

PUBLIC

Build a Workflow from Scratch with SAP Cloud Platform Workflow Management

DEV163

Exercise 1 | Workflow
Archana Shukla / SAP

TABLE OF CONTENTS

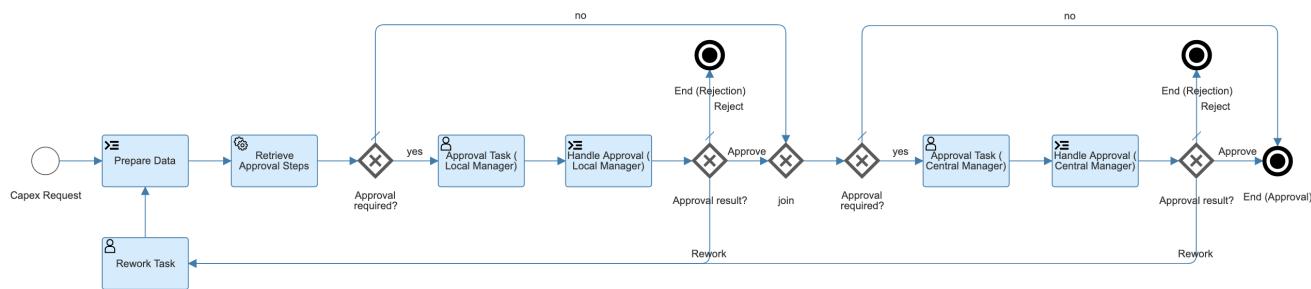
OVERVIEW	3
MODEL WORKFLOW USING BUSINESS APPLICATION STUDIO.....	4
BUILD AND DEPLOY WORKFLOW	46
EXECUTE AND MONITOR WORKFLOW.....	49
SUMMARY	54

OVERVIEW

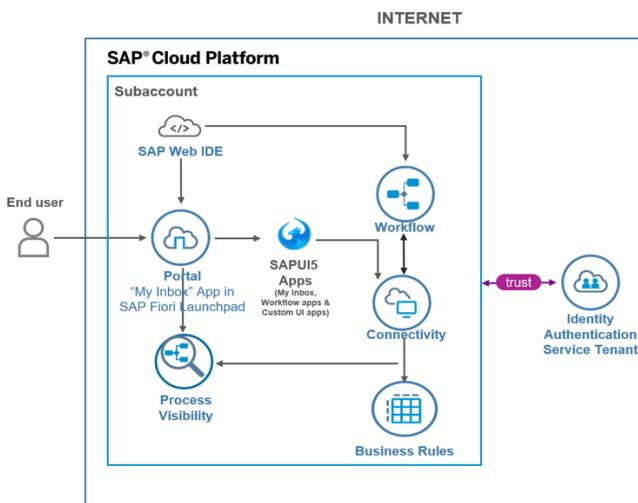
Estimated time: 90 minutes

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.

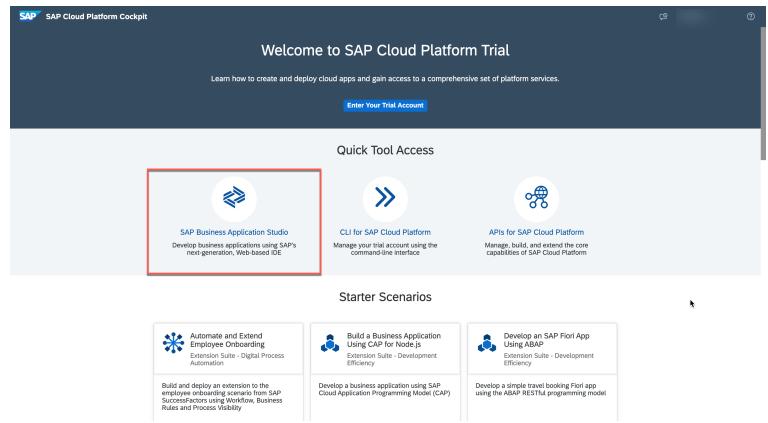
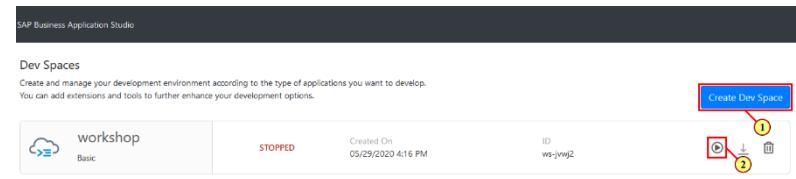
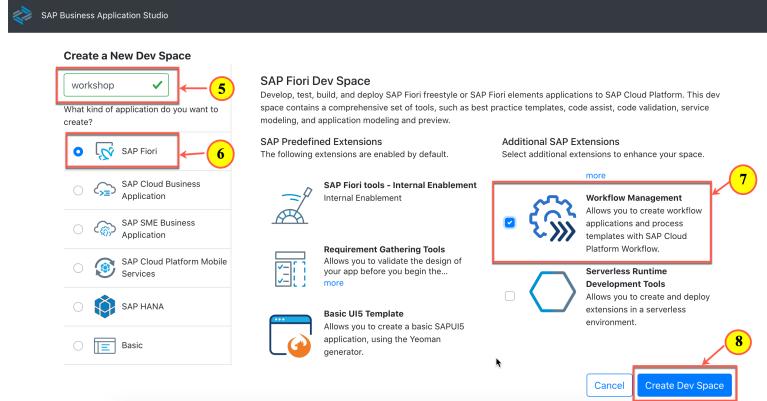
There are many challenges with capital expenditure process and companies are always looking for managing investment decisions in a transparent way. Automating capital expenditure approval process enables customer to accelerate investment decisions and to bring the agility into every step and decision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

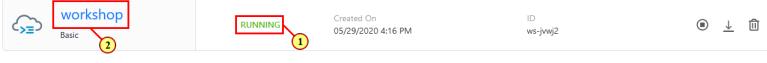
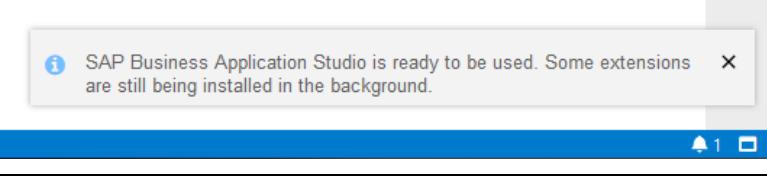
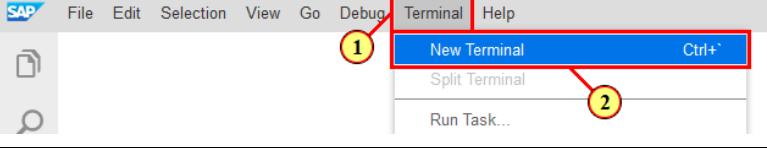
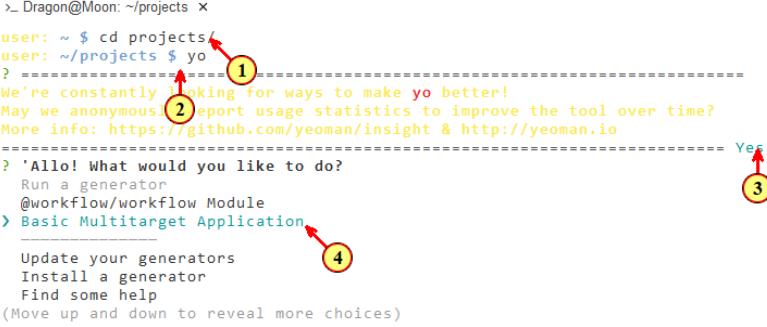
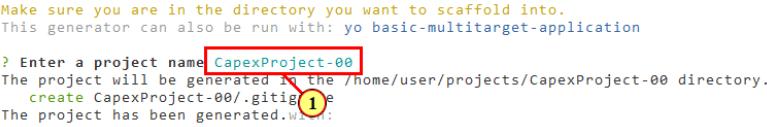


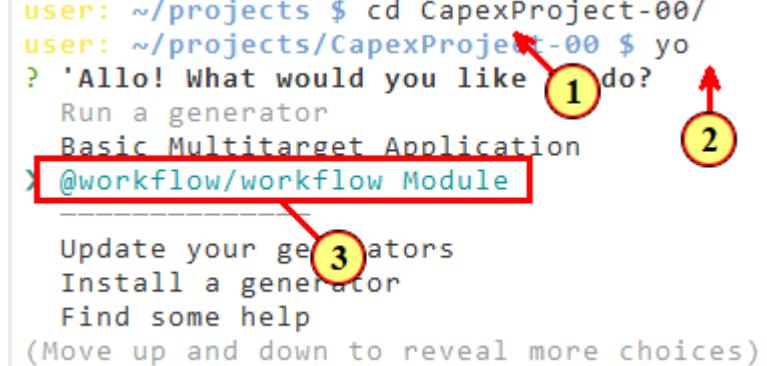
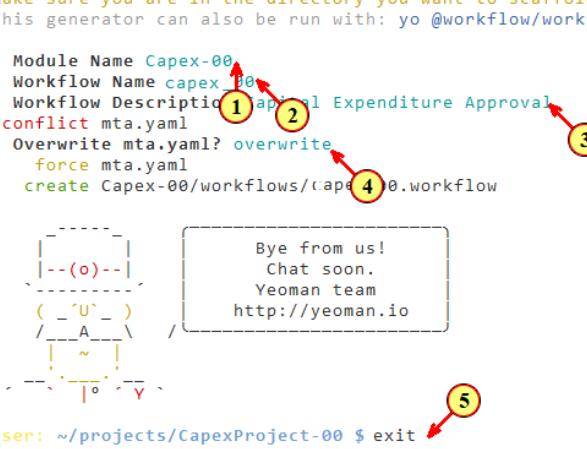
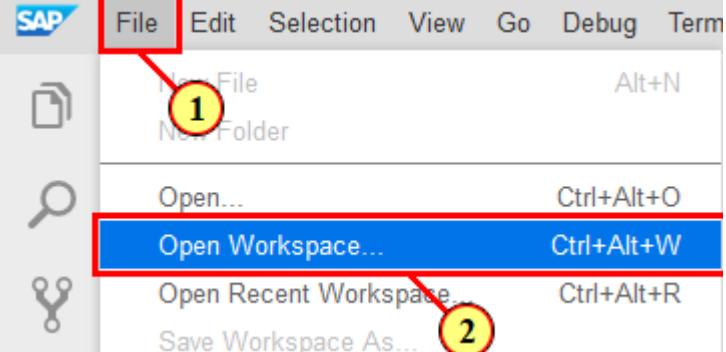
The below solution diagram explains how customer can automate CAPEX scenario using SAP Cloud Platform.

