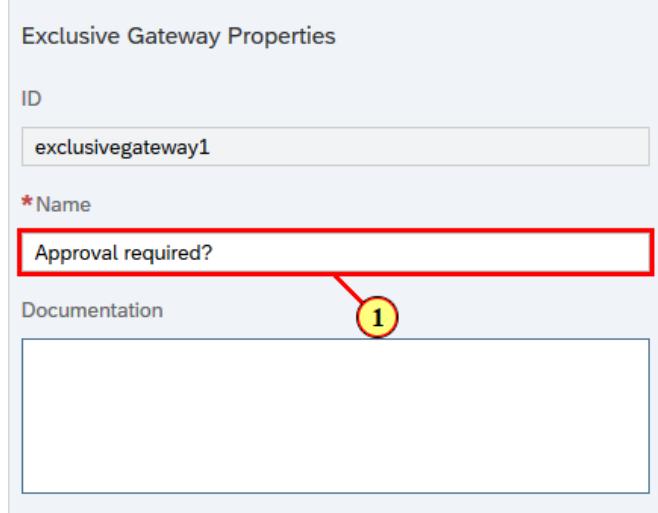
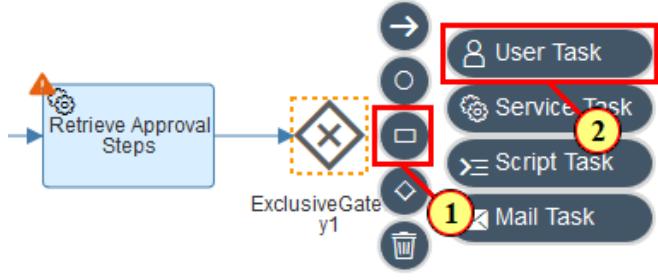
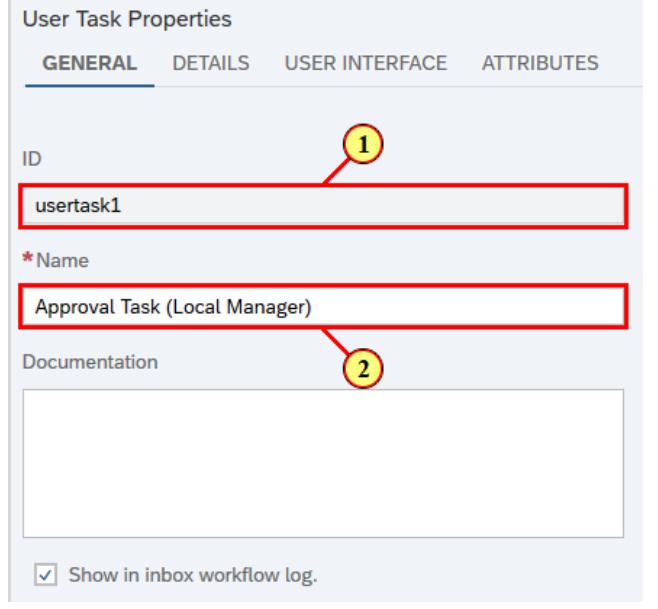
Explanation	Screenshot
<p>28. Select the directory containing your module <b>CapexProject-00</b>.</p> <p>29. Select <b>Open</b>.</p>	
<p>30. From the menu, select <b>File</b>.  Note that there is an <b>Auto Save</b> functionality (enabled by default). If you choose to disable it, remember to manually save your files.</p>	
<p>31. If it is not already open, select the <b>Explorer</b>  from the sidebar.</p> <p>32. Open the <b>Capex-00</b> module by selecting it.</p> <p>33. Open the <b>workflows</b> directory by selecting it.</p> <p>34. Open the <b>CAPEX_00.workflow</b> by double-clicking on it.</p> <p>Note: If the graphical workflow editor is not opened, or a textual (code) editor is opened instead, the workflow plugin did not load correctly. Refresh your browser window and try again after SAP Business Application Studio has fully loaded.</p>	

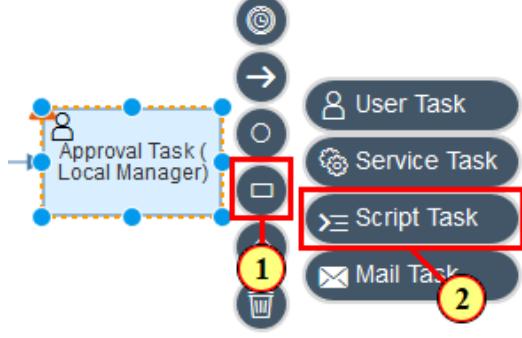
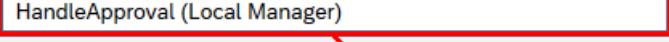
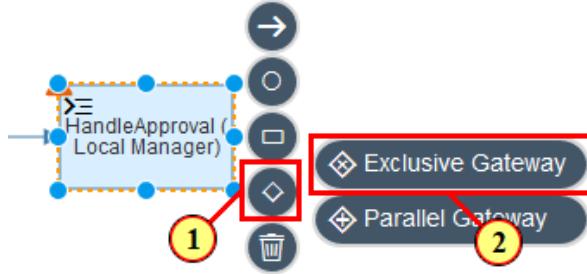
Explanation	Screenshot
<p>A new workflow with a default Start and End event is created.</p> <ul style="list-style-type: none"> <li>○ On the left side there is a toolbar for adding new flow elements.</li> <li>○ On the right side is the properties pane for the currently selected element.</li> </ul> <p>35. Click on the <b>Start Event</b>  to select it. This will show the quick action icons and the property pane will show the details of this element.</p> <p>36. Enter <b>Capex Request</b> as the <b>Name</b> of the start event.</p> <p>This will later be shown in the workflow <i>Execution Log</i> as well as in <i>Process Visibility</i>.</p>	
<p>37. Click on the tab <b>Details</b> of the start event.</p> <p>38. Select <b>Configure Sample Context</b>. When starting a new workflow, an initial context (JSON data) can be given. With this option, an example context can be configured.</p> <p>39. Click <b>Create File</b>.</p>	
<p>1. In the popup, enter <b>CapexSampleStartPayload</b> as <b>File Name</b>.</p> <p>2. Click <b>Create</b>.</p>	

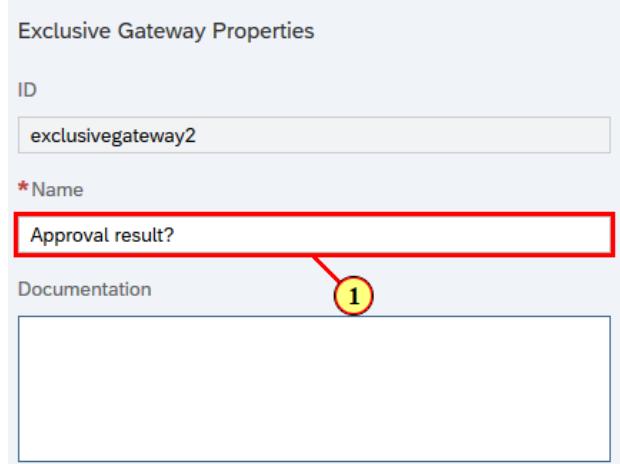
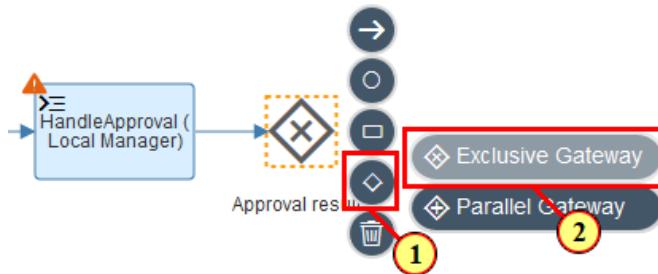
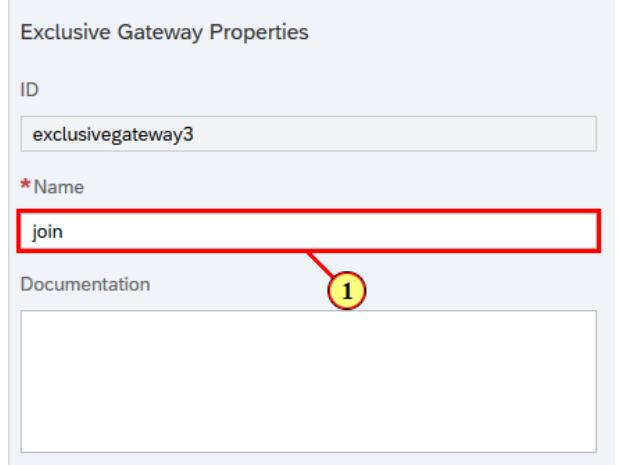
Explanation	Screenshot
<p>A new file with the name <b>CapexSampleStartPayload.json</b> is created in the project folder below <b>sample-data</b>. It is created with some default data as example.</p>	 <pre>* CAPEX_00.workflow x CapexSampleStartPayload.json x 1 &lt;! 2   "product": "Hamlet (Paperback)", 3   "inStock": true, 4   "inventory": 20000, 5   "price": 7.49, 6   "publishingDate": "1600-04-23T18:25:43.511Z", 7   "author": { "name": "William Shakespeare" }, 8   "publishers": [ "Simon &amp; Brown", "SparkNotes", "Dover Publications" ] 9 &gt;</pre>
<ol style="list-style-type: none"> <li>3. Remove the example content and Copy-Paste the context from the <b>CapexSampleStartPayload.json</b> file from the <i>Exercise1 → Helper</i> folder in GitHub.</li> <li>4. [Optional] You can choose to replace the <b>RequestId</b> of your choice.</li> <li>5. Insert your name, email address and user id into the respective fields. <b>UserId</b> is your trial user email ID.</li> <li>6. Close the file by clicking on  in the header bar. If shows a round circle instead this indicates that there are changes, which will be saved by closing.</li> </ol>	 <p>The screenshot shows the JSON file with several fields highlighted:</p> <ul style="list-style-type: none"> <li><b>RequestId</b>: A field with the value "CAPEX_REQ_00_001" highlighted with a red box and circled with a yellow circle labeled <b>1</b>.</li> <li><b>User Requester</b>: A nested object with fields "Name", "Email", and "UserId" highlighted with a red box and circled with a yellow circle labeled <b>2</b>.</li> <li><b>Close button</b>: The close button in the header bar is highlighted with a red box and circled with a yellow circle labeled <b>3</b>.</li> </ul> <pre>* CAPEX_00.workflow x CapexSampleStartPayload.json x 1 &lt;! 2   "RequestId": "CAPEX_REQ_00_001", <b>1</b> 3   "title": "Build mobile apps", 4   "Requester": { 5     "Name": "&lt;your name&gt;", 6     "Email": "&lt;your email address&gt;", 7     "UserId": "&lt;your userId&gt;", 8     "Comment": "Please approve" 9   }, 10  "Investment": { 11    "TotalCost": 10000, 12    "Type": "Software", 13    "CAPEX": 10000, 14    "ROI": 5, 15    "Country": "Germany", 16    "Currency": "EUR", 17    "Description": "Provide a fresh experience for our customers by providing new apps for our services" 18  } 19 &gt;</pre>
<ol style="list-style-type: none"> <li>7. Click the <b>Start event</b>  to select it.</li> <li>8. Click on <b>Add Task</b> .</li> <li>9. Select the <b>Script Task</b> .</li> </ol>	 <p>The screenshot shows the task palette with the following tasks:</p> <ul style="list-style-type: none"> <li><b>User Task</b> (represented by a person icon)</li> <li><b>Service Task</b> (represented by a gear icon)</li> <li><b>Script Task</b> (represented by a script icon, highlighted with a red box and circled with a yellow circle labeled <b>1</b>)</li> <li><b>Mail Task</b> (represented by an envelope icon)</li> </ul> <p>Step 1: The <b>Script Task</b> is selected.</p> <p>Step 2: The <b>User Task</b> is selected.</p> <p>Step 3: The <b>Service Task</b> is selected.</p>

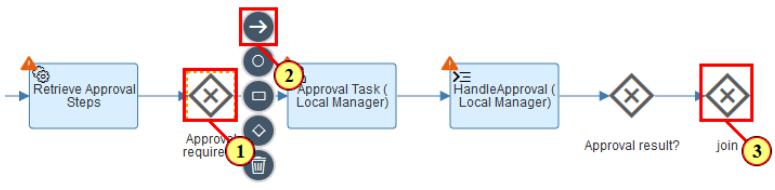
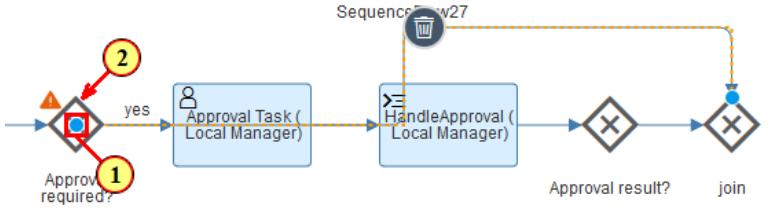
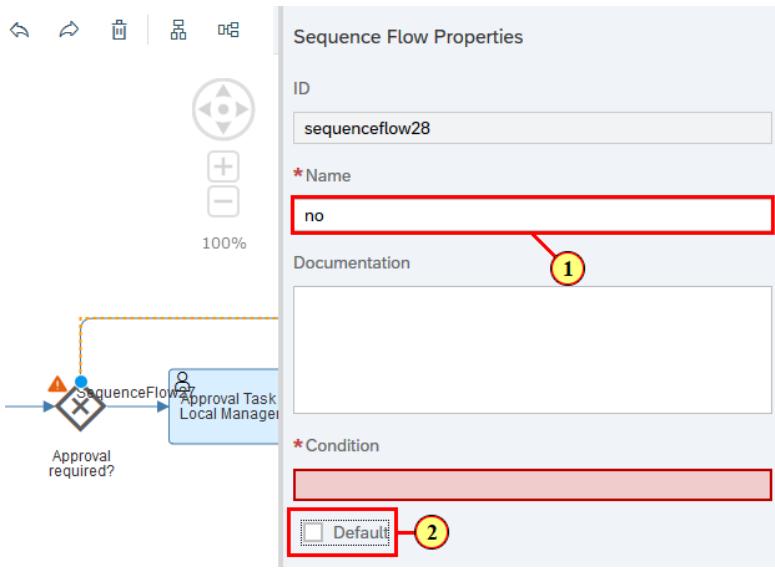
Explanation	Screenshot
<p>10. In the Script Task Properties, enter <b>Prepare Data</b> as Name.</p> <p>Note: We will configure the task later.</p>	 <p>The screenshot shows the 'Script Task Properties' dialog. The 'Name' field contains 'scripttask1'. Below it, the 'Documentation' section has a text area with a yellow circle containing the number '1' at the top right. The 'Name' field is highlighted with a red border. The 'Script File' section includes a 'Create File' button and a 'Select' button.</p>
<p>11. On the selected Script Task 'Prepare Data', click <b>Add Task</b>  in the speed buttons.</p> <p>12. Select <b>Service Task</b> .</p>	 <p>The screenshot shows a workflow diagram with a 'Capex Request' start event connected to a 'Prepare Data' script task. A context menu is open over the 'Prepare Data' task, listing options: 'User Task', 'Service Task', 'Script Task', and 'Mail Task'. The 'Service Task' option is highlighted with a red box and a yellow circle containing the number '1'. Other options are also visible.</p>

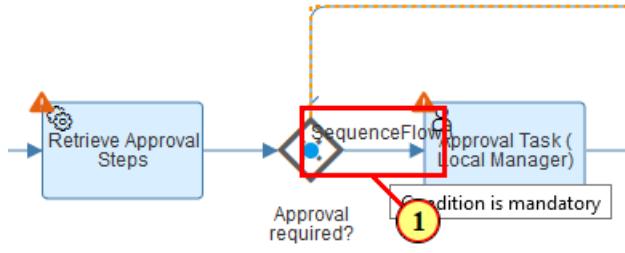
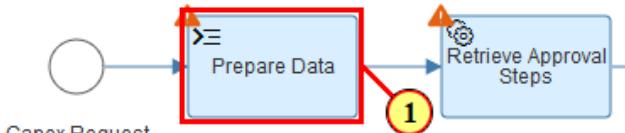
Explanation	Screenshot
<p>13. Enter <b>Retrieve Approval Steps</b> as Name.</p> <p>Note: We will configure the task later.</p>	 <p>Service Task Properties</p> <p>GENERAL DETAILS</p> <p>ID</p> <p>servicetask2</p> <p>*Name</p> <p>Retrieve Approval Steps</p> <p>Documentation</p>
<p>Note: If the workflow looks cluttered then you can use <b>Arrange Horizontally</b> on top-right corner of the editor, to realign the workflow tasks.</p>	 <p>Service Task Properties</p> <p>GENERAL DETAILS</p> <p>ID</p> <p>servicetask1</p> <p>*Name</p> <p>Retrieve Approval Steps</p> <p>Documentation</p>
<p>14. On the selected 'Retrieve Approval Steps' service task, click <b>Add Gateway</b> .</p> <p>15. Select <b>Exclusive Gateway</b> .</p>	