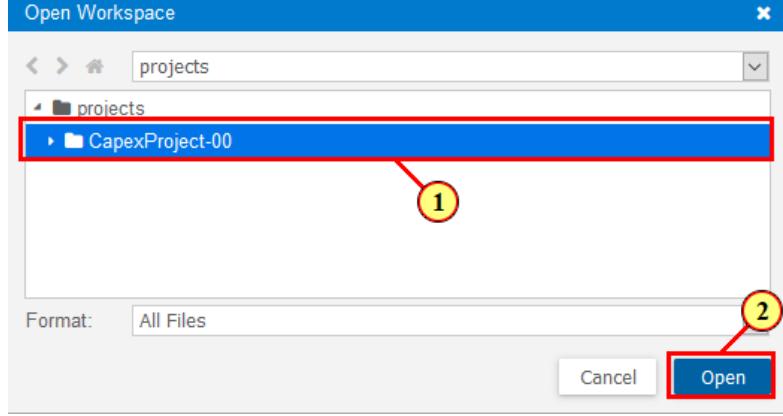
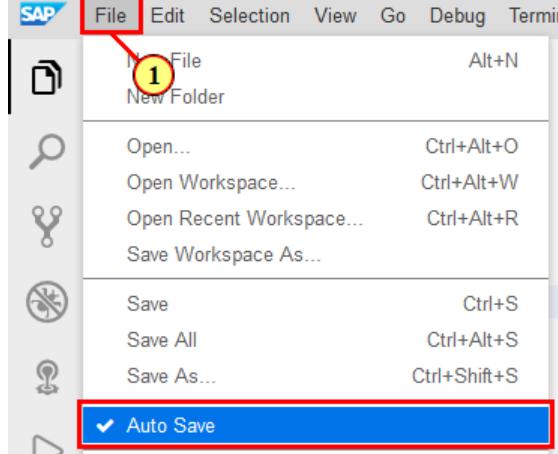
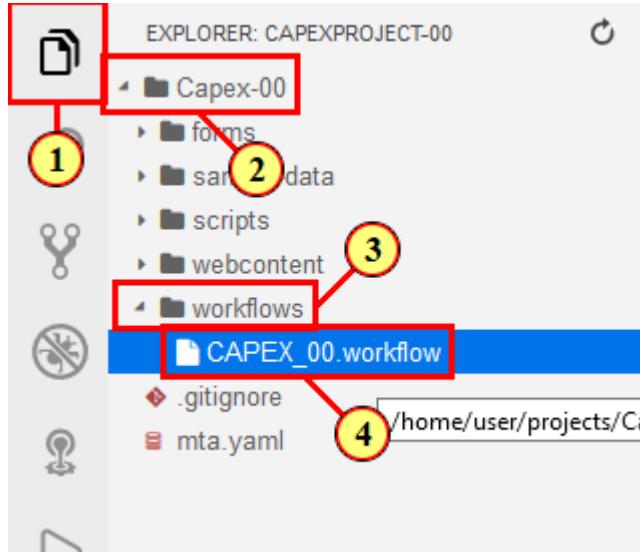


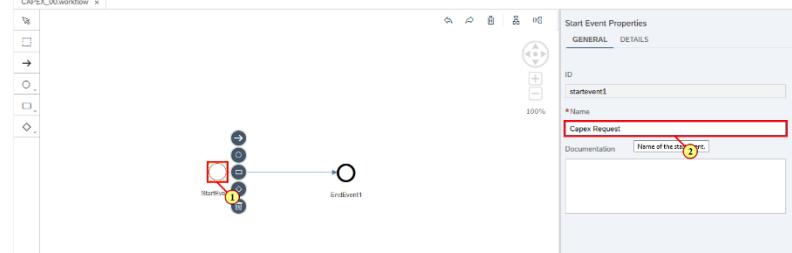
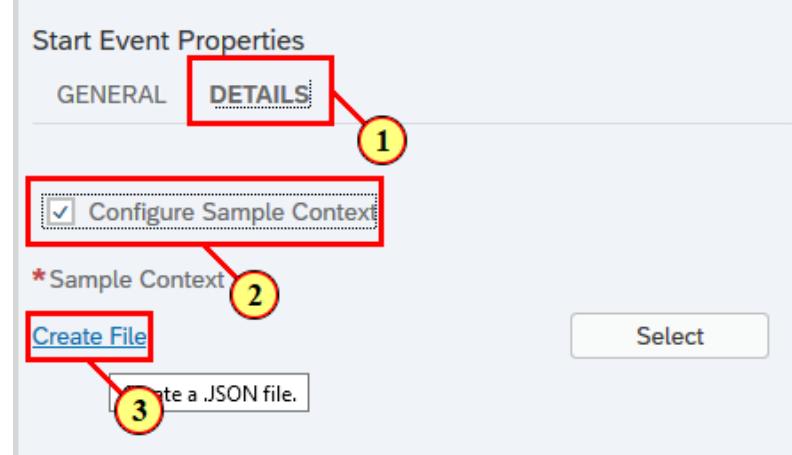
MODEL WORKFLOW USING BUSINESS APPLICATION STUDIO

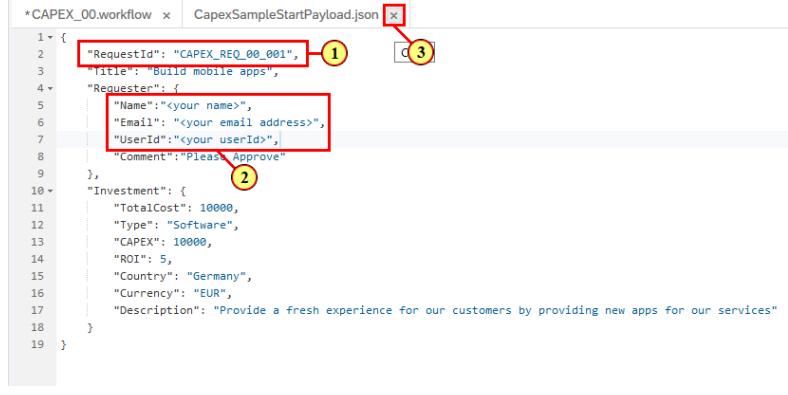
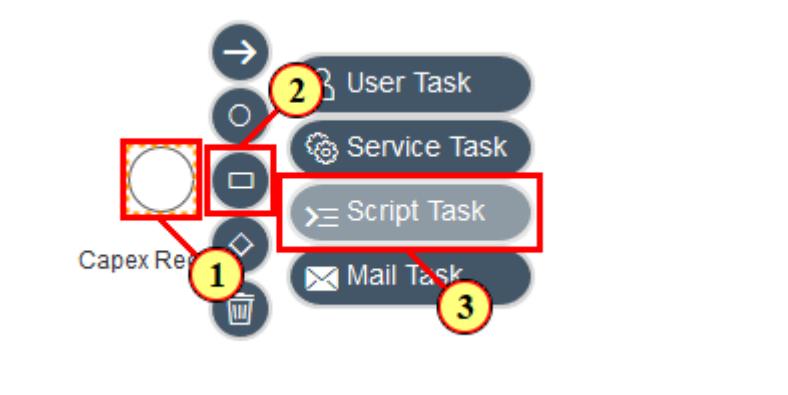
Explanation	Screenshot
<p>1. Logon to SAP Cloud Platform Trial Home Page with your trial username and password.</p> <p>https://cockpit.hanatrial.ondemand.com/cockpit/#/home/trial</p> <p>2. Choose SAP Business Application Studio.</p>	
<p>Each SAP Business Application Studio Development Space is a virtual machine hosting your project files, securely separated from other spaces.</p> <p>3. If you have no dev space, click the Create Dev Space and follow instructions in step 5.</p> <p>4. Else select PLAY to start the space if you see the status as STOPPED.</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 workshop as the new Dev Space name.</p> <p>6. Ensure Application Type is selected as SAP Fiori.</p> <p>7. Select the extension Workflow Management. Note: This extension will enable the workflow modelling features.</p> <p>8. Select Create Dev Space.</p>	

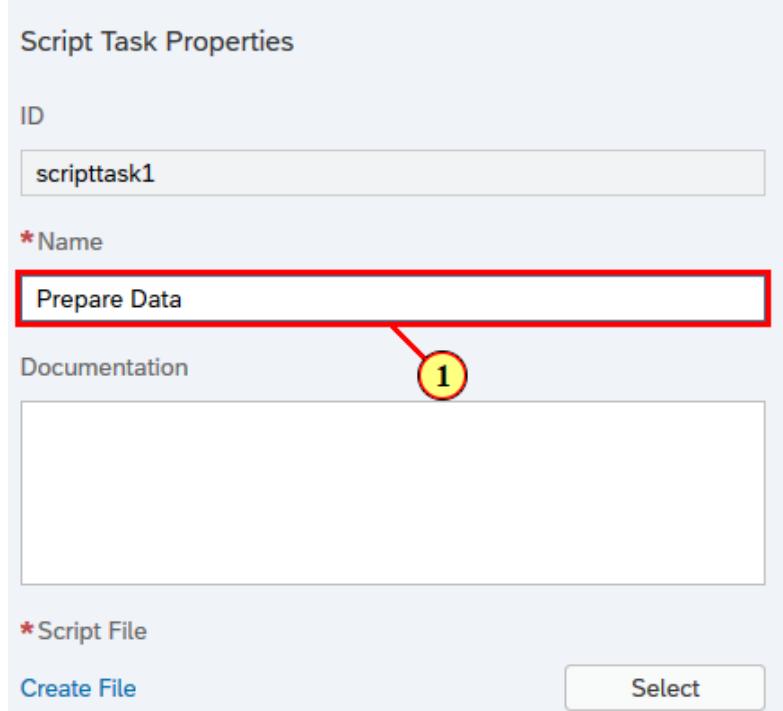
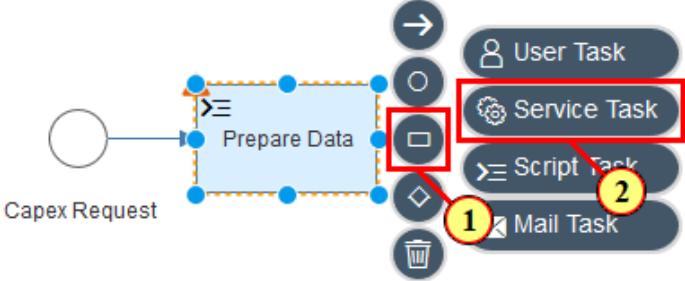
Explanation	Screenshot
<p>9. Wait until the virtual machine providing this dev space is in status RUNNING.</p> <p>10. Open the SAP Business Application Studio by clicking on the dev space name.</p>	 <p>The screenshot shows the SAP Dev Spaces interface. A dev space named "workshop" is listed, showing its status as "RUNNING". Other details include "Created On 05/29/2020 4:16 PM" and "ID ws-jwq2".</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>A message box in the SAP Business Application Studio interface states: "SAP Business Application Studio is ready to be used. Some extensions are still being installed in the background." The SAP logo is visible in the top left corner.</p>
<p>11. From the menu, select Terminal.</p> <p>12. Select New Terminal to open a Linux Terminal window.</p>	 <p>The SAP Business Application Studio menu bar is shown with the "Terminal" option highlighted. A red box and circle (1) highlight the "Terminal" menu item. A red box and circle (2) highlight the "New Terminal" option in the dropdown menu.</p>
<p>13. Enter cd projects/ in the Terminal 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 yo 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 Basic Multitarget Application by using the up/down arrow keys and press Enter.</p>	 <p>The terminal session shows the following steps:</p> <ul style="list-style-type: none"> cd projects/ yo We're constantly looking for ways to make yo better! May we anonymously report usage statistics to improve the tool over time? More info: https://github.com/yeoman/insight & http://yeoman.io Allo! What would you like to do? Run a generator @workflow/workflow Module Basic Multitarget Application Update your generators Install a generator Find some help (Move up and down to reveal more choices) <p>Arrows and numbers indicate specific interactions: (1) points to the "yo" command, (2) points to the "yeoman" feedback message, (3) points to the "Yes" option at the end of the feedback message, and (4) points to the "Basic Multitarget Application" choice in the list.</p>
<p>17. Enter CapexProject-00 or any name of your choice as Project name.</p>	 <p>The terminal session shows the creation of a new Yeoman project:</p> <ul style="list-style-type: none"> Make sure you are in the directory you want to scaffold into. This generator can also be run with: yo basic-multitarget-application ? Enter a project name CapexProject-00 The project will be generated in the /home/user/projects/CapexProject-00 directory. create CapexProject-00/.gitignore The project has been generated. <p>Arrows and numbers indicate specific interactions: (1) points to the "CapexProject-00" project name entry.</p>

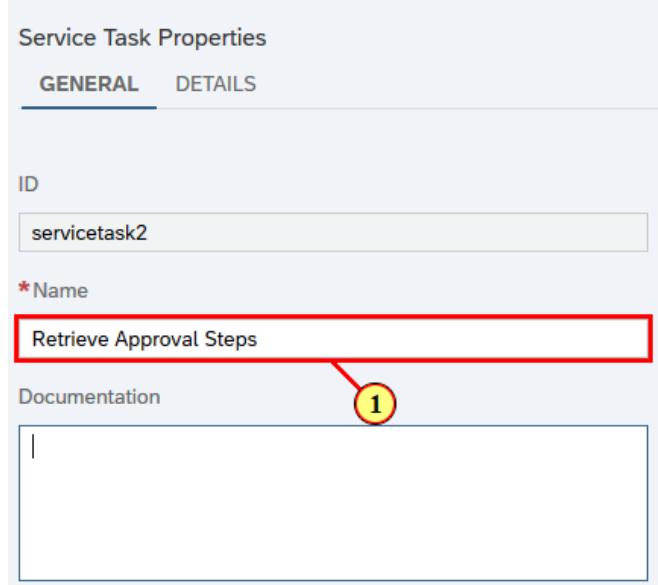
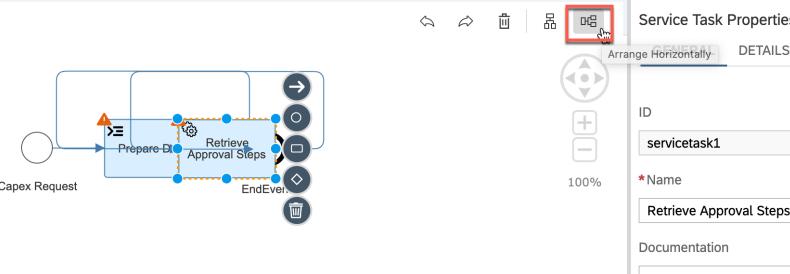
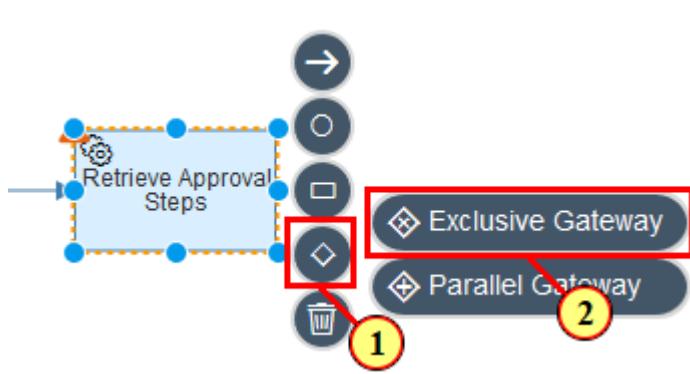
Explanation	Screenshot
<p>18. Change into the newly created directory by entering cd CapexProject-00. You can also use TAB for auto-completion, i.e. entering cd Cap<TAB> will be sufficient.</p> <p>19. Enter yo in the window to start Yeoman again.</p> <p>20. Select Workflow Module by using the up/down arrow keys and press Enter.</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 Capex-00 as Module Name.</p> <p>22. Enter Capex_00 as Workflow Name.</p> <p>23. Enter Capital Expenditure Approval as Workflow Description.</p> <p>24. Yeoman will warn you that there already is an mta.yaml file. Type Y to overwrite it.</p> <p>25. You can close the Terminal by typing exit 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 File.</p> <p>27. Select Open Workspace....</p>	 <p>SAP</p> <p>File Edit Selection View Go Debug Term</p> <p>New File... Alt+N New Folder</p> <p>Open... Ctrl+Alt+O Open Workspace... Ctrl+Alt+W</p> <p>Open Recent Workspace... Ctrl+Alt+R</p> <p>Save Workspace As... 2</p>