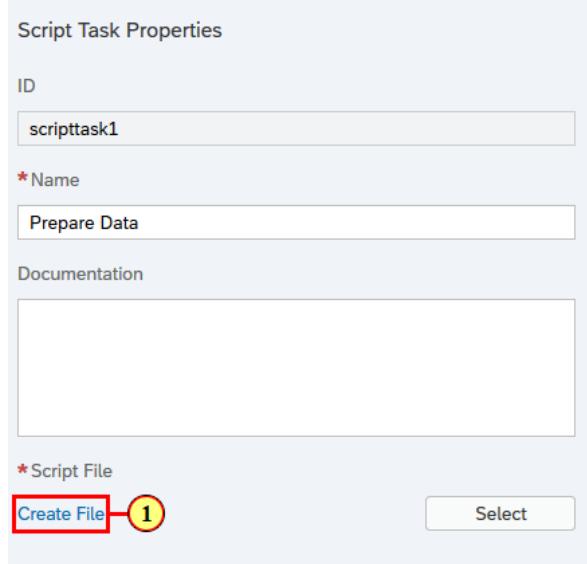
Explanation	Screenshot
<p>16. Enter <b>Approval required?</b> as gateway Name.</p> <p>Note: We will configure the gateway later, once we added more outgoing branches.</p>	
<p>17. Select <b>Add Task</b> .</p> <p>18. Select <b>User Task</b> .</p>	
<p>Note that the <b>Task ID</b> is <b>usertask1</b>. This will be needed later.</p> <p>19. Enter <b>Approval Task (Local Manager)</b> as <b>Name</b>.</p> <p>Note: IDs are automatically given to objects. If an object is deleted and re-created, it will get a new ID.</p>	

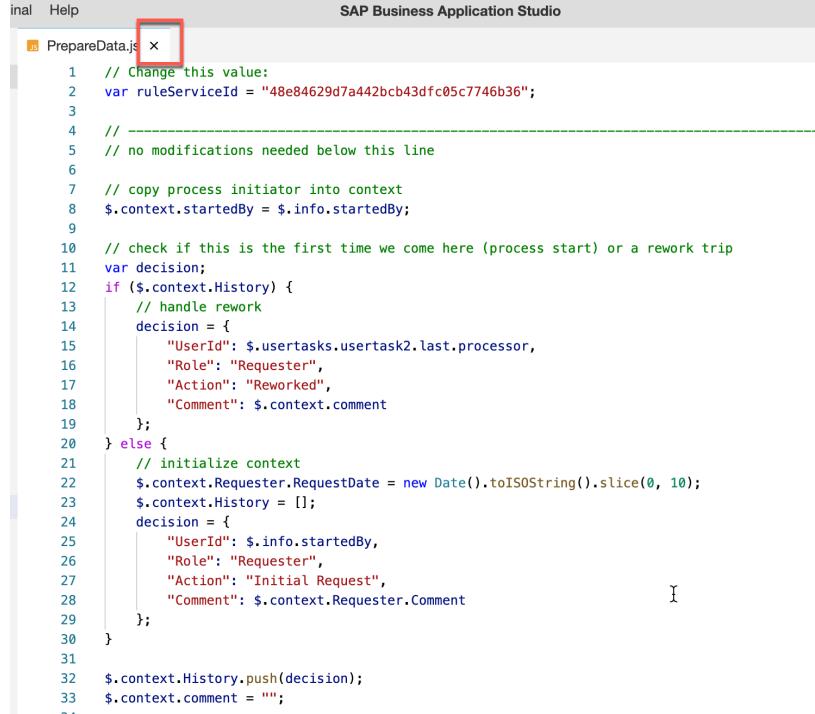
Explanation	Screenshot
<p>20. Select <b>Add Task</b> .</p> <p>21. Select <b>Script Task</b> .</p>	
<p>22. Enter <b>Handle Approval (Local Manager)</b> as <b>Name</b>.</p> <p>Note: We will configure the task later.</p>	<p>Script Task Properties</p> <p>ID: scripttask2</p> <p>* Name: HandleApproval (Local Manager)  </p> <p>Documentation:</p> <p>* Script File:  </p>
<p>23. Click <b>Add Gateway</b> .</p> <p>24. Select <b>Exclusive Gateway</b> .</p>	

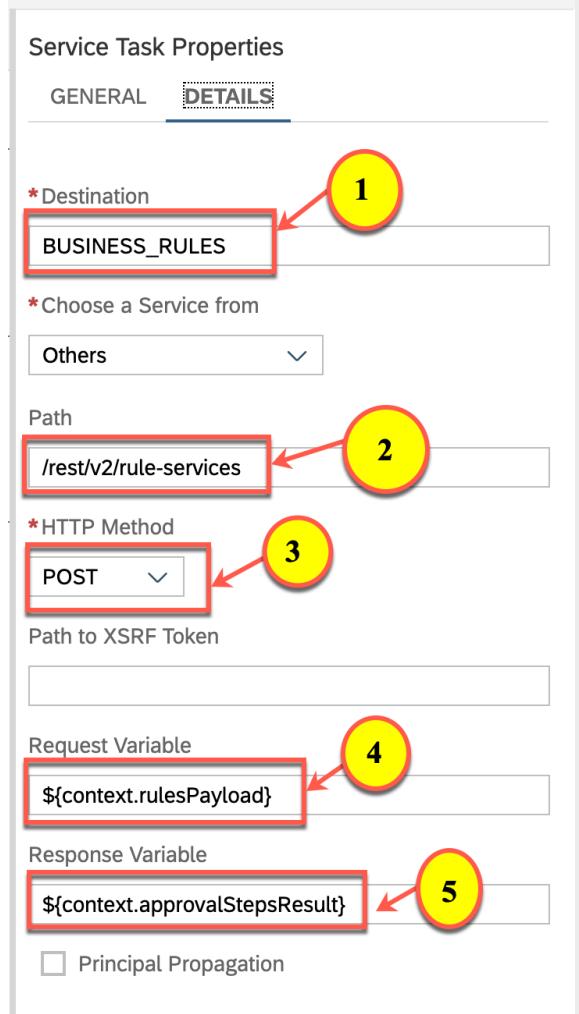
Explanation	Screenshot
25. Enter <b>Approval result?</b> as Name.	
26. Select <b>Add Gateway</b>  .	
27. Select <b>Exclusive Gateway</b>  .	

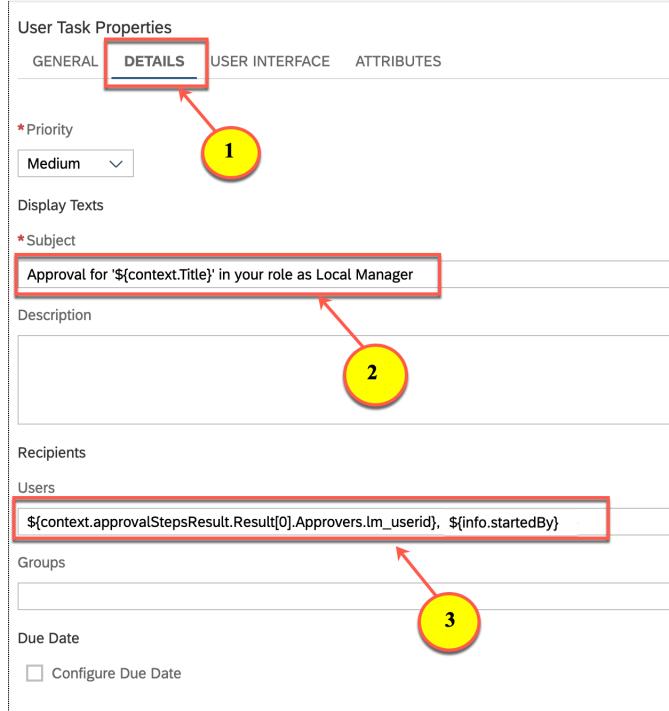
Explanation	Screenshot
<p>29. Select the <b>Exclusive Gateway</b>  with the name '<b>Approval required?</b>'</p> <p>30. Click on the <b>Sequence Flow Connector</b>  and start dragging it</p> <p>31. Drop the <b>Sequence Flow Connector</b> onto the top corner of the <b>join gateway</b>.</p> <p>Note: Sequence Flow Connectors have Connection Dots at each end. Those can be placed either in the middle or on certain positions on the edge of Flow Elements. The Layout Assistant takes this into account when drawing the connector. Additionally, the lines can be dragged with the mouse.</p>	 <p>The screenshot shows a workflow diagram with several steps: 'Retrieve Approval Steps', 'Approval Task (Local Manager)', and 'HandleApproval (Local Manager)'. An 'Exclusive Gateway' is labeled 'Approval required?'. A 'Sequence Flow Connector' is being dragged from the top corner of the 'join gateway' to the 'Approval required?' gateway. Three numbered circles (1, 2, 3) indicate specific points of interest: 1 is on the connector near the join gateway; 2 is on the connector near the exclusive gateway; and 3 is on the 'join gateway' itself.</p>
<p>32. Drag the <b>Connection Dot</b>  from the middle of the Exclusive Gateway with name <b>Approval required?</b> to its upper corner.</p> <p>33. Drop on <b>the upper corner</b> of the icon.</p>	 <p>The screenshot shows the workflow after step 32. The 'Connection Dot' has been moved to the upper corner of the 'Approval required?' gateway. A new sequence flow named 'SequenceFlow27' connects the 'yes' branch of the gateway to the 'HandleApproval' task. The 'join' gateway and its associated 'join' node are also visible.</p>
<p>Since there is now more than one Sequence Flow, each flow has a name and branch condition.</p> <p>34. Enter <b>no</b> as <b>Name</b> for the newly created (upper) Sequence Flow.</p> <p>35. Select <b>Default</b> to make this as the default branch. This will mean that if none of the condition will meet then this default route will be executed.</p>	 <p>The screenshot shows the 'Sequence Flow Properties' dialog for 'sequenceflow28'. The 'Name' field contains 'no' (circled in red). The 'Condition' field contains 'Default' (circled in red). The 'Default' checkbox is checked (circled in yellow).</p>

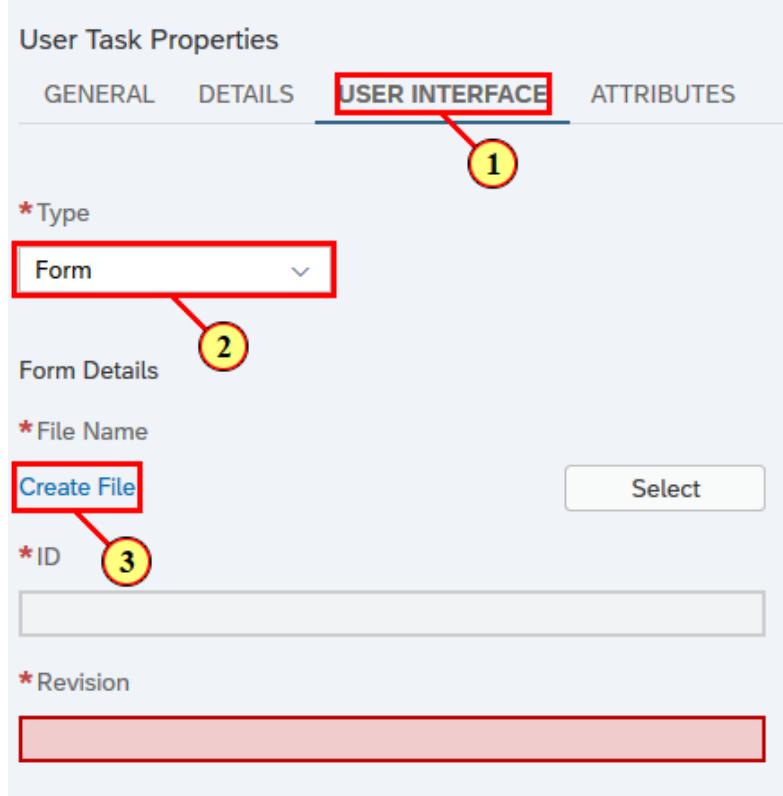
Explanation	Screenshot
<p>36. Select the <b>Sequence Flow</b> between the gateway 'Approval required?' and the user task 'Approval Task (Local Manager)'</p>	
<p>37. Enter <b>yes</b> as <b>Name</b> of the sequence flow.</p> <p>38. Enter  <code>#{context.approvalStepsResult.Result[0].Approvers.lm_required}</code> as <b>Condition</b>.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in the condition. Sometimes during copy, spaces are added automatically. In this case, they need to be removed.</p> <p>Note: This expression uses data in the workflow context to evaluate whether this branch should be taken. This data was retrieved in previous step by a service call to Business Rules.</p>	<p>Sequence Flow Properties</p> <p>ID: sequenceflow6</p> <p>*Name: yes (highlighted with a red box)</p> <p>Documentation: (highlighted with a yellow circle 1)</p> <p>*Condition: text.approvalStepsResult.Result[0].Approvers.lm_required (highlighted with a red box)</p> <p>[Default] (highlighted with a yellow circle 2)</p>
<p>39. Select the script task <b>Prepare Data</b>.</p>	

Explanation	Screenshot
<p>40. In the properties, click <b>Create File</b>.</p>	 <p>The screenshot shows the 'Script Task Properties' dialog. It includes fields for ID (scripttask1), Name (Prepare Data), Documentation, and a 'Script File' section. A red box highlights the 'Create File' button, which is circled in yellow.</p>
<p>41. Enter <b>PrepareData</b> as <b>Name</b>. 42. Click on <b>Create</b>.</p>	 <p>The screenshot shows the 'Create New File' dialog. It has a 'Name' field containing 'PrepareData' and a 'Create' button. Both the 'Name' field and the 'Create' button are highlighted with red boxes and circled in yellow.</p>

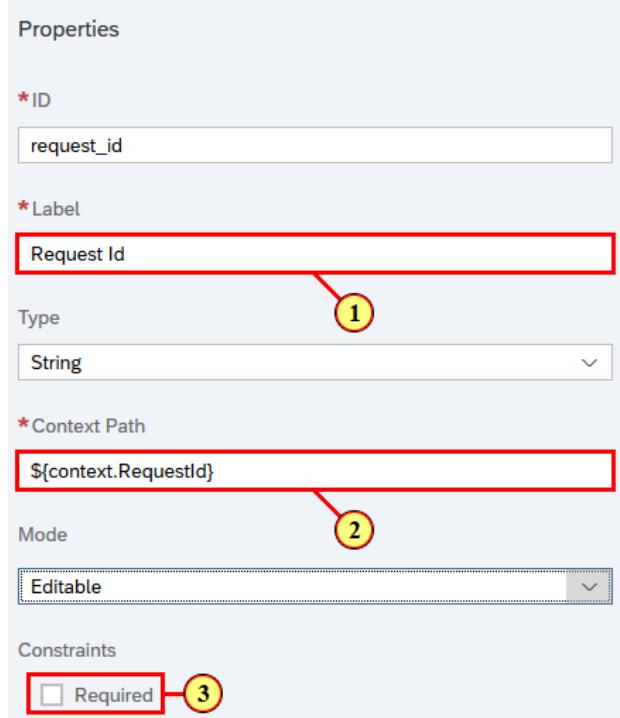
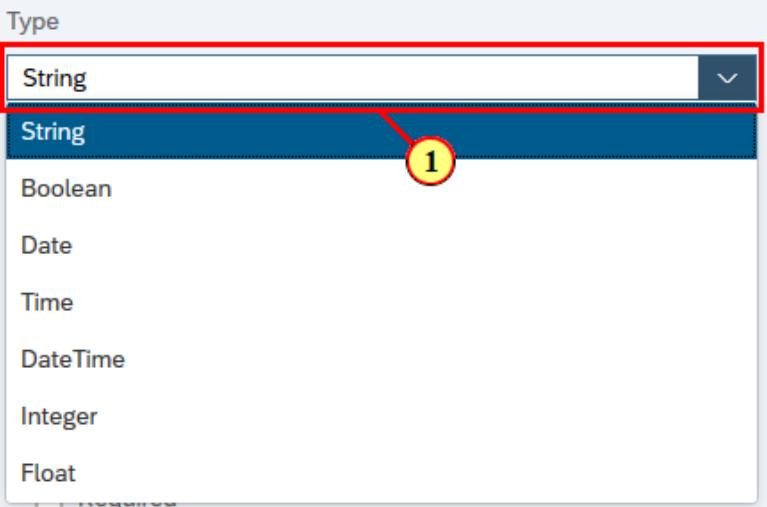
Explanation	Screenshot
<p>43. Delete all the default content and replace it with the <b>PrepareData.js</b> text from <i>Exercise 1 → Helper</i> folder in GitHub.</p> <p>Note: This workflow calls a pre-build rule which you don't need to change now. In a business rules exercise, a custom rule will be created and then you can replace it with your rule service id.</p> <p>44. Close the file by clicking on .</p>	 <pre data-bbox="755 397 1455 1079"> 1 // Change this value: 2 var ruleServiceId = "48e84629d7a442bcba3dfc05c7746b36"; 3 4 // ----- 5 // no modifications needed below this line 6 7 // copy process initiator into context 8 \$.context.startedBy = \$.info.startedBy; 9 10 // check if this is the first time we come here (process start) or a rework trip 11 var decision; 12 if (\$.context.History) { 13     // handle rework 14     decision = { 15         "UserId": \$.usertasks.usertask2.last.processor, 16         "Role": "Requester", 17         "Action": "Reworked", 18         "Comment": \$.context.comment 19     }; 20 } else { 21     // initialize context 22     \$.context.Requester.RequestDate = new Date().toISOString().slice(0, 10); 23     \$.context.History = []; 24     decision = { 25         "UserId": \$.info.startedBy, 26         "Role": "Requester", 27         "Action": "Initial Request", 28         "Comment": \$.context.Requester.Comment 29     }; 30 } 31 32 \$.context.History.push(decision); 33 \$.context.comment = ""; </pre>
<p>45. Select the Service Task <b>Retrieve Approval Steps</b>.</p>	 <pre data-bbox="747 1110 1390 1262"> graph LR     Start(( )) --&gt; &gt;=  Task1[Prepare Data]     Task1 --&gt; Task2[Retrieve Approval Steps]     Task2 --&gt; &gt;=  End(( ))     </pre>

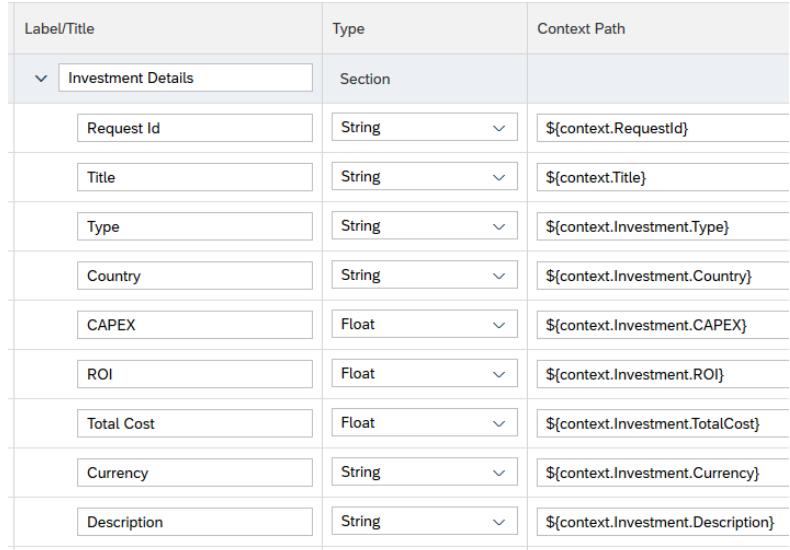
Explanation	Screenshot
<p>46. Select the <b>DETAILS</b> tab.</p> <p>47. Enter <b>BUSINESS_RULES</b> as <b>Destination</b>.</p> <p>48. Enter <b>/rest/v2/rule-services</b> as <b>Path</b>.</p> <p>49. Enter HTTP Method as <b>POST</b>.</p> <p>50. Enter <code>#{context.rulesPayload}</code> as <b>Request Variable</b>.</p> <p>51. Enter <code>#{context.approvalStepsResult}</code> as <b>Response Variable</b>.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in the <b>path, request and response variable</b> text. Sometimes during copy and paste, the spaces are added automatically. In this case, they need to be removed.</p>	 <p>Service Task Properties</p> <p>GENERAL DETAILS</p> <p>*Destination BUSINESS_RULES</p> <p>*Choose a Service from Others</p> <p>Path /rest/v2/rule-services</p> <p>*HTTP Method POST</p> <p>Path to XSRF Token</p> <p>Request Variable \${context.rulesPayload}</p> <p>Response Variable \${context.approvalStepsResult}</p> <p><input type="checkbox"/> Principal Propagation</p>
<p>52. Select the User Task <b>Approval Task (Local Manager)</b>.</p>	 <pre> graph LR     A{Approval required?} -- yes --&gt; B[Approval Task (Local Manager)]     B --&gt; C[HandleApproval (Local Manager)]     </pre>

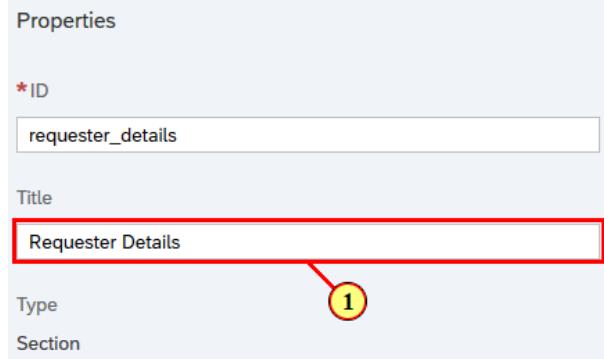
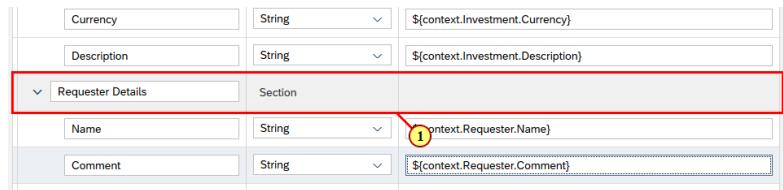
Explanation	Screenshot
<p>53. Select the <b>DETAILS</b> tab.</p> <p>54. Enter <b>Approval for</b> '<code>#{context.Title}</code>' in your role as Local Manager as <b>Subject</b>.</p> <p>55. Enter  <code>#{context.approvalStepsResult.Result[0].Approvers.lm_userid}, \${info.startedBy}</code> as <b>Recipients - Users</b>.  It means that this user task will be sent to the users retrieved from Business Rules and the workflow initiator.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces <b>Users</b> text. Sometimes during copy and paste, the spaces are added automatically. In this case, they need to be removed.</p>	 <p>User Task Properties</p> <p>GENERAL DETAILS USER INTERFACE ATTRIBUTES</p> <p>*Priority Medium</p> <p>Display Texts</p> <p>*Subject Approval for '\${context.Title}' in your role as Local Manager</p> <p>Description</p> <p>Recipients</p> <p>Users \${context.approvalStepsResult.Result[0].Approvers.lm_userid}, \${info.startedBy}</p> <p>Groups</p> <p>Due Date <input type="checkbox"/> Configure Due Date</p>

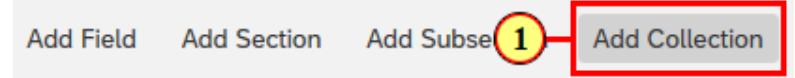
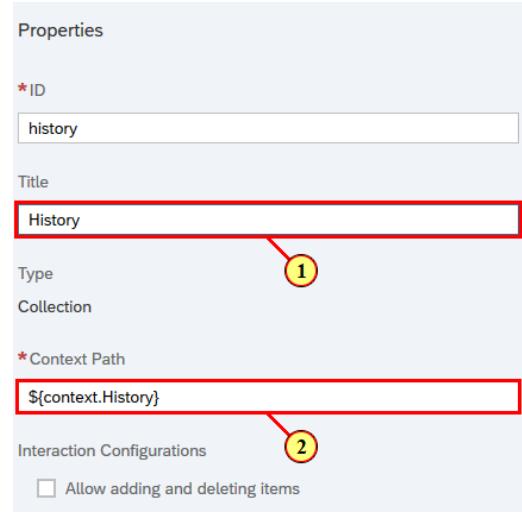
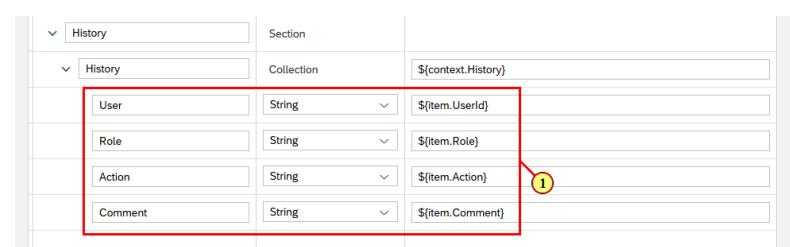
Explanation	Screenshot
<p>56. Select the <b>USER INTERFACE</b> tab.</p> <p>57. Select <b>Form</b> in the dropdown menu for <b>Type</b>.</p> <p>58. Click <b>Create File</b>.</p> <p>Note: Forms are a quick way to create simple task UIs.</p>	 <p>The screenshot shows the 'User Task Properties' dialog with the 'USER INTERFACE' tab selected (indicated by a red box and yellow circle 1). The 'Type' dropdown is set to 'Form' (indicated by a red box and yellow circle 2). The 'Create File' button is highlighted with a red box and yellow circle 3.</p>
<p>59. In the popup, enter <b>CapexApproval_00</b> as <b>Name</b>.</p> <p>60. Press <b>TAB</b> to accept the generated Id.</p> <p>61. Enter <b>1</b> as <b>Revision</b>.</p> <p>62. Click <b>Create</b>.</p>	 <p>The screenshot shows the 'New Form' dialog. The 'Name' field contains 'CapexApproval_00' (indicated by a red box and yellow circle 1). The 'ID' field contains 'capexapproval_00' (indicated by a red box and yellow circle 2). The 'Revision' field contains '1' (indicated by a red box and yellow circle 3). The 'Type' field is set to 'Task Form'. The 'Create' button is highlighted with a red box and yellow circle 4.</p>

Explanation	Screenshot
<p>Note: With Workflow Forms, building Fiori form UIs becomes as easy as building workflows. The Forms Editor enables a declarative low-code approach to rapidly develop simple form UIs. Due to its full integration into Workflow a form can directly work on the values of the workflow context.</p>	<p>The screenshot shows the SAP Cloud Platform Workflow Management Forms Editor interface. A task form titled 'CapexApproval_00' is displayed. The top bar shows the form's ID, revision, and type. Below the title, there are tabs for 'FIELDS' and 'DECISIONS'. A toolbar below the tabs includes buttons for 'Add Field', 'Add Section', 'Add Subsection', 'Add Collection', and other common operations like Cut, Copy, Paste, and Delete. A message at the bottom says 'To start, add an element.' On the right side, a vertical panel labeled 'Properties' shows the mode as 'Editable'. The main workspace is currently empty.</p>
<p>63. Click <b>Add Section</b>.</p>	<p>The screenshot shows the same task form 'CapexApproval_00'. The 'Add Section' button in the toolbar is highlighted with a red box and a yellow circle containing the number 1. The rest of the interface remains the same.</p>
<p>64. Enter <b>Investment Details</b> as Title.</p> <p>Note: The ID field is automatically filled out based on the given title.</p>	<p>The screenshot shows the 'Properties' panel for the section. It includes fields for 'ID' (set to 'investment_details'), 'Title' (set to 'Investment Details'), 'Type', and 'Section'. The 'Title' field is highlighted with a red box and a yellow circle containing the number 1.</p>
<p>65. Select <b>Add Field</b>.</p>	<p>The screenshot shows the toolbar again, with the 'Add Field' button highlighted with a red box and a yellow circle containing the number 1. The other buttons in the toolbar are visible but not highlighted.</p>

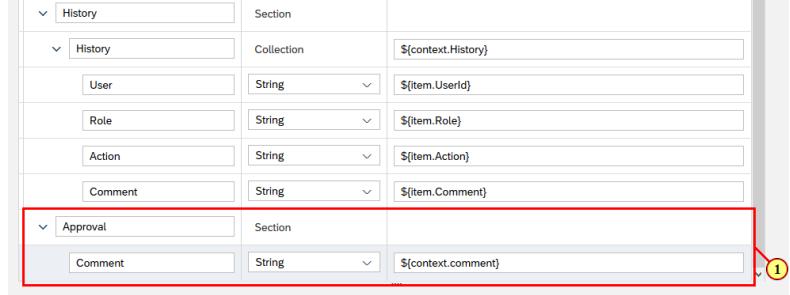
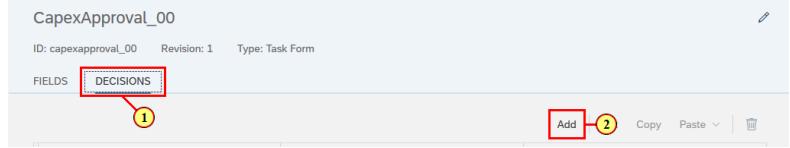
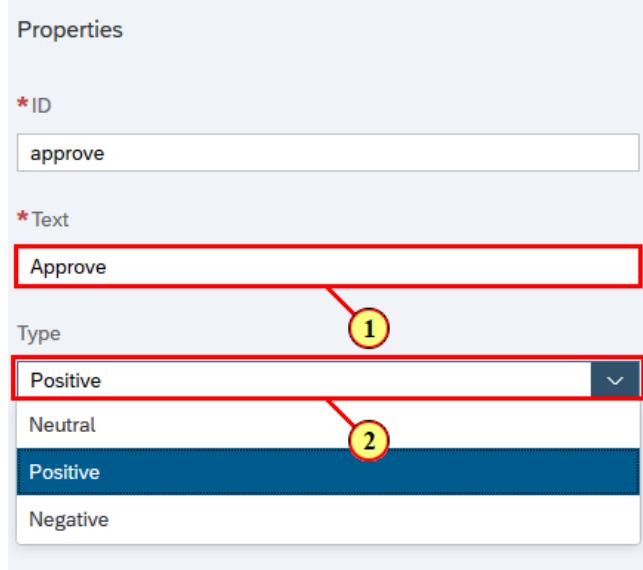
Explanation	Screenshot
<p>66. Enter <b>Request Id</b> as <b>Label</b>. The ID is then automatically derived from this.</p> <p>67. Enter <code> \${context.RequestId}</code> as <b>Context Path</b>.</p> <p>68. An input field can also be marked as <b>Required</b>. Since the <b>Request Id</b> is already given and should not be editable, leave it unchecked.</p>	 <p>Properties</p> <p>* ID request_id</p> <p>* Label Request Id</p> <p>Type String</p> <p>* Context Path \${context.RequestId}</p> <p>Mode Editable</p> <p>Constraints <input type="checkbox"/> Required</p>
<p>69. Clicking on the <b>Type</b> field opens a dropdown list. Select the type as <b>String</b>.</p>	 <p>Type</p> <p>String</p> <p>String</p> <p>Boolean</p> <p>Date</p> <p>Time</p> <p>DateTime</p> <p>Integer</p> <p>Float</p>
<p>70. Clicking on the <b>Mode</b> field opens a dropdown list. Select <b>Display-Only</b>.</p>	 <p>Mode</p> <p>Display-Only</p> <p>Editable</p> <p>Display-Only</p>

Explanation	Screenshot																																	
<p>71. In the same way, add the following fields with given Label, Type and Context Path:</p> <ul style="list-style-type: none"> <li>○ Set the property of all of the fields as <b>Editable</b> and <b>Required</b>.</li> </ul> <p><b>Label: Title</b>  <b>Type:</b> String  <b>Context Path:</b> \${context.Title}</p> <p><b>Label: Type</b>  <b>Type:</b> String  <b>Context Path:</b>  \${context.Investment.Type}</p> <p><b>Label: Country</b>  <b>Type:</b> String  <b>Context Path:</b>  \${context.Investment.Country}</p> <p><b>Label: CAPEX</b>  <b>Type:</b> Float  <b>Context Path:</b>  \${context.Investment.CAPEX}</p> <p><b>Label: ROI</b>  <b>Type:</b> Float  <b>Context Path:</b>  \${context.Investment.ROI}</p> <p><b>Label: Total Cost</b>  <b>Type:</b> Float  <b>Context Path:</b>  \${context.Investment.TotalCost}</p> <p><b>Label: Currency</b>  <b>Type:</b> String  <b>Context path:</b>  \${context.Investment.Currency}</p> <p><b>Label: Description</b>  <b>Type:</b> String  <b>Context Path:</b>  \${context.Investment.Description}</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in <b>ContextPath</b> text. Sometimes during copy and paste, the spaces are added automatically. In that case the field will show red box. Remove the spaces.</p>	 <table border="1" data-bbox="747 846 1537 1396"> <thead> <tr> <th data-bbox="755 857 1057 889">Label/Title</th> <th data-bbox="1057 857 1171 889">Type</th> <th data-bbox="1171 857 1537 889">Context Path</th> </tr> </thead> <tbody> <tr> <td data-bbox="755 899 1057 931" style="background-color: #e0e0e0;">▼ Investment Details</td><td data-bbox="1057 899 1171 931" style="background-color: #e0e0e0;">Section</td><td data-bbox="1171 899 1537 931"></td></tr> <tr> <td data-bbox="755 941 1057 973">Request Id</td><td data-bbox="1057 941 1171 973">String</td><td data-bbox="1171 941 1537 973">\${context.RequestId}</td></tr> <tr> <td data-bbox="755 984 1057 1015">Title</td><td data-bbox="1057 984 1171 1015">String</td><td data-bbox="1171 984 1537 1015">\${context.Title}</td></tr> <tr> <td data-bbox="755 1026 1057 1058">Type</td><td data-bbox="1057 1026 1171 1058">String</td><td data-bbox="1171 1026 1537 1058">\${context.Investment.Type}</td></tr> <tr> <td data-bbox="755 1068 1057 1100">Country</td><td data-bbox="1057 1068 1171 1100">String</td><td data-bbox="1171 1068 1537 1100">\${context.Investment.Country}</td></tr> <tr> <td data-bbox="755 1110 1057 1142">CAPEX</td><td data-bbox="1057 1110 1171 1142">Float</td><td data-bbox="1171 1110 1537 1142">\${context.Investment.CAPEX}</td></tr> <tr> <td data-bbox="755 1153 1057 1184">ROI</td><td data-bbox="1057 1153 1171 1184">Float</td><td data-bbox="1171 1153 1537 1184">\${context.Investment.ROI}</td></tr> <tr> <td data-bbox="755 1195 1057 1227">Total Cost</td><td data-bbox="1057 1195 1171 1227">Float</td><td data-bbox="1171 1195 1537 1227">\${context.Investment.TotalCost}</td></tr> <tr> <td data-bbox="755 1237 1057 1269">Currency</td><td data-bbox="1057 1237 1171 1269">String</td><td data-bbox="1171 1237 1537 1269">\${context.Investment.Currency}</td></tr> <tr> <td data-bbox="755 1279 1057 1311">Description</td><td data-bbox="1057 1279 1171 1311">String</td><td data-bbox="1171 1279 1537 1311">\${context.Investment.Description}</td></tr> </tbody> </table>	Label/Title	Type	Context Path	▼ Investment Details	Section		Request Id	String	\${context.RequestId}	Title	String	\${context.Title}	Type	String	\${context.Investment.Type}	Country	String	\${context.Investment.Country}	CAPEX	Float	\${context.Investment.CAPEX}	ROI	Float	\${context.Investment.ROI}	Total Cost	Float	\${context.Investment.TotalCost}	Currency	String	\${context.Investment.Currency}	Description	String	\${context.Investment.Description}
Label/Title	Type	Context Path																																
▼ Investment Details	Section																																	
Request Id	String	\${context.RequestId}																																
Title	String	\${context.Title}																																
Type	String	\${context.Investment.Type}																																
Country	String	\${context.Investment.Country}																																
CAPEX	Float	\${context.Investment.CAPEX}																																
ROI	Float	\${context.Investment.ROI}																																
Total Cost	Float	\${context.Investment.TotalCost}																																
Currency	String	\${context.Investment.Currency}																																
Description	String	\${context.Investment.Description}																																