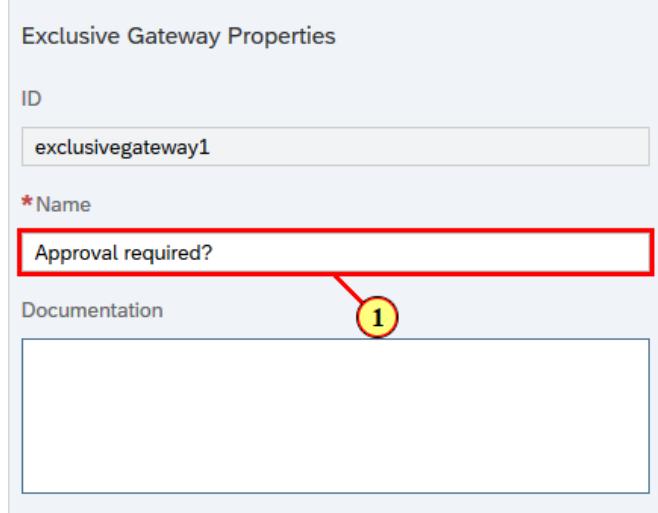
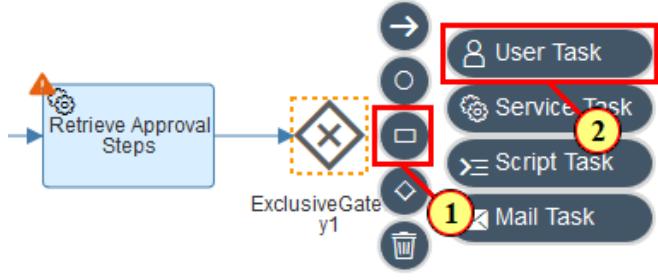
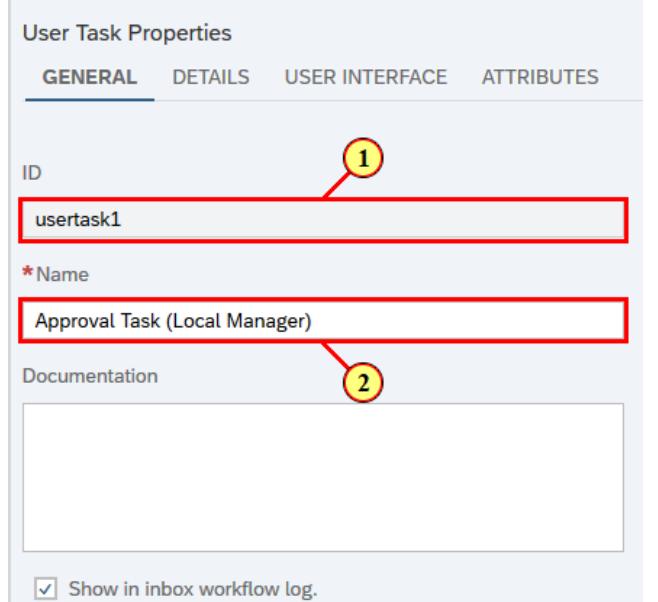
Explanation	Screenshot
<p>28. Select the directory containing your module CapexProject-00.</p> <p>29. Select Open.</p>	
<p>30. From the menu, select File. Note that there is an Auto Save functionality (enabled by default). If you choose to disable it, remember to manually save your files.</p>	
<p>31. If it is not already open, select the Explorer  from the sidebar.</p> <p>32. Open the Capex-00 module by selecting it.</p> <p>33. Open the workflows directory by selecting it.</p> <p>34. Open the CAPEX_00.workflow 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>	

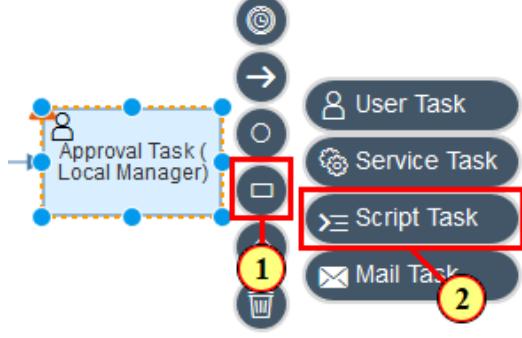
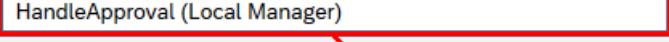
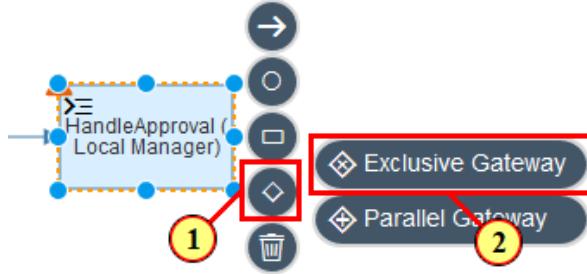
Explanation	Screenshot
<p>A new workflow with a default Start and End event is created.</p> <ul style="list-style-type: none"> ○ On the left side there is a toolbar for adding new flow elements. ○ On the right side is the properties pane for the currently selected element. <p>35. Click on the Start Event  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 Capex Request as the Name 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 Details of the start event.</p> <p>38. Select Configure Sample Context. 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 Create File.</p>	
<p>1. In the popup, enter CapexSampleStartPayload as File Name.</p> <p>2. Click Create.</p>	

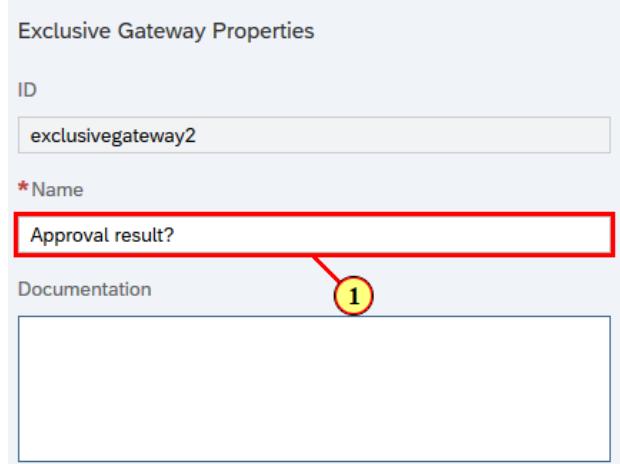
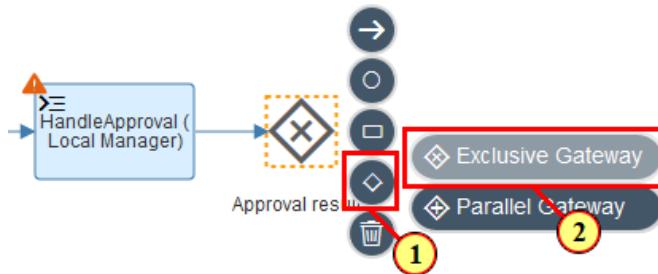
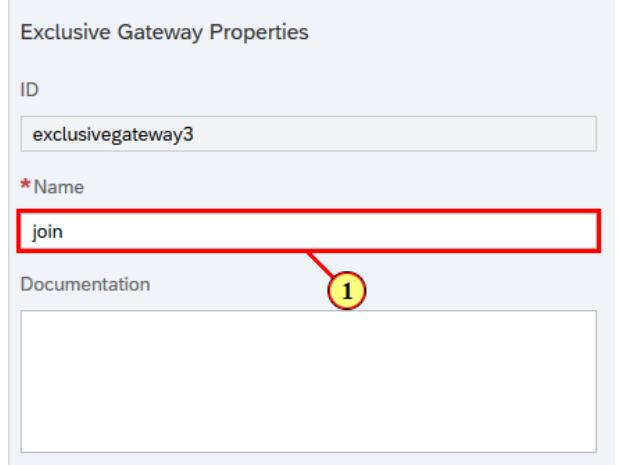
Explanation	Screenshot
<p>A new file with the name CapexSampleStartPayload.json is created in the project folder below sample-data. It is created with some default data as example.</p>	 <pre>* CAPEX_00.workflow x CapexSampleStartPayload.json x 1 <! 2 "product": "Hamlet (Paperback)", 3 "inStock": true, 4 "inventory": 20000, 5 "price": 7.49, 6 "publishingDate": "1600-04-23T18:25:43.511Z", 7 "author": { "name": "William Shakespeare" }, 8 "publishers": ["Simon & Brown", "SparkNotes", "Dover Publications"] 9 ></pre>
<ol style="list-style-type: none"> 3. Remove the example content and Copy-Paste the context from the CapexSampleStartPayload.json file from the <i>Exercise1 → Helper</i> folder in GitHub. 4. [Optional] You can choose to replace the RequestId of your choice. 5. Insert your name, email address and user id into the respective fields. UserId is your trial user email ID. 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. 	 <pre>* CAPEX_00.workflow x CapexSampleStartPayload.json x 1 <! 2 "RequestId": "CAPEX_REQ_00_001", 1 3 "title": "Build mobile apps", 4 "Requester": { 5 "Name": "<your name>", 6 "Email": "<your email address>", 7 "UserId": "<your userId>", 2 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 ></pre>
<ol style="list-style-type: none"> 7. Click the Start event  to select it. 8. Click on Add Task . 9. Select the Script Task . 	 <p>The screenshot shows the SAP Cloud Platform Workflow Management interface. A red box highlights the 'Start event' (Step 1). A red box highlights the 'Add Task' button (Step 2). A red box highlights the 'Script Task' option in the dropdown menu (Step 3).</p>