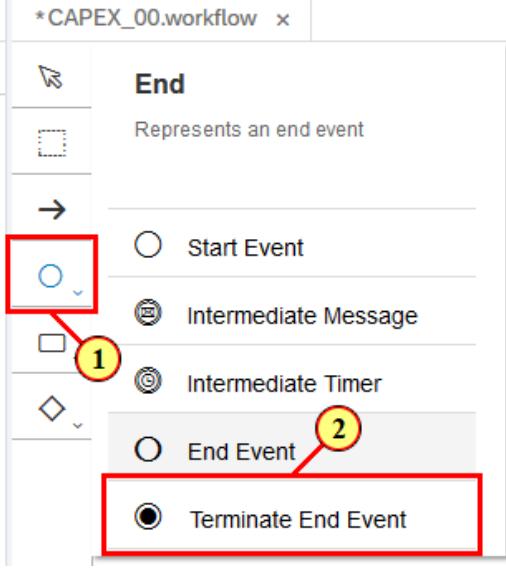
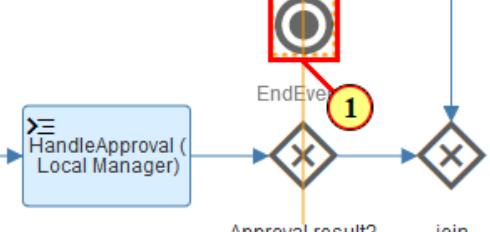
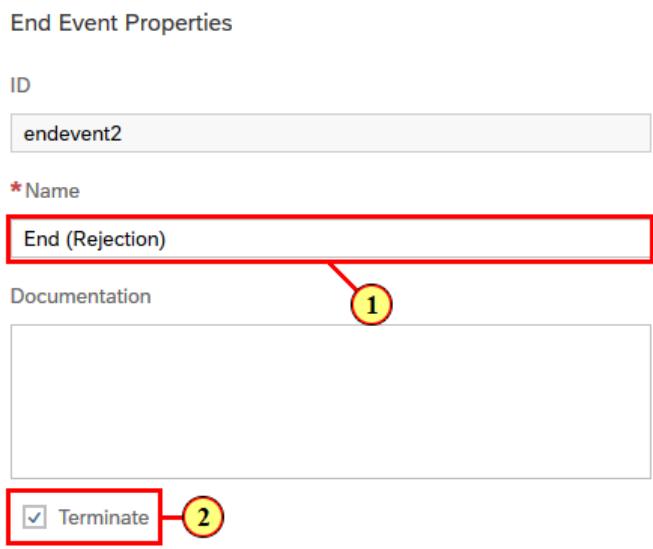
Explanation	Screenshot
<p>72. Select <b>Investment Details</b> section.</p> <p>73. Select <b>Add Section</b>.</p>	
<p>74. Enter <b>Requester Details</b> as Title.</p>	
<p>75. Add the following fields.</p> <p><b>Label:</b> Name  <b>Type:</b> String  <b>Mode:</b> Display-Only  <b>Control:</b> Input  <b>Context Path:</b> \${context.Requester.Name}</p> <p><b>Label:</b> Comment  <b>Type:</b> String  <b>Mode:</b> Editable  <b>Control:</b> Text Area  <b>Required:</b> no  <b>Context Path:</b> \${context.Requester.Comment}</p>	
<p>76. Select the <b>Requester Details</b> section.</p>	
<p>77. Select <b>Add Section</b>. Name the new Section <b>History</b>.</p>	

Explanation	Screenshot
<p>78. Select <b>Add Collection</b>.</p> <p>Note: Collections allow for an array of items to be displayed as table.</p>	
<p>79. Enter <b>History</b> as <b>Title</b>.</p> <p>80. Enter <code>#{context.History}</code> as <b>Context Path</b>.</p>	
<p>81. Add the following values for <b>Label</b> and <b>Context Path</b> and select Type <b>String</b> and mark them as <b>Display-Only</b>.</p> <p><b>Label: User</b> Context Path: <code>#{item.UserId}</code></p> <p><b>Label: Role</b> Context Path: <code>#{item.Role}</code></p> <p><b>Label: Action</b> Context Path: <code>#{item.Action}</code></p> <p><b>Label: Comment</b> Context Path: <code>#{item.Comment}</code></p> <p>Note that inside a collection, the variable <b>item</b> refers to a single row.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in <b>ContextPath</b> text. Sometimes during copy and paste, the spaces are added automatically (esp. the leading and trailing spaces). In that case the field will show red box. Remove the spaces.</p>	

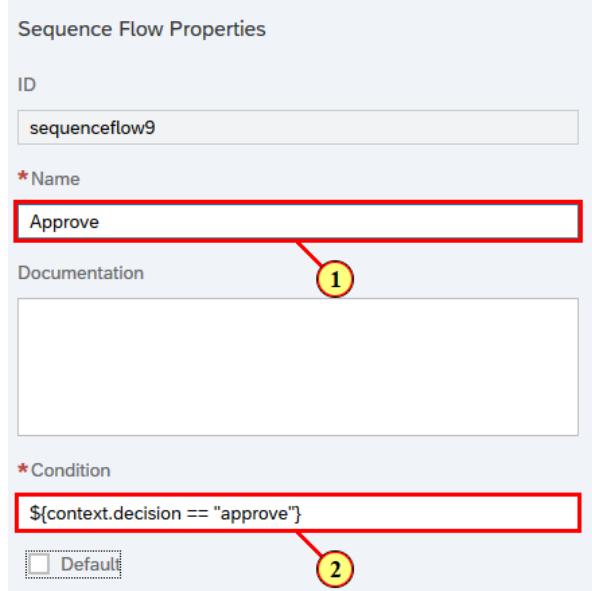
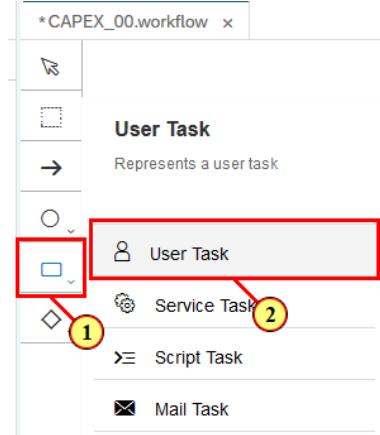
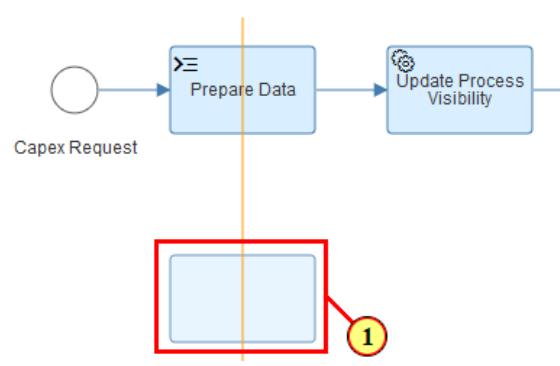
Explanation	Screenshot
<p>82. Add a new Section with title <b>Approval</b>.</p> <p>83. Add a new field and name it <b>Comment</b>.</p> <p>84. Enter <code>#{context.comment}</code> as Context Path</p> <p>85. Change the UI Control type from <b>Input field</b> to <b>Text Area</b>.</p>	<p>The screenshot shows the 'Properties' dialog for a field named 'comment'. The field has the following properties:</p> <ul style="list-style-type: none"><li><b>ID:</b> comment</li><li><b>Label:</b> Comment (highlighted with a red box and circled with a yellow number 2)</li><li><b>Type:</b> String (highlighted with a red box and circled with a yellow number 2)</li><li><b>Context Path:</b> \${context.comment} (highlighted with a red box and circled with a yellow number 3)</li><li><b>Mode:</b> Editable (highlighted with a red box and circled with a yellow number 3)</li><li><b>Constraints:</b> Required (unchecked)</li><li><b>UI Configuration:</b> Control<ul style="list-style-type: none"><li>Text Area (highlighted with a red box and circled with a yellow number 4)</li><li>Input</li><li>Dropdown</li><li>Radio Buttons</li></ul></li></ul>

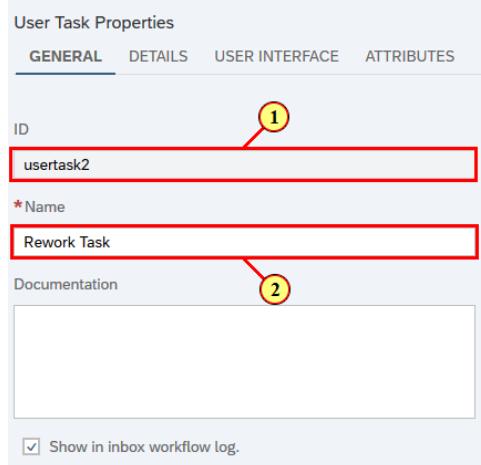
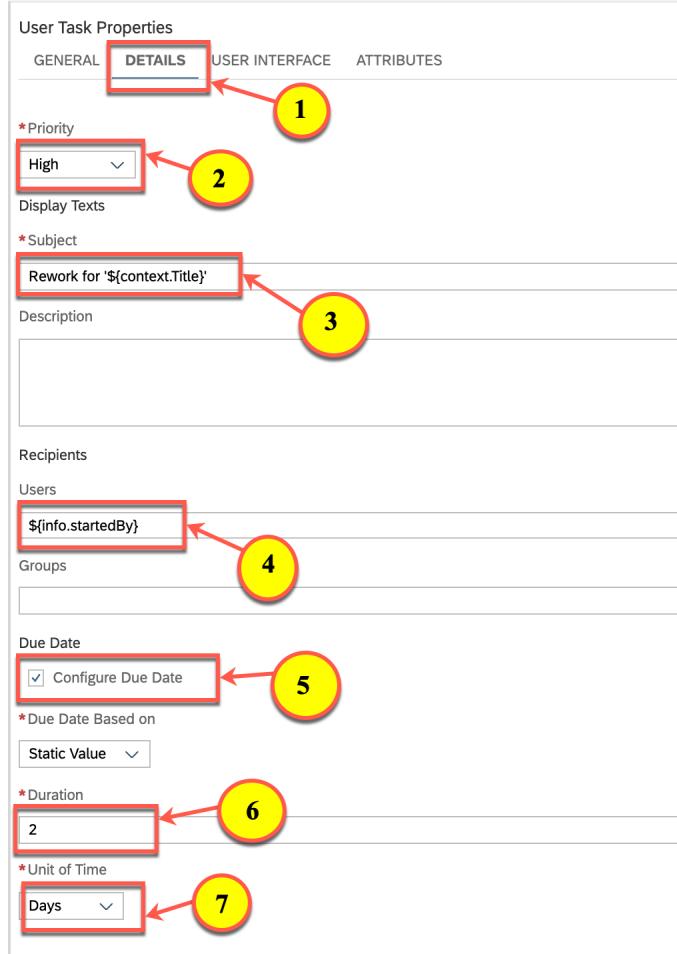
Explanation	Screenshot
86. Ensure your Form looks like this.	
87. Select the <b>Decisions</b> tab. 88. Select <b>Add</b> to add a new decision option.	
89. Enter <b>Approve</b> as <b>Text</b> of the decision button. 90. Select <b>Positive</b> as <b>Type</b> from the dropdown-list.	
91. In the same way, add a decision option <b>Reject</b> with type <b>Negative</b> and one with text <b>Rework</b> of <b>Neutral</b> type.	
92. <b>Close</b> the file by clicking <b>x</b> .	

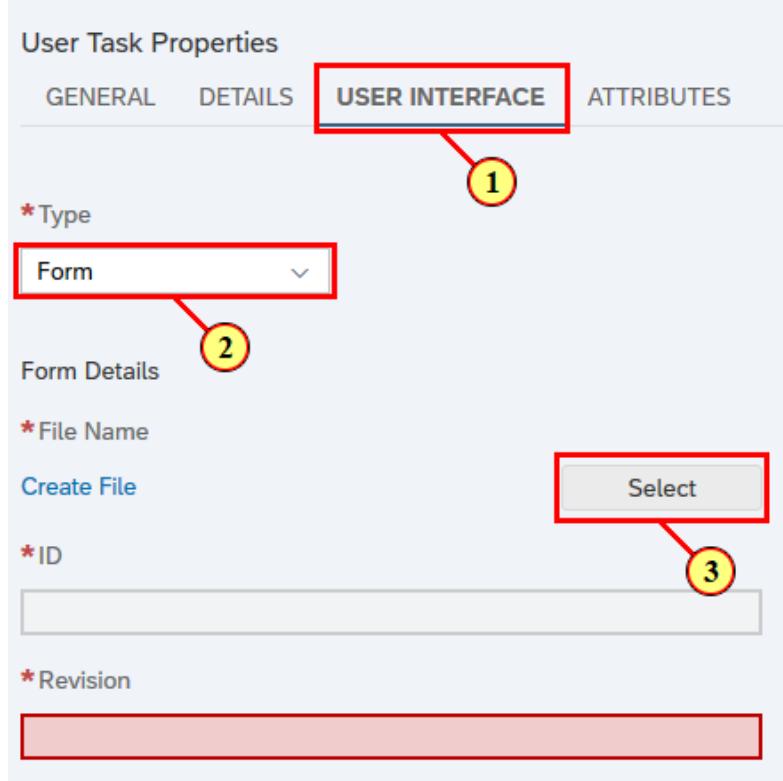
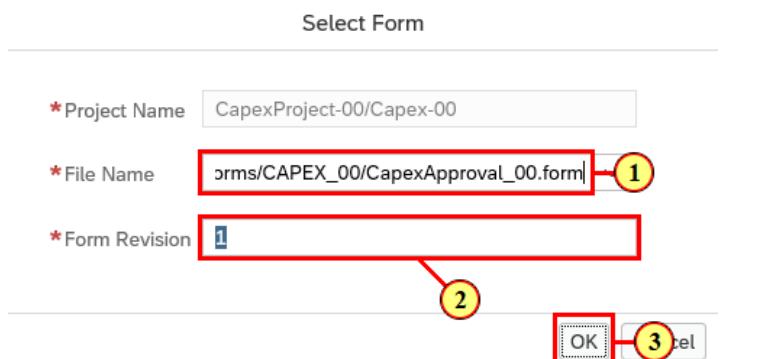
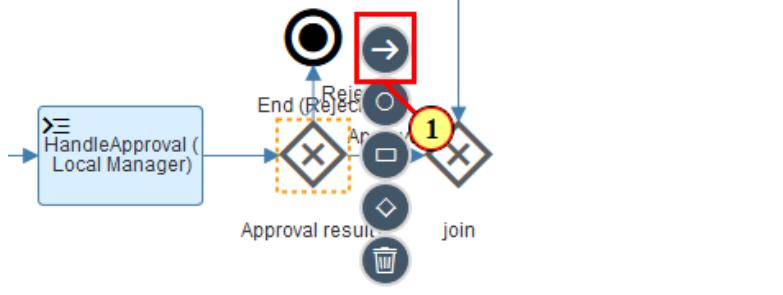
Explanation	Screenshot
<p>93. Select the script task <b>Handle Approval (Local Manager)</b>.</p>	<pre> graph LR     A[Approval Task (Local Manager)] --&gt; B[HandleApproval (Local Manager)]     B --&gt; C{X}     C --&gt; D((Approval result?))   </pre>
<p>94. In the <b>Script Task Properties</b>, select <b>Create File</b>.</p>	<p>* Script File Create File <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">1</span></p>
<p>95. Enter <b>HandleApprovalLocalManager</b> as <b>Name</b> of the new file.</p> <p>96. Select <b>Create</b>.</p>	<p>Create New File</p> <p>* Name <span style="border: 1px solid red; border-radius: 5px; padding: 2px;">HandleApprovalLocalManager</span> <span style="border: 1px solid red; border-radius: 5px; padding: 2px;">1</span></p> <p><span style="border: 1px solid red; border-radius: 5px; padding: 2px;">Create</span> <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">2</span> Cancel</p>
<p>97. Replace the default content with the <b>HandleApprovalLocalManager.js</b> file in the <i>Exercise 1 → Helper</i> folder in GitHub.</p> <p>98. Close the file by clicking .</p> <p>Note: This script refers to the Task ID (usertask1) multiple times. This is the id of the User Task "Approval Task (Local Manager)" which was mentioned earlier. It is automatically assigned to each task. If your ID is different, it needs to be replaced here. Please note only replace if your user task has a different id.</p>	<pre> * CAPEX_00.workflow x * HandleApprovalLocalManager.js x 1 var decision = { 2   "UserId": \$.usertasks.usertask1.last.processor, 3   "Role": "Local Manager", 4   "Action": \$.usertasks.usertask1.last.decision, 5   "Comment": \$.context.comment 6 }; 7 8 \$.context.History.push(decision); 9 \$.context.comment = ""; 10 11 \$.context.decision = \$.usertasks.usertask1.last.decision;   </pre>

Explanation	Screenshot
<p>99. From the left sidebar, select <b>Add Event</b>.</p> <p>100. Select the <b>Terminate End Event</b> .</p>	
<p>101. Place the <b>End Event</b> above the exclusive gateway named <b>Approval result</b>?</p> <p>Note: The yellow guiding lines help you position it nicely.</p>	
<p>102. Enter <b>End (Rejection)</b> as <b>Name</b>.</p> <p>103. Ensure that <b>Terminate</b> is checked.</p>	

Explanation	Screenshot
<p>104. Select the exclusive gateway with name <b>Approval result?</b></p> <p>105. Drag the sequence flow connector onto the end event with name <b>End (Rejection)</b></p>	
<p>106. Enter <b>Reject</b> as Name.</p> <p>107. Select the <b>checkbox for Default</b> condition.</p>	<p>Sequence Flow Properties</p> <p>ID: sequenceflow11</p> <p>* Name: <b>Reject</b></p> <p>Documentation: <b>1</b></p> <p><input checked="" type="checkbox"/> Default <b>2</b></p>
<p>108. Select the sequence flow connector between the gateways named <b>Approval result?</b> and <b>join</b>.</p> <p>Note: Once there is more than one outgoing sequence flow from an Exclusive Gateway, all of them can be named. Exactly one must have the 'default', and all others customized trigger conditions based on the workflow context.</p>	

Explanation	Screenshot
<p>109. Enter <b>Approve</b> as <b>Name</b>.</p> <p>110. Enter <code>#{context.decision == "approve"}</code> as <b>Condition</b>.</p>	 <p>The screenshot shows the 'Sequence Flow Properties' dialog. The 'Name' field contains 'Approve' (highlighted with a red box and circled with yellow number 1). The 'Condition' field contains the expression <code>#{context.decision == "approve"}</code> (highlighted with a red box and circled with yellow number 2).</p>
<p>111. From the left sidebar, select <b>Add Task</b> .</p> <p>112. Select <b>User Task</b> .</p>	 <p>The screenshot shows the task palette with 'User Task' selected (highlighted with a red box and circled with yellow number 1). Other options like 'Service Task' and 'Script Task' are also shown.</p>
<p>113. Place the <b>User Task</b> below the script task <b>Prepare Data</b>.</p> <p>Note: The yellow guiding lines help you position it nicely.</p>	 <p>The screenshot shows a workflow diagram. A blue rounded rectangle labeled 'Capex Request' is connected to a light blue rounded rectangle labeled 'Prepare Data'. This is followed by another light blue rounded rectangle labeled 'Update Process Visibility'. A yellow vertical guiding line is positioned to the right of the 'Prepare Data' task. A yellow rectangular box highlights the area where the 'User Task' would be placed between the two script tasks. A yellow circle with number 1 points to the bottom edge of the 'Prepare Data' task.</p>

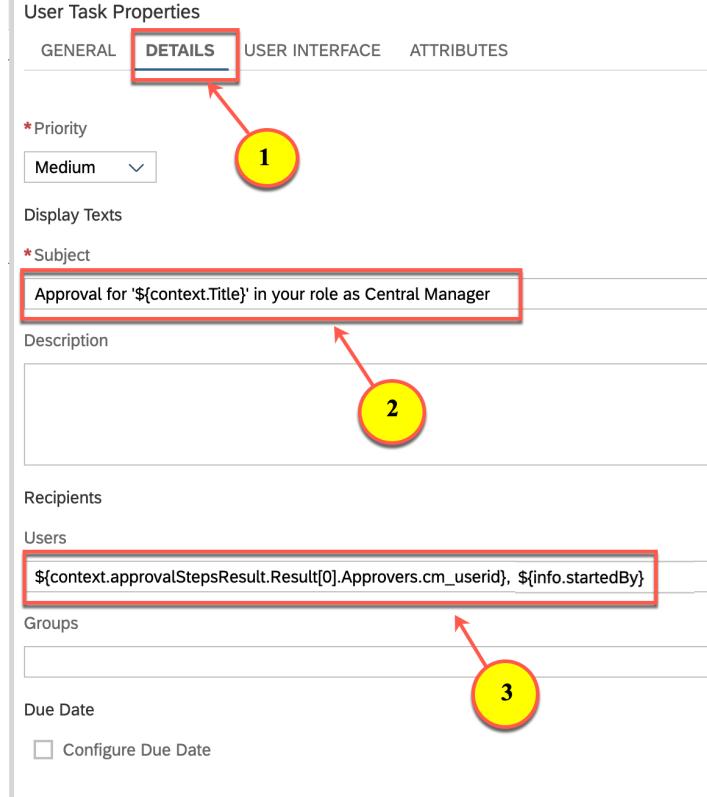
Explanation	Screenshot
<p>114. Verify that the <b>ID</b> is <code>usertask2</code>. If it differs, the references in the script file <b>Prepare Data.js</b> needs to be adapted accordingly.</p> <p>115. Enter <b>Rework Task</b> as <b>Name</b>.</p>	 <p>User Task Properties GENERAL DETAILS USER INTERFACE ATTRIBUTES</p> <p>ID usertask2</p> <p>*Name Rework Task</p> <p>Documentation Documentation area (empty)</p> <p><input checked="" type="checkbox"/> Show in inbox workflow log.</p>
<p>116. Select the <b>DETAILS</b> tab.</p> <p>117. Select the <b>Priority</b> as <b>High</b>.</p> <p>118. Enter <b>Rework for '\${context.Title}'</b> as <b>Subject</b>.</p> <p>119. Enter <b> \${info.startedBy}</b> as Recipients - User.</p> <p>120. Select <b>Configure Due Date</b>.</p> <p>121. Enter <b>2</b> as <b>Duration</b> and <b>Days</b> from the dropdown list <b>Unit of Time</b>, to mark the task as being due 2 days after it started.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in <i>Subject</i> and <i>Recipients</i> text. Sometimes during copy and paste, the spaces are added automatically. In that case the field will show red box. Remove the spaces.</p>	 <p>User Task Properties GENERAL DETAILS USER INTERFACE ATTRIBUTES</p> <p>*Priority High</p> <p>Display Texts</p> <p>*Subject Rework for '\${context.Title}'</p> <p>Description</p> <p>Recipients</p> <p>Users \${info.startedBy}</p> <p>Groups</p> <p>Due Date</p> <p><input checked="" type="checkbox"/> Configure Due Date</p> <p>*Due Date Based on Static Value</p> <p>*Duration 2</p> <p>*Unit of Time Days</p>

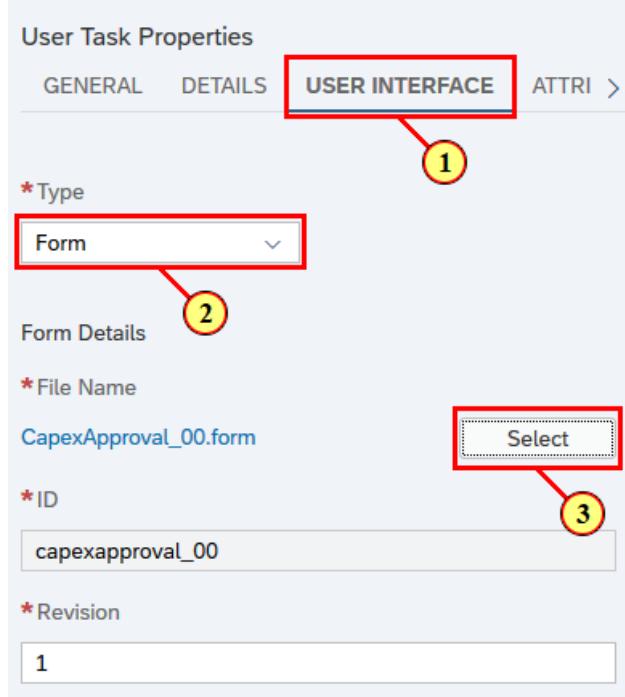
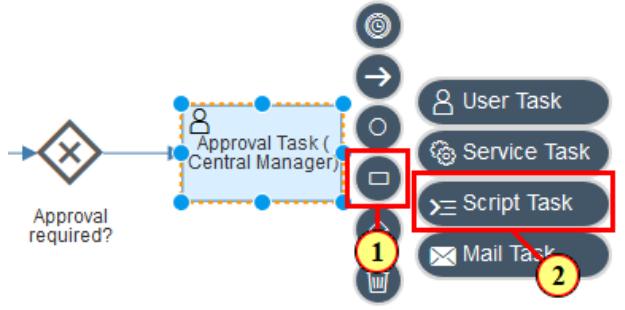
Explanation	Screenshot
<p>122. Select the <b>USER INTERFACE</b> tab.</p> <p>123. Select <b>Form</b> as Type.</p> <p>124. Click <b>Select</b> to choose an existing form.</p>	 <p>User Task Properties</p> <p>GENERAL DETAILS <b>USER INTERFACE</b> ATTRIBUTES</p> <p>*Type</p> <p>Form</p> <p>Form Details</p> <p>*File Name</p> <p>Create File Select</p> <p>*ID</p> <p>*Revision</p>
<p>125. Select the previously created form <b>CapexApproval_00</b></p> <p>126. Ensure that the <b>Revision</b> is 1.</p> <p>127. Select <b>OK</b>.</p>	 <p>Select Form</p> <p>*Project Name CapexProject-00/Capex-00</p> <p>*File Name forms/CAPEX_00/CapexApproval_00.form</p> <p>*Form Revision 1</p> <p>OK</p>
<p>128. Select the Exclusive Gateway <b>Approval result?</b> and drag an additional <b>Sequence Flow Connector...</b></p>	 <pre> graph LR     Start(( )) --&gt; Task[HandleApproval Local Manager]     Task --&gt; Gateway{Approval result?}     Gateway --&gt; EndReject(( ))     Gateway --&gt; ApprovalResult1(( ))     Gateway --&gt; ApprovalResult2(( ))     Gateway --&gt; Join(( ))     </pre>

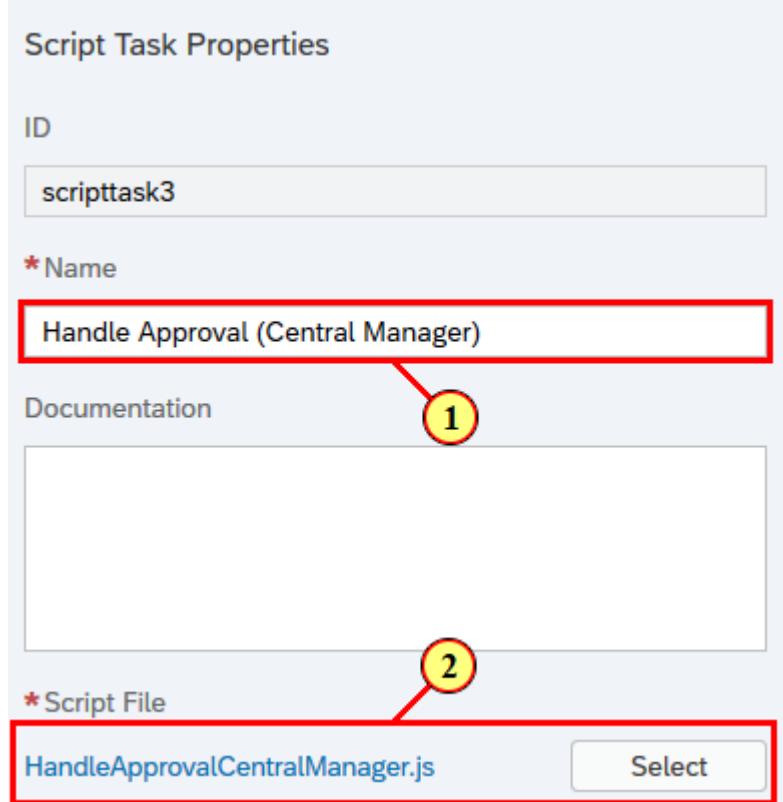
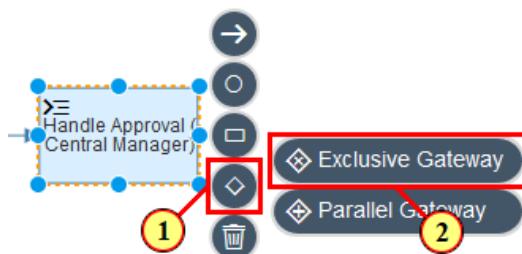
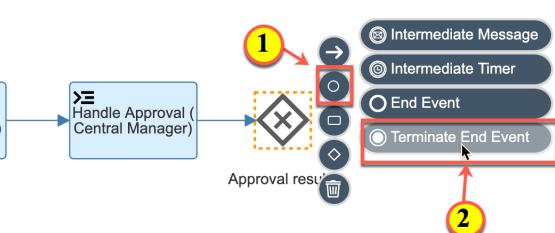
Explanation	Screenshot
<p>...onto the User Task <b>Rework Task</b>. Position it onto the highlighted middle dot.</p>	<pre> graph LR     Start(( )) -- Capex Request --&gt; Prep[Prepare Data]     Prep --&gt; Ret[Retrieve Approval Steps]     Ret --&gt; Decision{Approval required?}     Decision -- yes --&gt; ApprovalTask[Approval Task (Local Manager)]     ApprovalTask --&gt; Handle1[Handle Approval (Local Manager)]     Decision -- no --&gt; Join(( ))     Handle1 --&gt; EndReject((End (Reject)))     EndReject -- Reject --&gt; Decision     EndReject -- Approve --&gt; ApprovalResult{Approval result?}     ApprovalResult -- join --&gt; Join     </pre>
<p>129. Click on the <b>Connection Dot</b>  of the new Sequence Flow and drag it from the middle of the Exclusive Gateway... onto the lower edge of the Exclusive Gateway.</p>	
<p>130. Ensure your outgoing Sequence Flow Connectors look like this.</p>	<p>SequenceFlow12</p>

Explanation	Screenshot
<p>131. Enter <b>Rework</b> as Name.</p> <p>132. Enter <code>#{context.decision == "rework"}</code> as Condition.</p>	<p>Sequence Flow Properties</p> <p>ID: sequenceflow12</p> <p>* Name: Rework (highlighted by red box)</p> <p>Documentation: (empty)</p> <p>* Condition: \${context.decision == "rework"} (highlighted by red box)</p> <p><input type="checkbox"/> Default (highlighted by red box)</p>
<p>133. Select User Task <b>Rework Task</b></p> <p>134. Select <b>Sequence Flow Connector</b> and drag it...</p>	<p>Capex Request</p> <p>Sequence flow from Start Event to Rework Task.</p> <p>Rework Task selected (highlighted by red box).</p> <p>Sequence Flow Connector selected (highlighted by red box) and being dragged to the Prepare Data task.</p>
<p>...onto the Script Task <b>Prepare Data</b>.</p>	<p>Capex Request</p> <p>Sequence flow connected from Rework Task to Prepare Data task.</p>