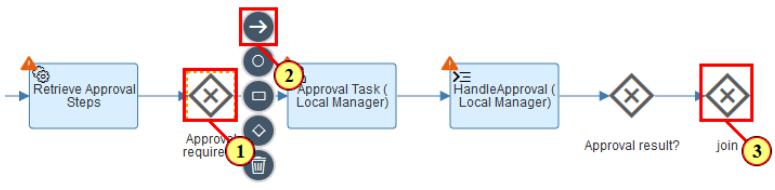
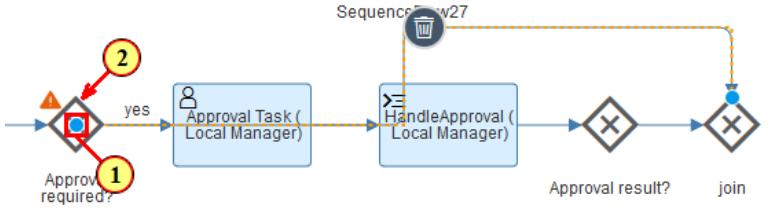
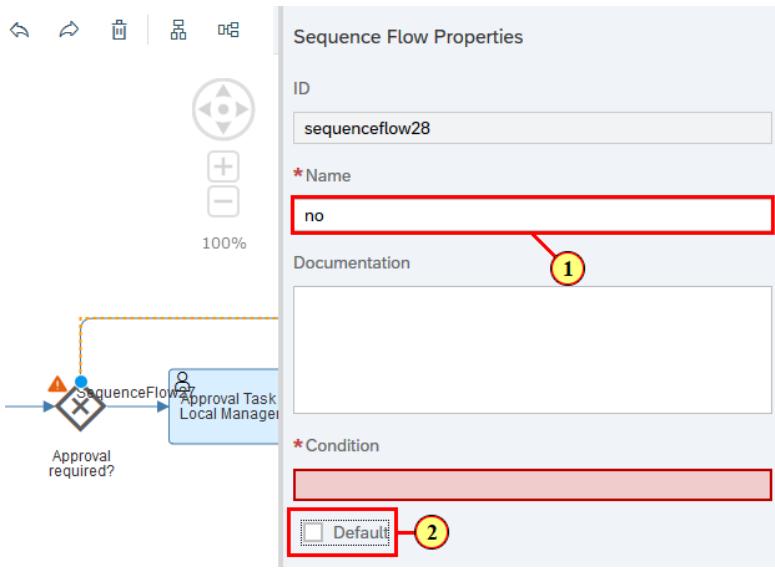
Explanation	Screenshot
<p>10. In the Script Task Properties, enter Prepare Data as Name.</p> <p>Note: We will configure the task later.</p>	 <p>The screenshot shows the 'Script Task Properties' dialog. The 'Name' field contains 'scripttask1'. Below it, the 'Documentation' section has a text area with 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 Add Task  in the speed buttons.</p> <p>12. Select Service Task .</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>

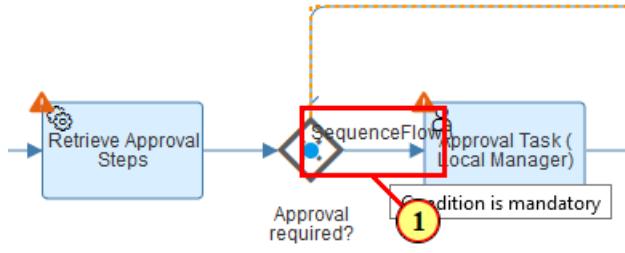
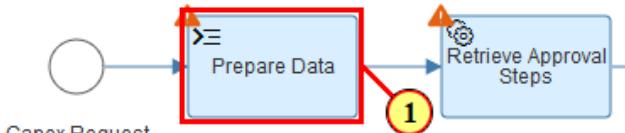
Explanation	Screenshot
<p>13. Enter Retrieve Approval Steps as Name.</p> <p>Note: We will configure the task later.</p>	 <p>Service Task Properties</p> <p>GENERAL DETAILS</p> <p>ID</p> <p>servicetask2</p> <p>*Name</p> <p>Retrieve Approval Steps</p> <p>Documentation</p>
<p>Note: If the workflow looks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

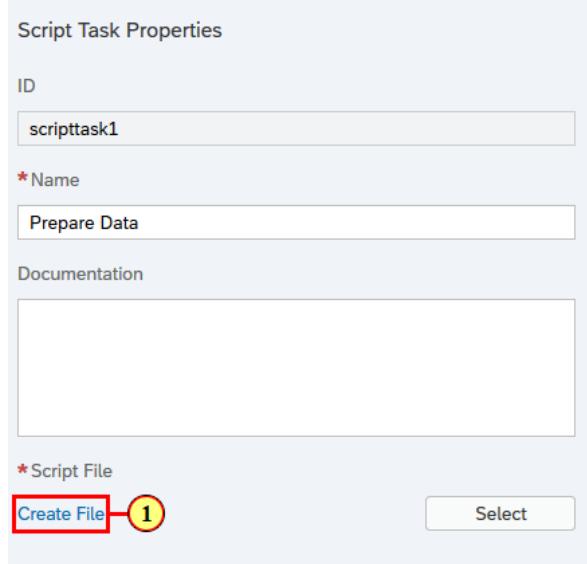
Explanation	Screenshot
<p>16. Enter Approval required? as gateway Name.</p> <p>Note: We will configure the gateway later, once we added more outgoing branches.</p>	
<p>17. Select Add Task .</p> <p>18. Select User Task .</p>	
<p>Note that the Task ID is usertask1. This will be needed later.</p> <p>19.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

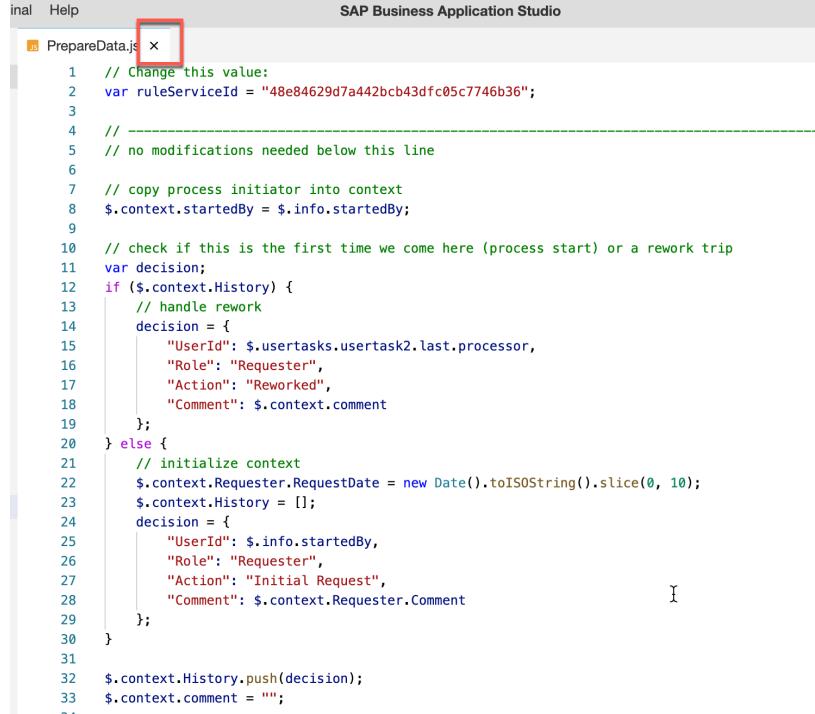
Explanation	Screenshot
<p>20. Select Add Task .</p> <p>21. Select Script Task .</p>	
<p>22. Enter Handle Approval (Local Manager) as Name.</p> <p>Note: We will configure the task later.</p>	<p>Script Task Properties</p> <p>ID: scripttask2</p> <p>* Name: HandleApproval (Local Manager)  </p> <p>Documentation: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

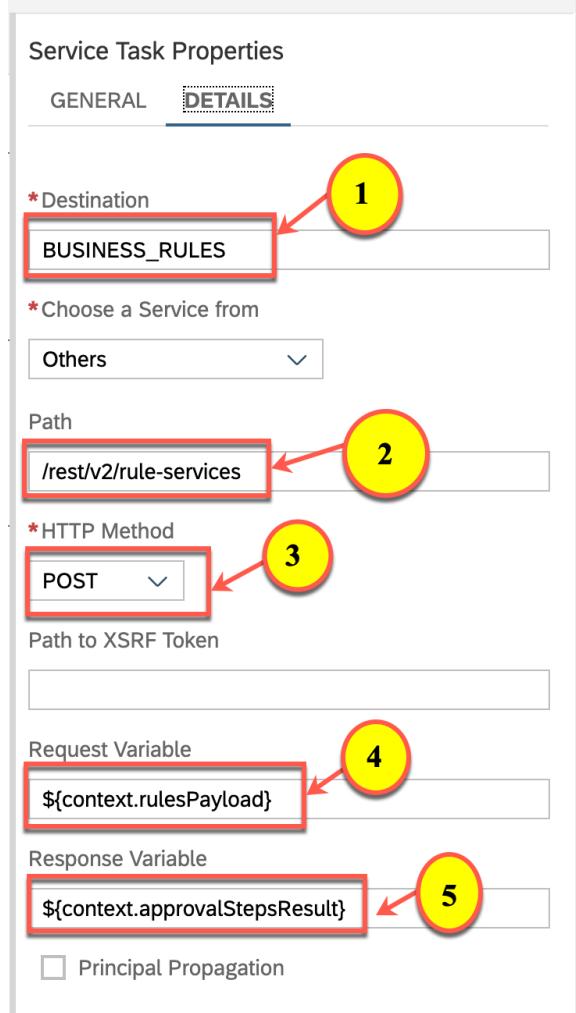
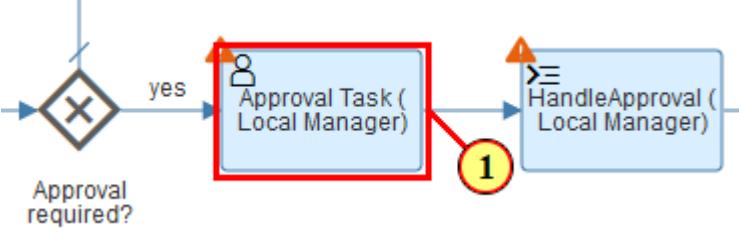
Explanation	Screenshot
25. Enter Approval result? as Name.	
26. Select Add Gateway  .	
27. Select Exclusive Gateway  .	