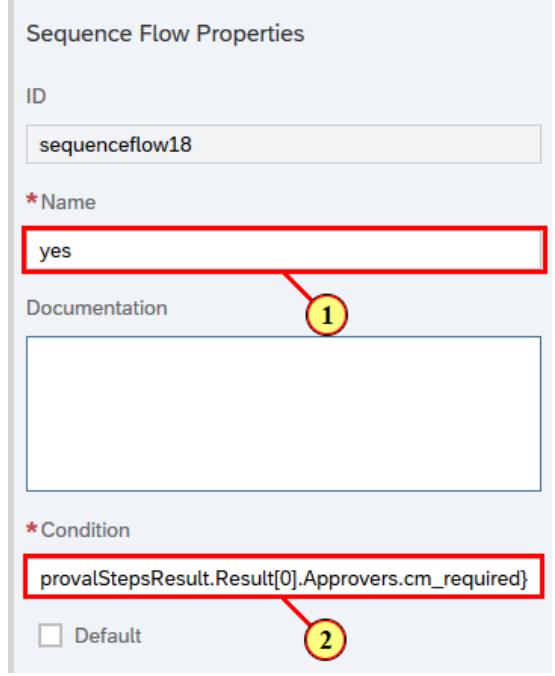
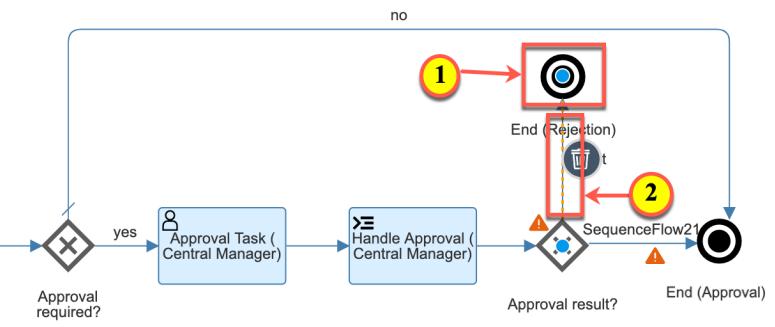
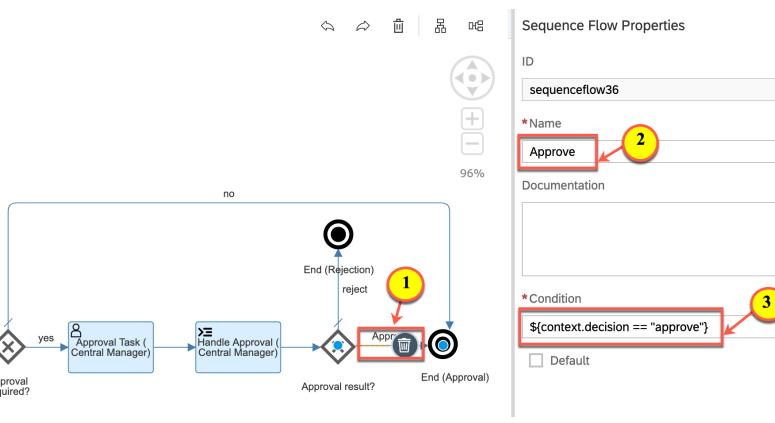
Explanation	Screenshot
<p>Ensure your workflow looks like this.</p>	<pre> graph LR     Start(( )) --&gt; PrepareData[Prepare Data]     PrepareData --&gt; RetrieveSteps[Retrieve Approval Steps]     RetrieveSteps --&gt; Decision{Approval required?}     Decision -- yes --&gt; ApprovalTask[Approval Task (Local Manager)]     ApprovalTask --&gt; HandleApproval[Handle Approval (Local Manager)]     HandleApproval --&gt; Decision     Decision -- no --&gt; EndReject((End (Reject)))     Decision --&gt; ReworkTask[Rework Task]     ReworkTask --&gt; Start     ApprovalTask --&gt; ApprovalResult{Approval result?}     ApprovalResult --&gt; ApprovalJoin((join))     ApprovalJoin --&gt; ApprovalApprove[Approve]     ApprovalApprove --&gt; EndAccept((End (Accept)))     ApprovalApprove --&gt; Decision     </pre>
<p>135. Select the exclusive gateway with name <b>join</b>.          136. Select <b>Add Gateway</b> .</p> <p>137. Select the <b>Exclusive Gateway</b> option .</p> <p>138. Name it as <b>Approval required?</b></p>	
<p>139. Select <b>Add Task</b> .</p> <p>140. Select <b>User Task</b> .</p> <p>141. Enter <b>Approval Task (Central Manager)</b> as <b>Name</b>.</p>	

Explanation	Screenshot
<p>142. Select the <b>Details</b> tab.</p> <p>143. Enter <b>Approval for '\${context.Title}' in your role as Central Manager</b> as <b>Subject</b>.</p> <p>144. Enter  <code> \${context.approvalStepsResult.Result[0].Approvers.cm_userid}, \${info.startedBy}</code> as <b>Recipients - Users</b>.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in <i>Subject and Recipients</i> text. Sometimes during copy and paste, the spaces are added automatically. In that case the field will show red box. Remove the spaces.</p> <p>Note: During development, the user tasks will be sent to the process initiator. Later, the user Id will be retrieved from SAP Cloud Platform Business Rules.</p>	 <p>User Task Properties</p> <p>GENERAL DETAILS USER INTERFACE ATTRIBUTES</p> <p>*Priority Medium</p> <p>*Subject Approval for '\${context.Title}' in your role as Central Manager</p> <p>Description</p> <p>Recipients</p> <p>Users \${context.approvalStepsResult.Result[0].Approvers.cm_userid}, \${info.startedBy}</p> <p>Groups</p> <p>Due Date</p> <p><input type="checkbox"/> Configure Due Date</p>

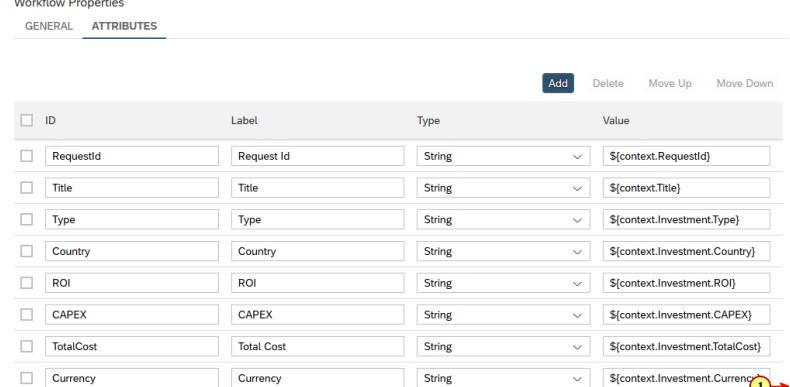
Explanation	Screenshot
<p>145. Select the <b>User Interface</b> tab.</p> <p>146. Select <b>Form</b> as <b>Type</b>.</p> <p>147. Use the <b>Select</b> button to choose the previously created form <b>capexapproval_00</b>.</p>	 <p>User Task Properties</p> <p>GENERAL DETAILS <b>USER INTERFACE</b> ATTRI &gt;</p> <p>* Type: Form</p> <p>Form Details</p> <p>* File Name: CapexApproval_00.form</p> <p>Select</p> <p>* ID: capexapproval_00</p> <p>* Revision: 1</p>
<p>148. Select <b>Add Task</b> on the central manager approval.</p> <p>149. Select the <b>Script Task</b>.</p>	 <p>Approval required?</p> <p>Approval Task (Central Manager)</p> <p>Script Task</p> <p>User Task</p> <p>Service Task</p> <p>Mail Task</p>

Explanation	Screenshot
<p>150. Enter <b>Handle Approval (Central Manager)</b> as Name.</p> <p>151. Click <b>Create File</b> to create a new file and <b>Name it HandleApprovalCentralManager</b>.</p> <p>152. Copy-paste the content of the file with the same name from the <i>Exercise 1 → Helper</i> folder in GitHub.</p> <p>Note: This script refers to the <b>Task ID</b> (usertask3) multiple times. This is the id of the User Task <b>Approval Task (Central Manager)</b> which was mentioned earlier. It is automatically assigned to each task. If your ID is different, then it needs to be replaced here.</p> <p>Note: To access the task decision the ID of the User Task is used. The difference in the two <b>Handle Approval</b> scripts are which task they refer to, and the 'Role' stored in the history element of workflow context.</p>	 <p>The screenshot shows the 'Script Task Properties' dialog. The 'Name' field contains 'Handle Approval (Central Manager)' with a red box around it. A yellow circle with the number '1' points to the 'Name' field. The 'Script File' field contains 'HandleApprovalCentralManager.js' with a red box around it. A yellow circle with the number '2' points to the 'Select' button next to the 'Script File' field.</p>
<p>153. Select <b>Add Gateway</b> .</p> <p>154. Select the <b>Exclusive Gateway</b> .</p> <p>155. Name it as <b>Approval result?</b></p>	 <p>The screenshot shows a workflow diagram with a user task labeled 'Handle Approval (Central Manager)'. An exclusive gateway is selected, indicated by a red box and a yellow circle with '1'. Another gateway is labeled 'Parallel Gateway' with a yellow circle with '2'.</p>
<p>156. Click <b>Approval result?</b> and Select <b>Add Event</b>.</p> <p>157. Select the <b>Terminate End Event</b>. Name the event as <b>End (Approval)</b>.</p>	 <p>The screenshot shows a workflow diagram with a user task labeled 'Handle Approval (Central Manager)'. An event gateway is selected, indicated by a red box and a yellow circle with '1'. A terminate end event is selected, indicated by a red box and a yellow circle with '2'.</p>

Explanation	Screenshot
<p>158. On the previous exclusive gateway with name <b>Approval required?</b> add a new <b>Sequence Flow Connector</b> and drag it onto the newly added <b>Terminate End Event</b>.</p> <p>Note: To make it look nice, move the blue connection dots on both sides to the upper edge of the icons and optionally drag the line upwards further upwards.</p>	
<p>159. Enter <b>no</b> as <b>Name</b>.</p> <p>160. Select it as <b>Default</b>.</p>	<p><b>Sequence Flow Properties</b></p> <p><b>ID</b>: sequenceflow23</p> <p><b>* Name</b>: no (circled with yellow circle 1)</p> <p><b>Documentation</b>: (empty field)</p> <p><b>Default</b>: checked (circled with yellow circle 2)</p>
<p>161. Select the Sequence Flow Connector  between <b>Approval required?</b> and <b>Approval Task (Central Manager)</b>.</p>	

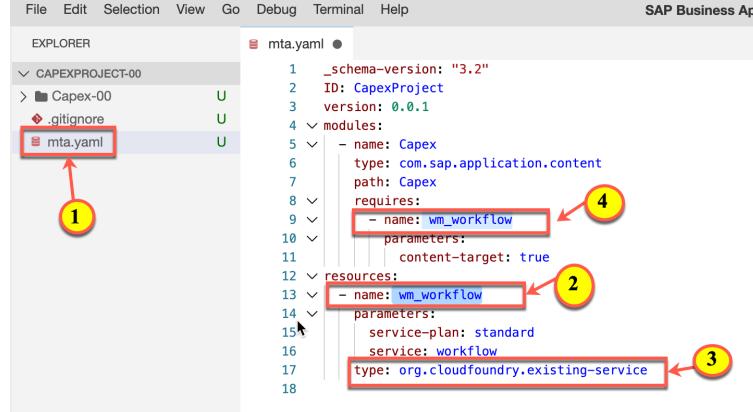
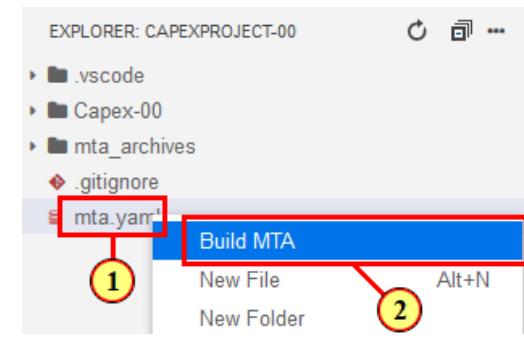
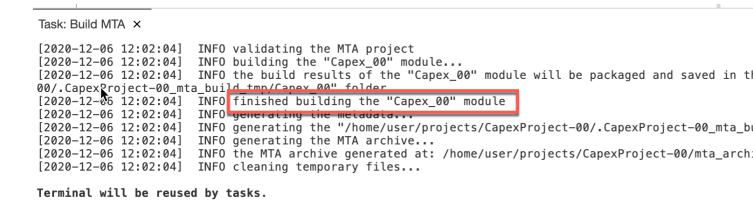
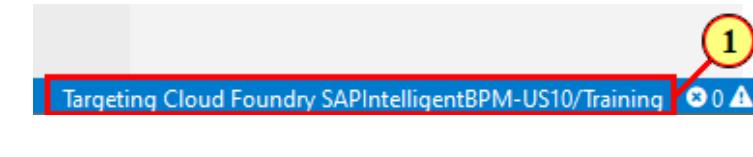
Explanation	Screenshot
<p>162. Enter <b>Yes</b> as Name.</p> <p>163. Enter  <code>#{context.approvalStepsResult.Result[0].Approvers.cm_required}</code> as Condition.</p> <p>⚠️ Be careful during copying. There are no spaces in the <b>Condition</b> text. Sometimes during copy, spaces are added automatically. In this case, then remove them.</p> <p>Note: how this accesses the field <i>cm_required</i>, for "approval required from Central Manager", whereas for the first approval, the field was called <i>lm_required</i>, for the Local Manager.</p>	 <p>The screenshot shows the 'Sequence Flow Properties' dialog. The 'Name' field contains 'yes'. The 'Condition' field contains the expression <code>#{context.approvalStepsResult.Result[0].Approvers.cm_required}</code>. A red box highlights the condition expression, and yellow circles numbered 1 and 2 point to the 'Name' field and the condition expression respectively.</p>
<p>164. Create a new <b>Terminating End Event</b> and place it above the <b>Approval result?</b> gateway. Name it <b>End (Rejection)</b>.</p> <p>165. Drag a new Sequence Flow Connection from <b>Approval result?</b> onto the <b>End (Rejection)</b> event. Name it <b>Reject</b> and select <b>Default</b> as condition.</p>	
<p>166. Select the Sequence Flow Connection from <b>Approval result?</b> onto the <b>End (Approval)</b></p> <p>167. Name it <b>Approve</b> and enter <code>#{context.decision == "approve"}</code> as condition.</p>	 <p>The screenshot shows the 'Sequence Flow Properties' dialog for 'SequenceFlow36'. The 'Name' field contains 'Approve'. The 'Condition' field contains the expression <code>#{context.decision == "approve"}</code>. A red box highlights the condition expression, and yellow circles numbered 2 and 3 point to the 'Name' field and the condition expression respectively.</p>

Explanation	Screenshot
<p>168. Drag a new Sequence Flow Connection from <b>Approval result?</b> onto the <b>Rework Task</b>.</p> <p>169. Adjust it that it starts directly downward by selecting the starting Connection Dot and move it to the lower edge of the Exclusive Gateway.</p>	<p>The screenshot shows a complex workflow diagram with multiple parallel paths involving approval tasks and handles. A new sequence flow has been added from the 'Approval result?' gateway to the 'Rework Task'. This flow is highlighted with a red box and a yellow circle labeled '1' at its start point.</p>
<p>170. Name it as <b>Rework</b> and enter <code>#{context.decision == "rework"}</code> as condition.</p>	<p>The screenshot shows the 'Sequence Flow Properties' pane for the newly created sequence flow. The name is set to 'Rework' (highlighted with a red box and yellow circle '2') and the condition is set to <code>#{context.decision == "rework"}</code> (highlighted with a red box and yellow circle '3').</p>
<p>171. Click anywhere on an empty space in the workflow editor to open <b>Workflow Properties</b>.</p> <p>172. In <b>Workflow Properties</b>, enter <b>Subject</b> as <b>CAPEX Approval for '#{context.Title}'</b></p> <p>173. Enter <b>Business Key</b> as <code>#{context.RequestId}</code></p> <p>174. Select the <b>ATTRIBUTES</b> tab.</p>	<p>The screenshot shows the 'Workflow Properties' pane. The subject is set to 'CAPEX Approval for #{context.Title}' (highlighted with a red box and yellow circle '2'), the business key is set to <code>#{context.RequestId}</code> (highlighted with a red box and yellow circle '3'), and the attributes tab is selected (highlighted with a red box and yellow circle '4').</p>
<p>Note: You can drag the vertical divider line of the <b>Workflow Properties</b> Pane to enlarge it.</p> <p>Now we will add attributes from the workflow context that will be exposed to be seen in Process Visibility scenario (in exercise 3 later)</p>	<p>The screenshot shows the 'Workflow Properties' pane with the 'ATTRIBUTES' tab selected. A yellow circle '1' highlights the tab indicator.</p>

Explanation	Screenshot
<p>175. Select <b>Add</b>.</p> <p>176. Enter <b>Currency</b> as <b>ID</b>.</p> <p>177. Enter <b>Currency</b> as <b>Label</b>.</p> <p>178. Enter  <b>Value:</b> \${context.Investment.Currency} as  <b>Value.</b></p>	
<p>179. In the same way, add the following fields:</p> <p><b>ID:</b> TotalCost  <b>Label:</b> Total Cost  <b>Value:</b> \${context.Investment.TotalCost}</p> <p><b>ID:</b> CAPEX  <b>Label:</b> CAPEX  <b>Value:</b> \${context.Investment.CAPEX}</p> <p><b>ID:</b> ROI  <b>Label:</b> ROI  <b>Value:</b> \${context.Investment.ROI}</p> <p><b>ID:</b> Country  <b>Label:</b> Country  <b>Value:</b> \${context.Investment.Country}</p> <p><b>ID:</b> Type  <b>Label:</b> Type  <b>Value:</b> \${context.Investment.Type}</p> <p><b>ID:</b> Title  <b>Label:</b> Title  <b>Value:</b> \${context.Title}</p> <p><b>ID:</b> RequestId  <b>Label:</b> Request Id  <b>Value:</b> \${context.RequestId}</p>	
<p>Note: Currently only String type is supported.</p> <p><b>⚠️</b> Be careful during copying. There are no spaces in the <b>Value</b> text. Sometimes during copy, spaces are added automatically. In this case, remove them. Check and remove leading and trailing spaces as well.</p>	

Explanation	Screenshot
<p>Your workflow should finally look like this.</p>	<pre> graph LR     Start(( )) --&gt; Prep[Prepare Data]     Prep --&gt; Ret[Retrieve Approval Steps]     Ret --&gt; Decision{Approval required?}     Decision -- yes --&gt; Local[Approval Task Local Manager]     Local --&gt; LocalEnd[End (Reject)]     Local -- no --&gt; Join(( ))     Join --&gt; Central[Approval Task Central Manager]     Central --&gt; CentralEnd[End (Reject)]     Central -- no --&gt; Decision     Decision -- no --&gt; EndAccept[End (Accept)]     </pre>
<p>Note: If you did not use <b>Auto Save</b>, ensure that all files are saved.</p>	<p>The screenshot shows the SAP IDEAS interface with the 'File' menu open. The 'Save All' option is highlighted with a red box and a yellow circle containing the number '1'. A red arrow points from the note text to this highlighted item.</p>

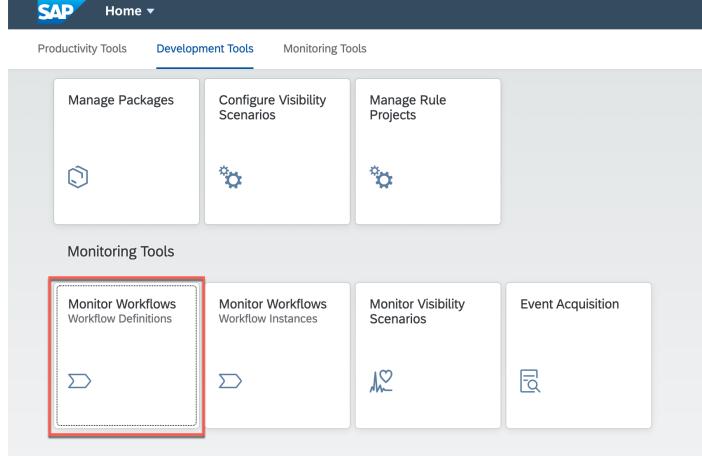
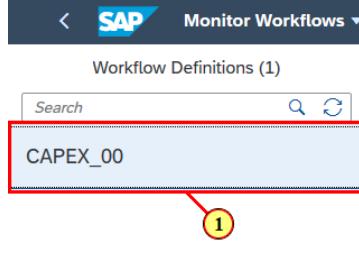
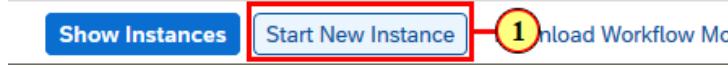
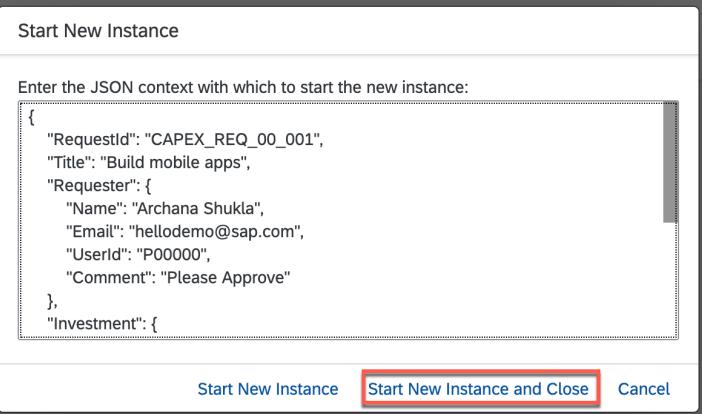
## BUILD AND DEPLOY WORKFLOW

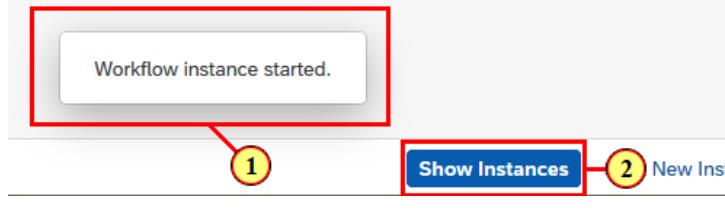
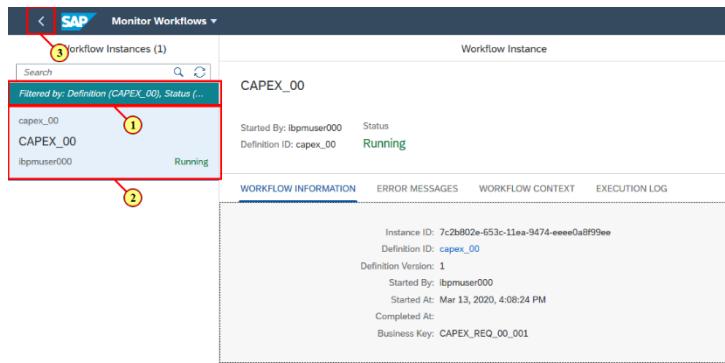
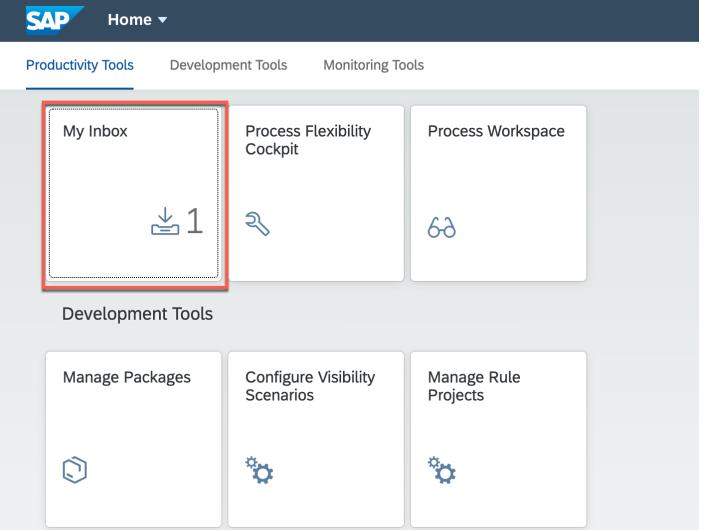
Explanation	Screenshot
<p>180. Double-click <b>mta.yaml</b> to open it.</p> <p>Note: The default MTA will mention a reference to a new workflow service instance. When you ran the Booster in the setup section, the workflow service instance is already created and so change the resource to the existing instance name.</p> <p>181. Change the resource name to <b>wm_workflow</b>.</p> <p>182. Change the resource type to <b>org.cloudfoundry(existing-service)</b>.</p> <p>183. Change the resource reference to <b>wm_workflow</b>.</p> <p>184. Close the tab and ensure the file is saved.</p>	 <pre> File Edit Selection View Go Debug Terminal Help SAP Business App EXPLORER mta.yaml CAPEXPROJECT-00 &gt; Capex-00 U .gitignore U mta.yaml U 1 _schema-version: "3.2" 2 ID: CapexProject 3 version: 0.0.1 4 modules: 5 - name: Capex 6 type: com.sap.application.content 7 path: Capex 8 requires: 9 - name: wm_workflow 10 parameters: 11   content-target: true 12 resources: 13 - name: wm_workflow 14 parameters: 15   service-plan: standard 16   service: workflow 17 type: org.cloudfoundry(existing-service)   </pre>
<p>Before you build, ensure that there are no errors shown in the project. If there are red indicators, then open <b>View → Problems</b> menu option to see the modelling errors. Fix those first.</p> <p>185. To build the project, right-click on <b>mta.yaml</b> to open the context menu.</p> <p>186. Select <b>Build MTA</b>.</p>	
<p>187. Ensure that the <b>Build MTA</b> task exits with successful message, indicating the build is completed.</p>	 <pre> [2020-12-06 12:02:04] INFO validating the MTA project [2020-12-06 12:02:04] INFO building the "Capex_00" module... [2020-12-06 12:02:04] INFO the build results of the "Capex_00" module will be packaged and saved in the "/home/user/projects/CapexProject-00/mta_capex_00" folder [2020-12-06 12:02:04] INFO finished building the "Capex_00" module [2020-12-06 12:02:04] INFO generating the "/home/user/projects/CapexProject-00/.CapexProject-00_mta_bu... [2020-12-06 12:02:04] INFO generating the MTA archive... [2020-12-06 12:02:04] INFO the MTA archive generated at: /home/user/projects/CapexProject-00/mta_archi... [2020-12-06 12:02:04] INFO cleaning temporary files...   </pre> <p>Terminal will be reused by tasks.</p>
<p>188. Ensure that a cloud tenant has already been configured. It will be shown in the bottom left corner on a blue bar. If it is not configured yet, then click on the blue bar and follow the wizard to enter the details.</p>	

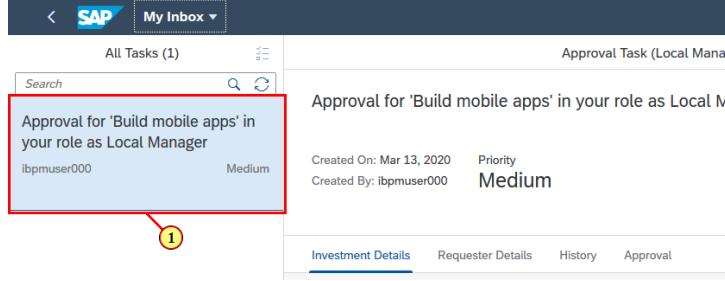
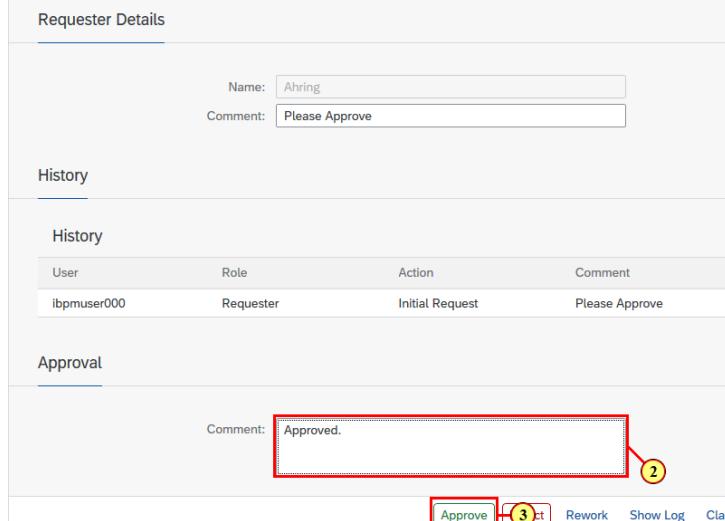
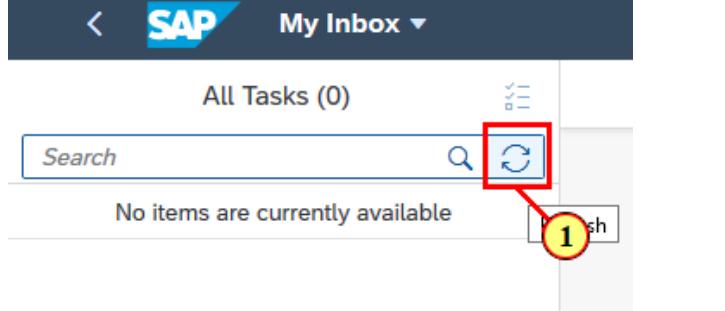
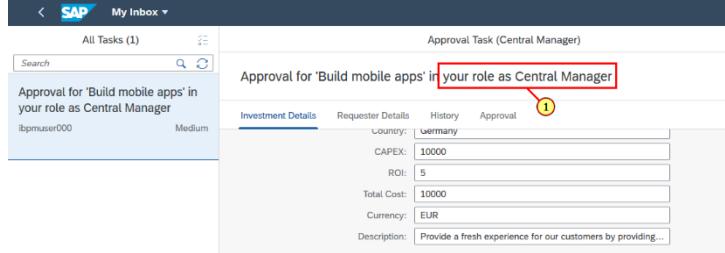
Explanation	Screenshot
<p>You can get the need information about the Login URL, organization name and space name from you trial account.</p> <p>You have to login with your trial user email and password.</p>	<p>The screenshot shows the SAP Cloud Platform Cockpit interface. In the left sidebar, under 'Cloud Foundry', the 'Cloud Foundry Environment' tab is selected. The main area displays the trial account details. A red box highlights the 'Org Name' field containing 'db73ff38trial', which is labeled 'Organisation Name'. Another red box highlights the 'API Endpoint' field containing 'https://api.cf.eu10.hana.ondemand.com', which is labeled 'Login URL'. A third red box highlights the 'Space' table row for 'dev', which is labeled 'Space Name'.</p>
<p>[Optional] If you get this error which means you have 2FA enabled for this user in your account.</p> <p>In that case,</p> <ul style="list-style-type: none"> <li>open a <b>New Terminal</b> window and enter command <b>cf login --sso</b></li> <li>You will then be given a URL to get <b>Temporary Authentication Code</b></li> <li>Copy and open the URL in browser window, follow the instructions to copy the authorization code and paste in the terminal and click Enter. (note: Just paste is and click Enter. The code wont with be shown).</li> <li>If the code is correct, you will be prompted to select the organization else it will throw <b>Invalid passcode</b>. For latter case, get the passcode again, paste and click Enter key.</li> <li>Enter the organization index number or the organization name from the list.</li> <li>If you have only 1 space then it will automatically be selected, else it will ask you to enter the space as well.</li> <li>When your connection is done, you will see the connected cloud account information in the blue-bar.</li> </ul>	<p>The screenshot shows a terminal session with the following output:</p> <pre> user@workspaces-ws-6229h-deployment-7bffbcb8494-r4fc5: ~/projects × user: projects \$ cf login --sso API endpoint: https://api.cf.eu10.hana.ondemand.com Temporary Authentication Code ( Get one at https://login.cf.eu10.hana.ondemand.com/passcode ): █  user@workspaces-ws-6229h-deployment-7bffbcb8494-r4fc5: ~/projects × user: projects \$ cf login --sso API endpoint: https://api.cf.eu10.hana.ondemand.com Temporary Authentication Code ( Get one at https://login.cf.eu10.hana.ondemand.com/passcode ): Authenticating... Invalid passcode  Temporary Authentication Code ( Get one at https://login.cf.eu10.hana.ondemand.com/passcode ): Authenticating... OK  Select an org: 1. bpm-beta_inbox-consumer-9 2. BPM_bpm_services 3. bpm-cf_bpm-workflow 4. bpm-cf_bpm-workflow-consumer 5. RDM_rP_Fnschlammt_1  Org (enter to skip): 11 Targeted org db73ff38trial Targeted space dev  API endpoint: https://api.cf.eu10.hana.ondemand.com (API version: 3.91.0) User: ██████████ Org: db73ff38trial Space: dev user: projects \$ █   </pre> <p>A red box highlights the 'API endpoint' line in the terminal output, which is labeled 'Login URL'.</p>