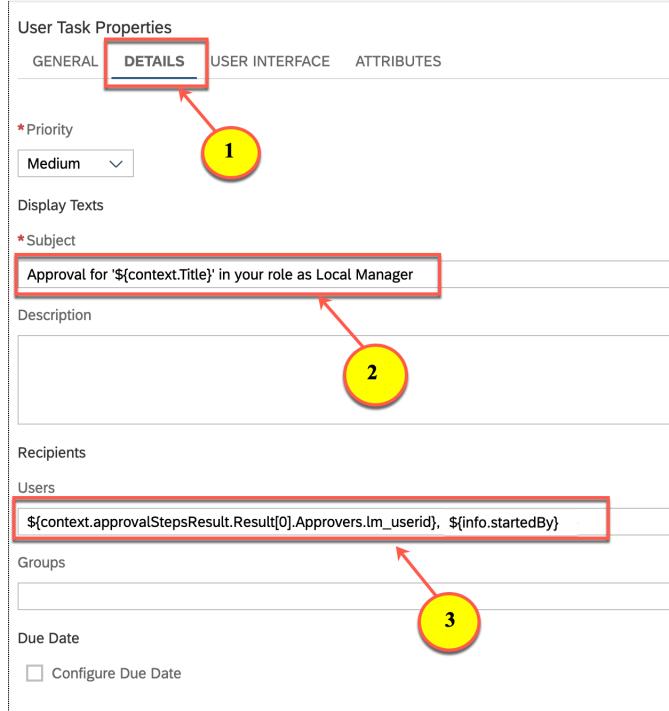
Explanation	Screenshot
<p>29. Select the Exclusive Gateway  with the name 'Approval required?'</p> <p>30. Click on the Sequence Flow Connector  and start dragging it</p> <p>31. Drop the Sequence Flow Connector onto the top corner of the join gateway.</p> <p>Note: Sequence Flow Connectors have Connection 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' (script), an 'Exclusive Gateway' labeled 'Approval required?' (with a red box around it), 'Approval Task (Local Manager)', 'HandleApproval (Local Manager)', another 'Exclusive Gateway' (with a red box around it), and a 'join gateway' (with a red box around it). A sequence flow connector is being dragged from the top corner of the first gateway to the top corner of the join gateway. Three numbered circles (1, 2, 3) indicate specific points of interest: 1 is on the connector near the gateway, 2 is on the connector near the join gateway, and 3 is on the join gateway itself.</p>
<p>32. Drag the Connection Dot  from the middle of the Exclusive Gateway with name Approval required? to its upper corner.</p> <p>33. Drop on the upper corner of the icon.</p>	 <p>The screenshot shows the workflow diagram after step 32. The connection dot has been moved to the upper corner of the 'Approval required?' gateway. A sequence flow labeled 'SequenceFlow27' connects the gateway to the 'HandleApproval' task. The 'join' gateway is also visible at the end of the flow.</p>
<p>Since there is now more than one Sequence Flow, each flow has a name and branch condition.</p> <p>34. Enter no as Name for the newly created (upper) Sequence Flow.</p> <p>35. Select Default 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 is highlighted with a red box and contains the value 'no'. The 'Condition' field is also highlighted with a red box and contains the value 'Default'. A numbered circle (1) is on the 'Name' field, and another (2) is on the 'Default' condition.</p>

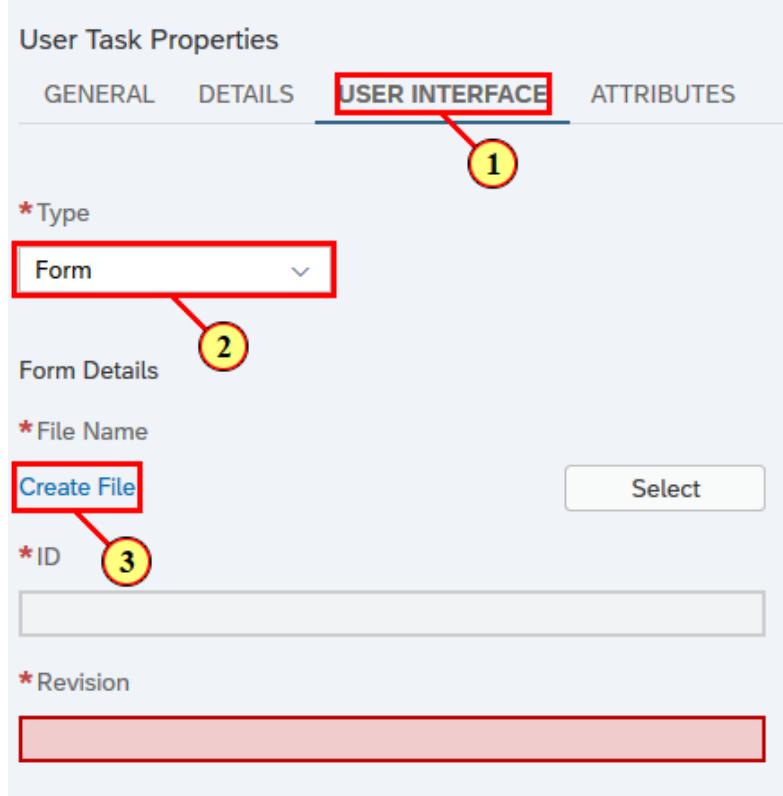
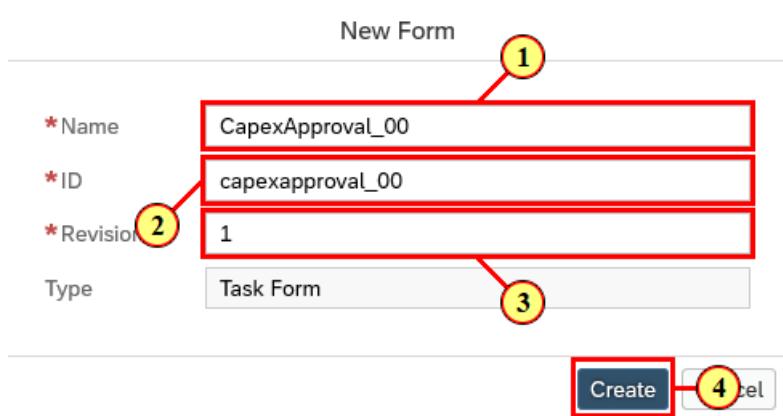
Explanation	Screenshot
<p>36. Select the Sequence Flow between the gateway 'Approval required?' and the user task 'Approval Task (Local Manager)'</p>	
<p>37. Enter yes as Name of the sequence flow.</p> <p>38. Enter <code>#{context.approvalStepsResult.Result[0].Approvers.lm_required}</code> as Condition.</p> <p>⚠️ 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 numbered 1)</p> <p>*Condition: <code>text.approvalStepsResult.Result[0].Approvers.lm_required</code> (highlighted with a red box)</p> <p>[Default] (highlighted with a yellow circle numbered 2)</p>
<p>39. Select the script task Prepare Data.</p>	

Explanation	Screenshot
<p>40. In the properties, click Create File.</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 PrepareData as Name. 42. Click on Create.</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 PrepareData.js 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 Retrieve Approval Steps.</p>	 <pre data-bbox="747 1110 1390 1262"> graph LR Start(()) --> >= Task1[Prepare Data] Task1 --> Task2[Retrieve Approval Steps] Task2 --> >= End(()) </pre>

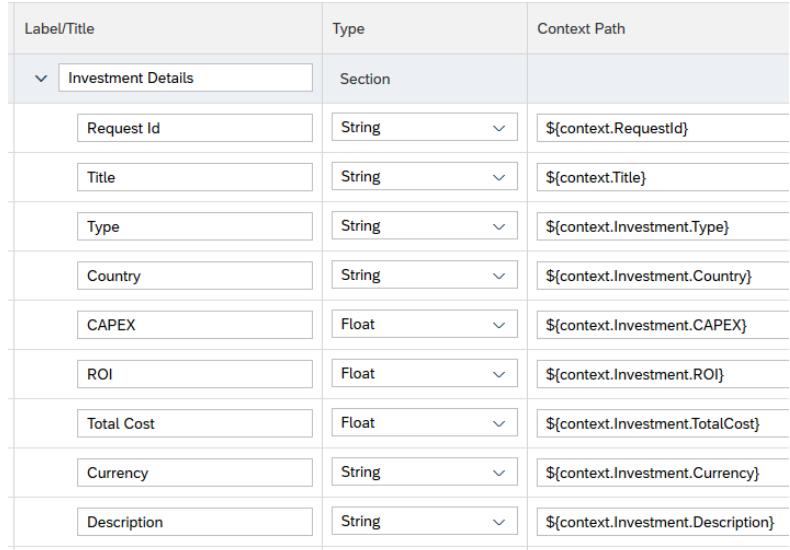
Explanation	Screenshot
<p>46. Select the DETAILS tab.</p> <p>47. Enter BUSINESS_RULES as Destination.</p> <p>48. Enter /rest/v2/rule-services as Path.</p> <p>49. Enter HTTP Method as POST.</p> <p>50. Enter <code> \${context.rulesPayload}</code> as Request Variable.</p> <p>51. Enter <code> \${context.approvalStepsResult}</code> as Response Variable.</p> <p>⚠️ Be careful during copying. There are no spaces in the path, request and response variable 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 Approval Task (Local Manager).</p>	 <pre> graph LR A{Approval required?} -- yes --> B[Approval Task (Local Manager)] B --> C[HandleApproval (Local Manager)] </pre>