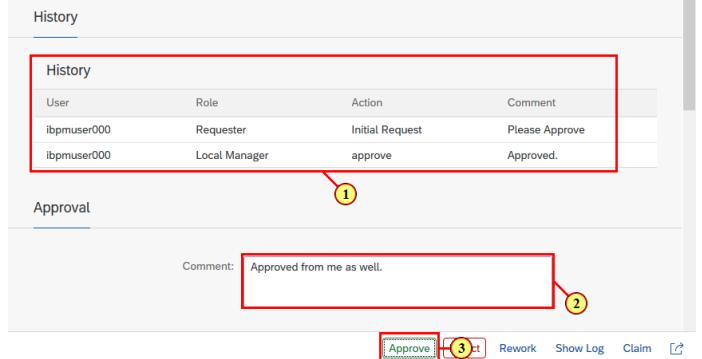
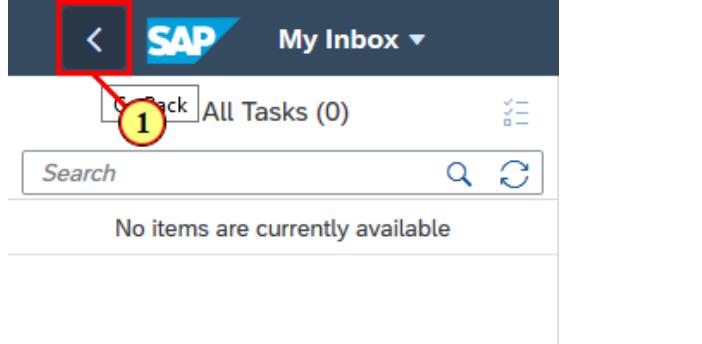
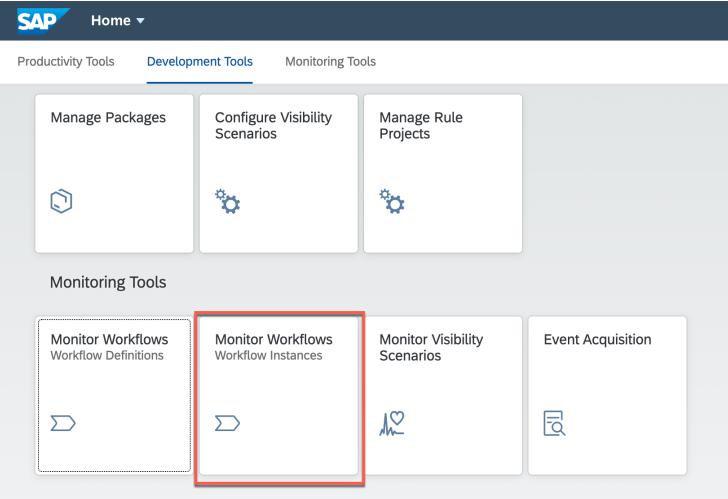
Explanation	Screenshot
<p>189. Select the newly created directory <b>mta_archives</b> to open it.</p> <p>190. Right-click on the created file <b>CapexProject-00_0.0.1.mtar</b> to open the context menu.</p> <p>191. Select <b>Deploy MTA Archive</b>.</p>	
<p>192. Ensure that the <b>Deploy MTA Archive</b> task exits with the successful message indicating that the deployment has finished.</p> <p>Note: Your deployment console should look similar to this with the "<b>Process finished</b>" in the second-to-last line, indicating success.</p> <p>If there are any errors then fix the error, build and deploy the workflow project again.</p>	<pre>&gt; Executing task: cf deploy /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar &lt; Deploying multi-target app archive /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar in org db73ff38trial / sp dev as jsap.com... There is an ongoing operation for multi-target app CapexProject-00. Do you want to abort it? (y/n)&gt; y Executing action 'abort' on operation 9b2ab30e-37b0-11eb-b80e-eeee0a91282a... OK Operation ID: b6cd7a41-37c0-11eb-b95c-eeee0a9af9a4 Deploying in org "db73ff38trial" and space "dev" Detected MTA schema version: "3" No deployment detected, this is initial deployment Detected new MTA services: "wm_workflow" Service key "Capex_00-wm_workflow-credentials" for service "wm_workflow" already exists Uploading content module "Capex_00" in target service "wm_workflow"... Deploying content module "Capex_00" in target service "wm_workflow"... Content deployment module "Capex_00" failed. Download the logs "Capex_00_wm_workflow". Some deployment modules failed. Try again or skip services, because the command line option "--delete-services" is not specified. Process finished. Use "cf dmolt -i b6cd7a41-37c0-11eb-b95c-eeee0a9af9a4" to download the logs of the process.  Terminal will be reused by tasks. Task 'Deploy MTA Archive (/)' was terminated by signal SIGPIPE.</pre>
<p>[Optional]</p> <p>Note: If the deployment fails saying "<b>Content deployment for module Capex_00 failed</b>", then carefully go through the deployment message in the terminal itself, it will let you know which part of the workflow model is not correctly configured.</p> <p>Note: Once you have fixed the errors in the workflow model, you have to build and deploy again. On doing so enter <b>y</b> if you get option saying: <b>There is an ongoing operation for multi-target app CapexProject-00. Do you want to abort it? (y/n)?</b></p>	<pre>Content deployment for module "Capex_00" failed. Download the logs "Capex_00_wm_workflow" via the dmolt command and check them for more information. A step of the process has failed. Retrying it may solve the issue. Proceeding with automatic retry... (3 of 3 attempts left) Deployment content module "Capex_00" in target service "wm_workflow"... [Capex_00] [FATAL] Deployment of file Id d1d8bede-37bd-11eb-9381-2b02b728ee4cd Failed, Error details: while trying to invoke the method com.sap.bpm.wds.core.model.WorkflowModuleContent.getSubContents() of a null object loaded from local variable 'obj' Content deployment module "Capex_00" failed. Download the logs "Capex_00_wm_workflow". A step of the process has failed. Retrying it may solve the issue. Proceeding with automatic retry... (3 of 3 attempts left) Deployment content module "Capex_00" in target service "wm_workflow"... [Capex_00] [FATAL] Deployment of file Id d1d8bede-37bd-11eb-9381-2b02b728ee4cd Failed, Error details: while trying to invoke the method com.sap.bpm.wds.core.model.WorkflowModuleContent.getSubContents() of a null object loaded from local variable 'obj' Content deployment module "Capex_00" failed. Download the logs "Capex_00_wm_workflow". A step of the process has failed. Retrying it may solve the issue. Proceeding with automatic retry... (3 of 3 attempts left) Use "cf deploy -i ca43f571-37bd-11eb-b80e-eeee0a91282a -a abort" to abort the process. Use "cf deploy -i ca43f571-37bd-11eb-b80e-eeee0a91282a -a retry" to retry the process. Use "cf dmolt -i ca43f571-37bd-11eb-b80e-eeee0a91282a" to download the logs of the process.  Terminal will be reused by tasks.  &gt; Executing task: cf deploy /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar &lt; Deploying multi-target app archive /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar in org db73ff38trial / sp dev as jsap.com... There is an ongoing operation for multi-target app CapexProject-00. Do you want to abort it? (y/n)&gt; y</pre>

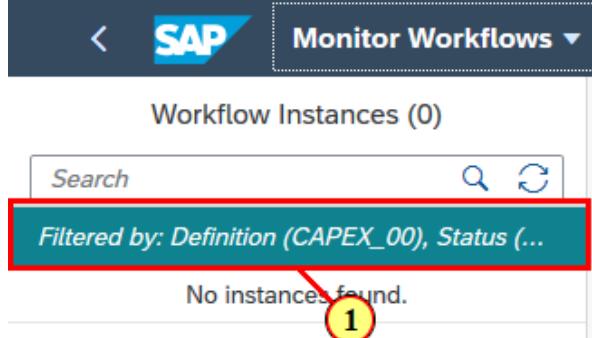
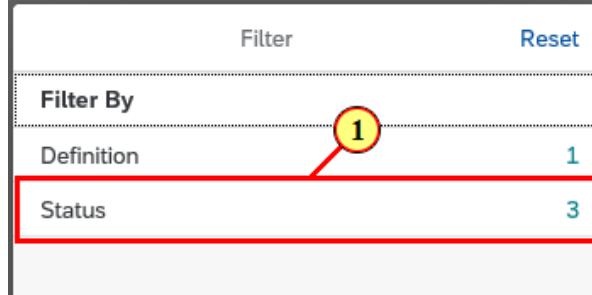
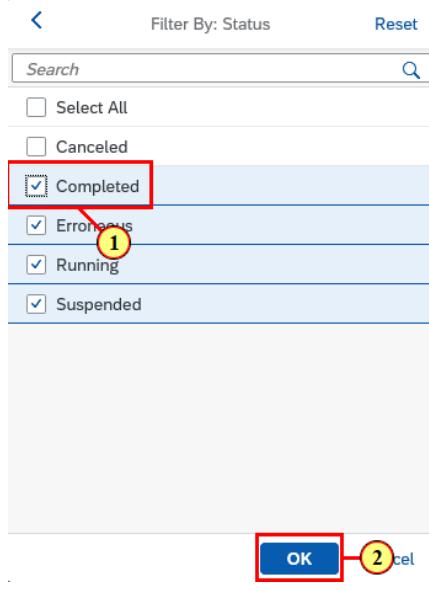
## EXECUTE AND MONITOR WORKFLOW

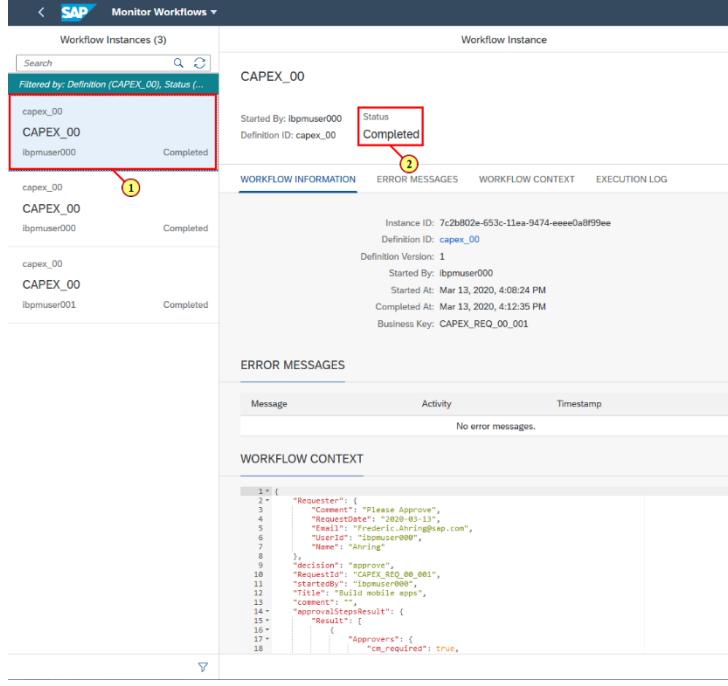
Explanation	Screenshot
<ol style="list-style-type: none"> <li>1. Open the <b>Workflow Management Launchpad</b></li> <li>2. <b>Log On</b> with your trial username and password.</li> <li>3. Select the tile <b>Monitor Workflows - Workflow Definitions</b>.</li> </ol>	
<p>Note: <b>Monitor Workflows</b> application has Master-Detail layout where you see all the workflows deployed on left. Ensure that the workflow you just deployed - <b>CAPEX_00</b> - is existing.</p> <ol style="list-style-type: none"> <li>4. Select the workflow.</li> </ol>	
<ol style="list-style-type: none"> <li>5. Select <b>Start New Instance</b>.</li> </ol> <p>Note: This way of starting a workflow is mainly used for testing. Workflows can be started by end users using a Start Form or a custom UI5 application, and by other systems using the <a href="#">Workflow API</a>.</p>	
<ol style="list-style-type: none"> <li>6. Check the start payload</li> <li>7. Click <b>Start New Instance and Close</b></li> </ol> <p>Note: The other <i>Start New Instance</i> option is to start multiple instances of the workflow from the same popup.</p>	

Explanation	Screenshot
<p>A popup confirms that the workflow instance has been started.</p> <p>8. Select <b>Show Instances</b> to switch to the <b>Monitor Workflows - Workflow Instances</b> application where you can view all the running, completed, suspended workflow instances and their details.</p>	
<p>Note: When coming from <b>Show Instances</b>, a <b>Filter</b> is automatically added to the specific <b>Workflow Definition</b>.</p> <p>9. Select the newly started workflow instance and look at its details.</p> <p>10. Select <b>SAP icon</b> to go back to the Home Page of the Workflow Management Launchpad.</p>	
<p>11. Select the tile for <b>My Inbox</b>. It should show at least one new task.</p>	

Explanation	Screenshot								
<p>12. Ensure that in the list view the new task "Approval for..." is selected.</p>	 <p>Approval Task (Local Manager)</p> <p>Approval for 'Build mobile apps' in your role as Local Manager</p> <p>ibpmuser000 Medium</p> <p>Created On: Mar 13, 2020 Priority: Medium Created By: ibpmuser000</p> <p>Investment Details Requester Details History Approval</p>								
<p>13. Scroll down to the <b>Approval</b> section.</p> <p>14. Enter a <b>Comment</b> of your choice.</p> <p>15. Click <b>Approve</b>.</p>	 <p>Requester Details</p> <p>Name: Ahring Comment: Please Approve</p> <p>History</p> <table border="1"> <thead> <tr> <th>User</th> <th>Role</th> <th>Action</th> <th>Comment</th> </tr> </thead> <tbody> <tr> <td>ibpmuser000</td> <td>Requester</td> <td>Initial Request</td> <td>Please Approve</td> </tr> </tbody> </table> <p>Approval</p> <p>Comment: Approved.</p> <p>Approve Rework Show Log Claim</p>	User	Role	Action	Comment	ibpmuser000	Requester	Initial Request	Please Approve
User	Role	Action	Comment						
ibpmuser000	Requester	Initial Request	Please Approve						
<p>16. In this exercise, the user task in the workflow has been configured to send all tasks in your inbox. To see the second approval task, select <b>Refresh</b> .</p>	 <p>All Tasks (0)</p> <p>No items are currently available</p>								
<p>Note: You can now see the task for the Central Manager.</p>	 <p>Approval Task (Central Manager)</p> <p>Approval for 'Build mobile apps' in your role as Central Manager</p> <p>ibpmuser000 Medium</p> <p>Investment Details Requester Details History Approval</p> <p>Country: Germany CAPEX: 10000 ROI: 5 Total Cost: 10000 Currency: EUR Description: Provide a fresh experience for our customers by providing...</p>								

Explanation	Screenshot
<p>Note: The <b>History</b> table is filled with the comment entered in the previous task.</p> <p>17. Enter again a <b>Comment</b> of your choice.</p> <p>18. Click <b>Approve</b>.</p>	
<p>19. Select <b>SAP icon</b> to go back to the Workflow Management Home Page.</p>	
<p>20. Select the <b>Monitor Workflows - Workflow Instances</b> tile.</p>	

Explanation	Screenshot
<p>21. Be default, <b>Monitor Instances</b> application only shows Running, Suspended and Erroneous workflow instance. To see the completed workflow, click on the line showing the currently active <b>Filters</b> or select the <b>Filter</b> icon at the bottom of the screen.</p>	 <p>The screenshot shows the SAP Monitor Workflows interface. At the top, it says "Monitor Workflows". Below that, it displays "Workflow Instances (0)". There is a search bar with a magnifying glass icon. A red box highlights the status filter section, which says "Filtered by: Definition (CAPEX_00), Status (...)" with a yellow circle labeled "1" pointing to the filter icon. Below this, it says "No instances found." with another yellow circle labeled "1" pointing to the text.</p>
<p>22. In the popup, select <b>Status</b>.</p>	 <p>The screenshot shows a "Filter By" dialog. It has a "Filter" button and a "Reset" button. Under "Filter By", there is a "Definition" field with a value of "1" and a "Status" field with a value of "3". A red box highlights the "Status" field, and a yellow circle labeled "1" points to the "Definition" field.</p>
<p>23. Select <b>Completed</b> to also show already completed workflow instances.</p> <p>24. Select <b>OK</b> to activate the changed filter settings.</p>	 <p>The screenshot shows a "Filter By: Status" dialog. It has a "Search" bar and an "OK" button. Below the search bar are several filter options: "Select All", "Canceled", "Completed" (which is checked and highlighted with a red box and yellow circle labeled "1"), "Erroneous", "Running" (which is checked), and "Suspended". The "OK" button is highlighted with a red box and yellow circle labeled "2".</p>

Explanation	Screenshot												
<p>26. Ensure the correct instance is selected.</p> <p>27. Look at the details of the completed workflow instance.</p> <p><b>Congratulations! You completed the Workflow exercise.</b></p> <p>In this exercise, you learnt how to:</p> <ul style="list-style-type: none"> <li>○ Model workflow from <b>SAP Business Application Studio</b>.</li> <li>○ Build &amp; deploy workflow</li> <li>○ Start workflow instance from <b>Monitor Definitions</b></li> <li>○ Track the workflow instances from <b>Monitor Instances</b></li> <li>○ Process your work items from <b>My Inbox</b></li> </ul>	 <p>The screenshot shows the SAP Monitor Workflows interface. In the top navigation bar, it says "Workflow Instances (3)". Below this, a table lists three workflow instances:</p> <table border="1"> <thead> <tr> <th>Instance ID</th> <th>Definition ID</th> <th>Status</th> </tr> </thead> <tbody> <tr> <td>capex_00</td> <td>CAPEX_00</td> <td>Completed</td> </tr> <tr> <td>CAPEX_00</td> <td>CAPEX_00</td> <td>Completed</td> </tr> <tr> <td>ibpmuser001</td> <td>CAPEX_00</td> <td>Completed</td> </tr> </tbody> </table> <p>A red box highlights the first row (capex_00). A yellow circle with the number 1 is on the "Status" column of the first row. Another yellow circle with the number 2 is on the "Status Completed" text in the "Workflow Instance" section below the table.</p> <p>In the "Workflow Instance" section, the following details are shown:</p> <ul style="list-style-type: none"> <li>Started By: ibpmuser000</li> <li>Definition ID: capex_00</li> <li>Status: Completed</li> <li>Instance ID: 7c2b802e-653c-11ea-9474-e000a08f99ee</li> <li>Definition Version: 1</li> <li>Started By: ibpmuser000</li> <li>Started At: Mar 13, 2020, 4:08:24 PM</li> <li>Completed At: Mar 13, 2020, 4:12:35 PM</li> <li>Business Key: CAPEX_REQ_00_001</li> </ul> <p>Below this, there are sections for "ERROR MESSAGES" (No error messages) and "WORKFLOW CONTEXT". The "WORKFLOW CONTEXT" section displays a JSON log entry:</p> <pre> 1: {   "Requester": {     "Comment": "Please Approve",     "RequestDate": "2020-03-13",     "RequesterEmail": "ibpmuser001@sap.com",     "UserId": "ibpmuser001",     "Name": "Amit"   },   "decision": "approve",   "RequestId": "CAPEX_REQ_00_001",   "Requester": "ibpmuser001",   "Title": "Build mobile apps",   "comment": "",   "ApprovalResult": {     "Result": [       {         "Approver": {           "ck_required": true,         }       }     ]   } } </pre>	Instance ID	Definition ID	Status	capex_00	CAPEX_00	Completed	CAPEX_00	CAPEX_00	Completed	ibpmuser001	CAPEX_00	Completed
Instance ID	Definition ID	Status											
capex_00	CAPEX_00	Completed											
CAPEX_00	CAPEX_00	Completed											
ibpmuser001	CAPEX_00	Completed											

## SUMMARY

You have completed the exercise!

You are now able to:

1. Create a new workflow in **Business Application Studio**.
2. Add & configure **service task**, **script task**, **user task** and **gateways** in the workflow model.
3. **Build** and **deploy** workflow from Business Application Studio
4. Test the workflow using **Monitor Workflow | Workflow Definition** application.
5. Track and monitor the workflow using **Monitor Workflow | Workflow Instances** application.
6. View user tasks and take action from **My Inbox** application.

[www.sap.com/contactsap](http://www.sap.com/contactsap)

© 2020 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. See [www.sap.com/copyright](http://www.sap.com/copyright) for additional trademark information and notices.