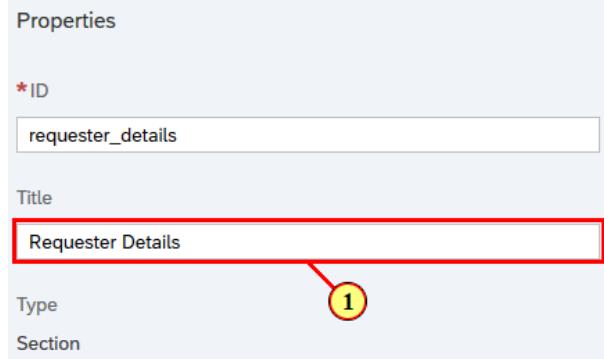
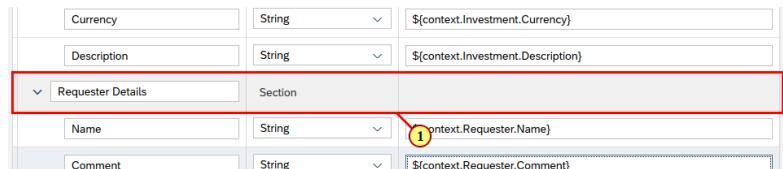
Explanation	Screenshot
<p>53. Select the DETAILS tab.</p> <p>54. Enter Approval for '<code>#{context.Title}</code>' in your role as Local Manager as Subject.</p> <p>55. Enter <code>#{context.approvalStepsResult.Result[0].Approvers.lm_userid}, \${info.startedBy}</code> as Recipients - Users.</p> <p>It means that this user task will be sent to the users retrieved from Business Rules and the workflow initiator.</p> <p>⚠️ Be careful during copying. There are no spaces Users 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>

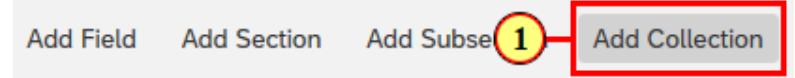
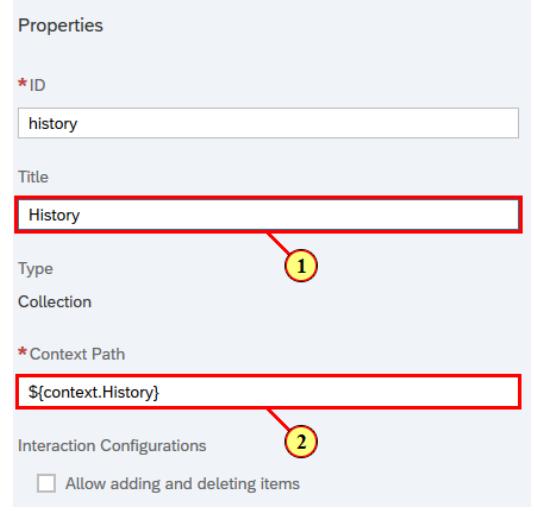
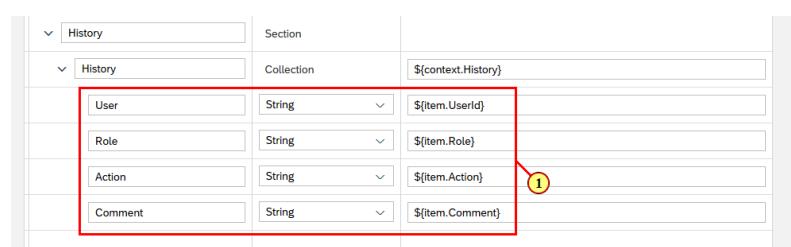
Explanation	Screenshot
<p>56. Select the USER INTERFACE tab.</p> <p>57. Select Form in the dropdown menu for Type.</p> <p>58. Click Create File.</p> <p>Note: Forms are a quick way to create simple task UIs.</p>	 <p>The screenshot shows the 'User Task Properties' dialog with the 'USER INTERFACE' tab selected (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 CapexApproval_00 as Name.</p> <p>60. Press TAB to accept the generated Id.</p> <p>61. Enter 1 as Revision.</p> <p>62. Click Create.</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 Add Section.</p>	<p>The screenshot shows the same task form 'CapexApproval_00'. The 'Add Section' button in the toolbar has been 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 Investment Details 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 Add Field.</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>

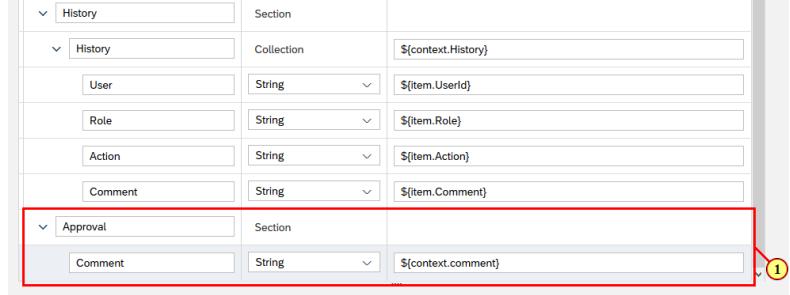
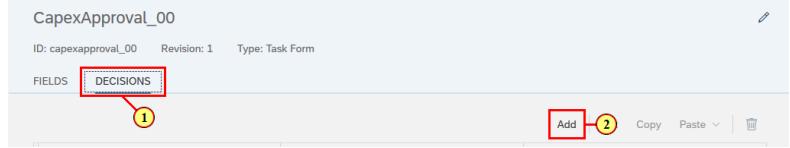
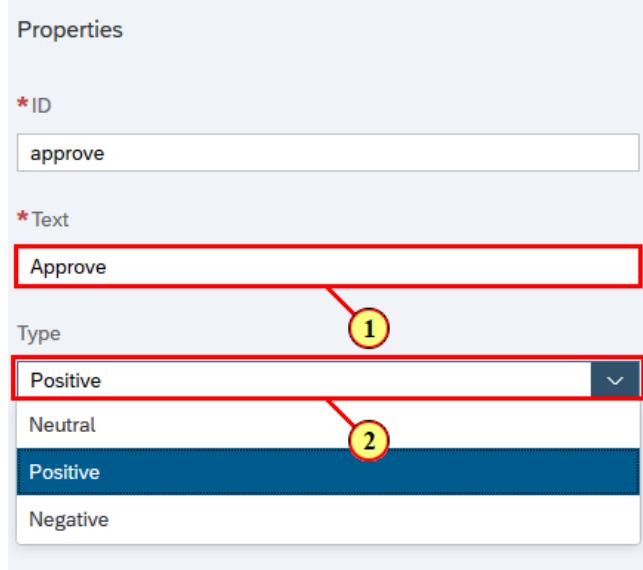
Explanation	Screenshot
<p>66. Enter Request Id as Label. The ID is then automatically derived from this.</p> <p>67. Enter <code> \${context.RequestId}</code> as Context Path.</p> <p>68. An input field can also be marked as Required. Since the Request Id is already given and should not be editable, leave 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 Type field opens a dropdown list. Select the type as String.</p>	<p>Type</p> <ul style="list-style-type: none"> String String Boolean Date Time DateTime Integer Float
<p>70. Clicking on the Mode field opens a dropdown list. Select Display-Only.</p>	<p>Mode</p> <ul style="list-style-type: none"> Display-Only Editable Display-Only

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"> ○ Set the property of all of the fields as Editable and Required. <p>Label: Title Type: String Context Path: \${context.Title}</p> <p>Label: Type Type: String Context Path: \${context.Investment.Type}</p> <p>Label: Country Type: String Context Path: \${context.Investment.Country}</p> <p>Label: CAPEX Type: Float Context Path: \${context.Investment.CAPEX}</p> <p>Label: ROI Type: Float Context Path: \${context.Investment.ROI}</p> <p>Label: Total Cost Type: Float Context Path: \${context.Investment.TotalCost}</p> <p>Label: Currency Type: String Context path: \${context.Investment.Currency}</p> <p>Label: Description Type: String Context Path: \${context.Investment.Description}</p> <p>⚠️ Be careful during copying. There are no spaces in ContextPath 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 1529 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 1529 931" style="background-color: #e0e0e0;"></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 1529 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 1529 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 1529 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 1529 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 1529 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 1529 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 1529 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 1529 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 1529 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}																																

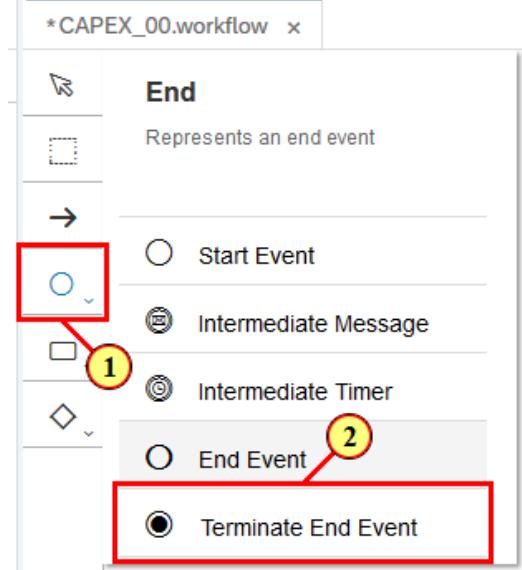
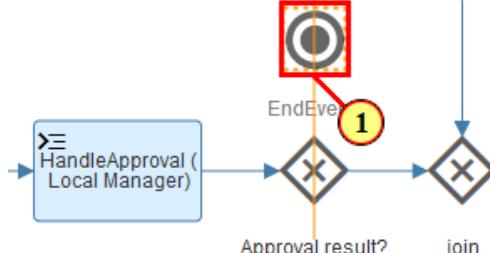
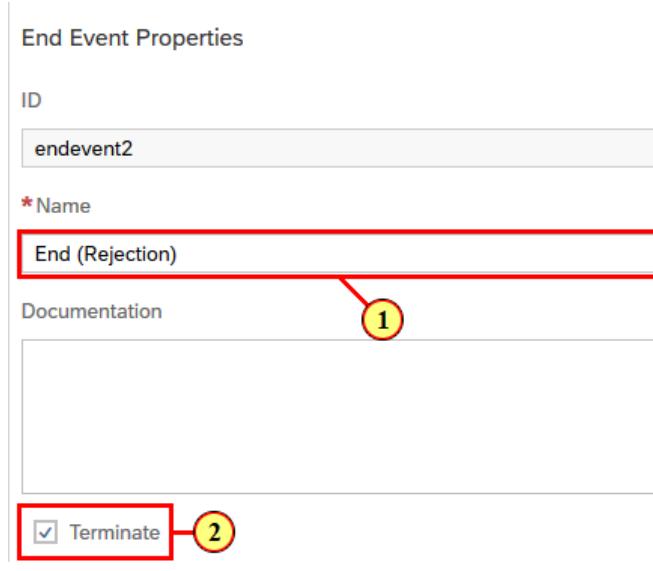
Explanation	Screenshot
<p>72. Select Investment Details section.</p> <p>73. Select Add Section.</p>	
<p>74. Enter Requester Details as Title.</p>	
<p>75. Add the following fields.</p> <p>Label: Name Type: String Mode: Display-Only Control: Input Context Path: \${context.Requester.Name}</p> <p>Label: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

Explanation	Screenshot
<p>78. Select Add Collection.</p> <p>Note: Collections allow for an array of items to be displayed as table.</p>	
<p>79. Enter History as Title.</p> <p>80. Enter <code>#{context.History}</code> as Context Path.</p>	
<p>81. Add the following values for Label and Context Path and select Type String and mark them as Display-Only.</p> <p>Label: User Context Path: <code>#{item.UserId}</code></p> <p>Label: Role Context Path: <code>#{item.Role}</code></p> <p>Label: Action Context Path: <code>#{item.Action}</code></p> <p>Label: Comment Context Path: <code>#{item.Comment}</code></p> <p>Note that inside a collection, the variable item refers to a single row.</p> <p>⚠️ Be careful during copying. There are no spaces in ContextPath 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>	

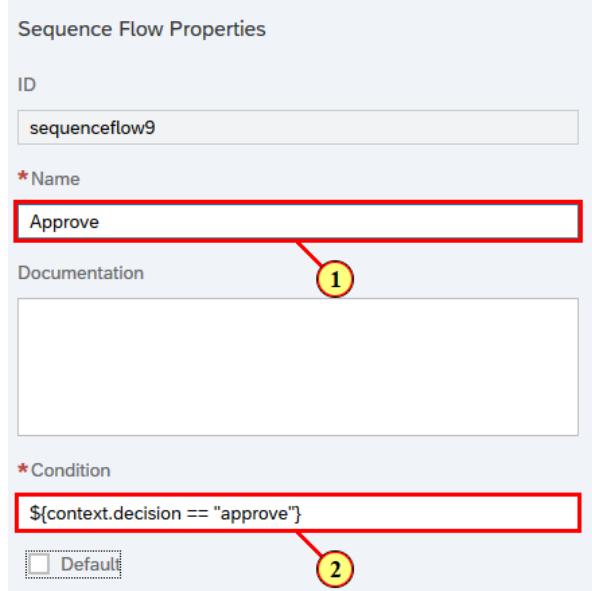
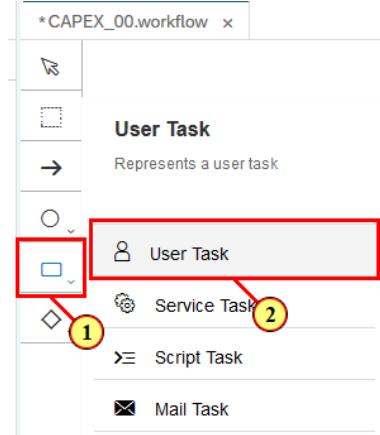
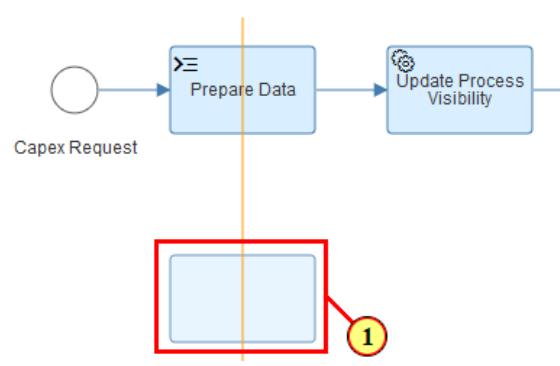
Explanation	Screenshot
<p>82. Add a new Section with title Approval.</p> <p>83. Add a new field and name it Comment.</p> <p>84. Enter <code>#{context.comment}</code> as Context Path</p> <p>85. Change the UI Control type from Input field to Text Area.</p>	<p>The screenshot shows the 'Properties' dialog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

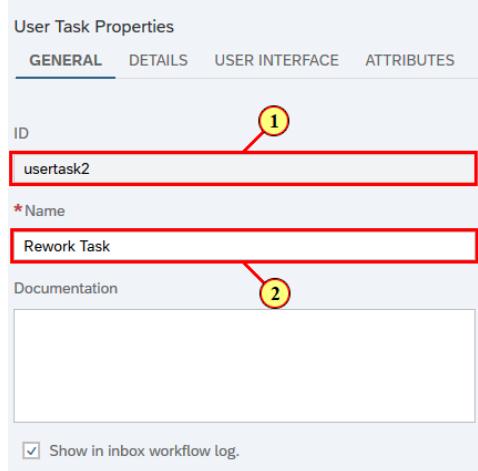
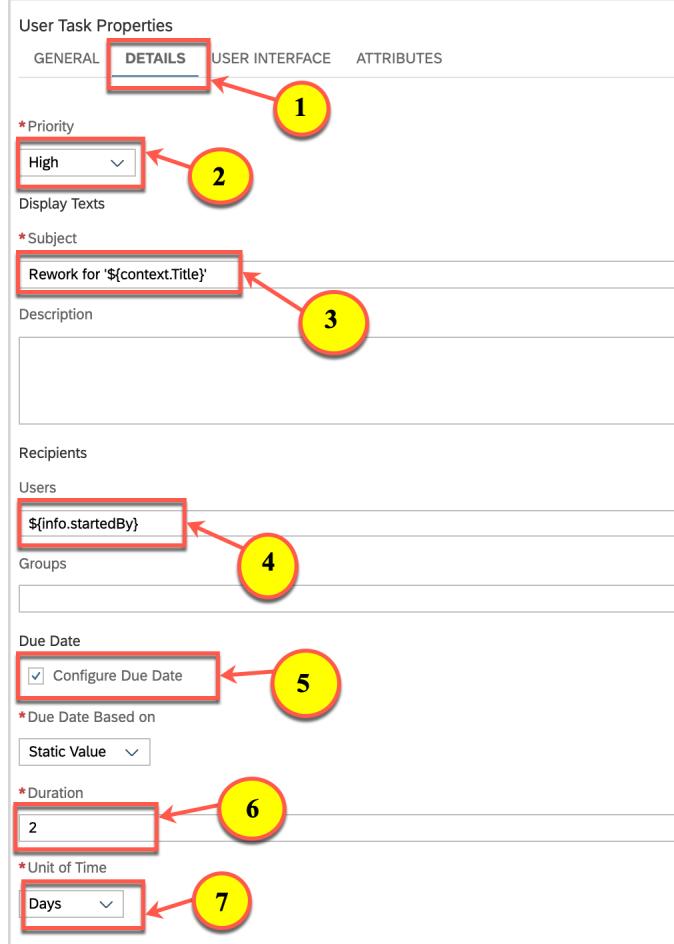
Explanation	Screenshot
86. Ensure your Form looks like this.	
87. Select the Decisions tab. 88. Select Add to add a new decision option.	
89. Enter Approve as Text of the decision button. 90. Select Positive as Type from the dropdown-list.	
91. In the same way, add a decision option Reject with type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

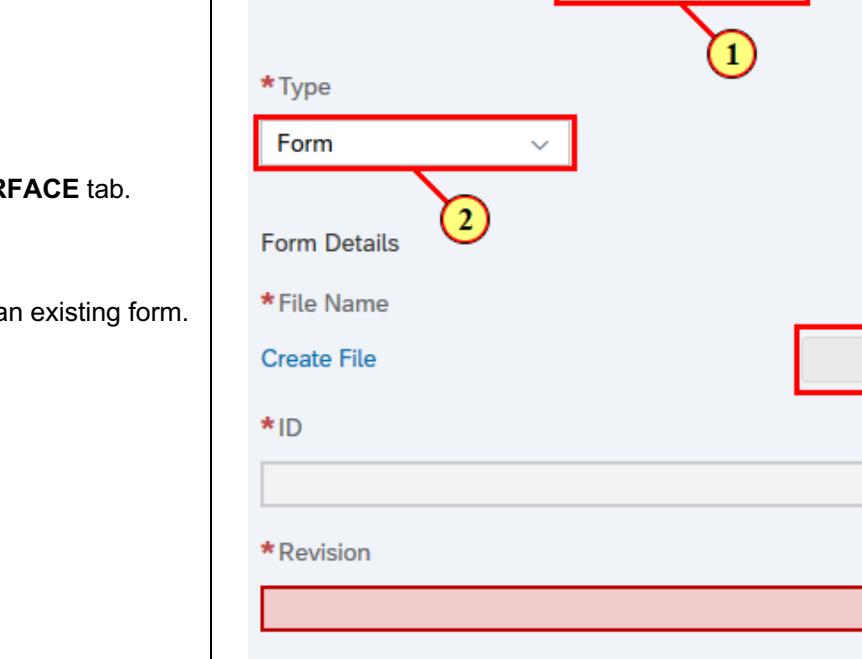
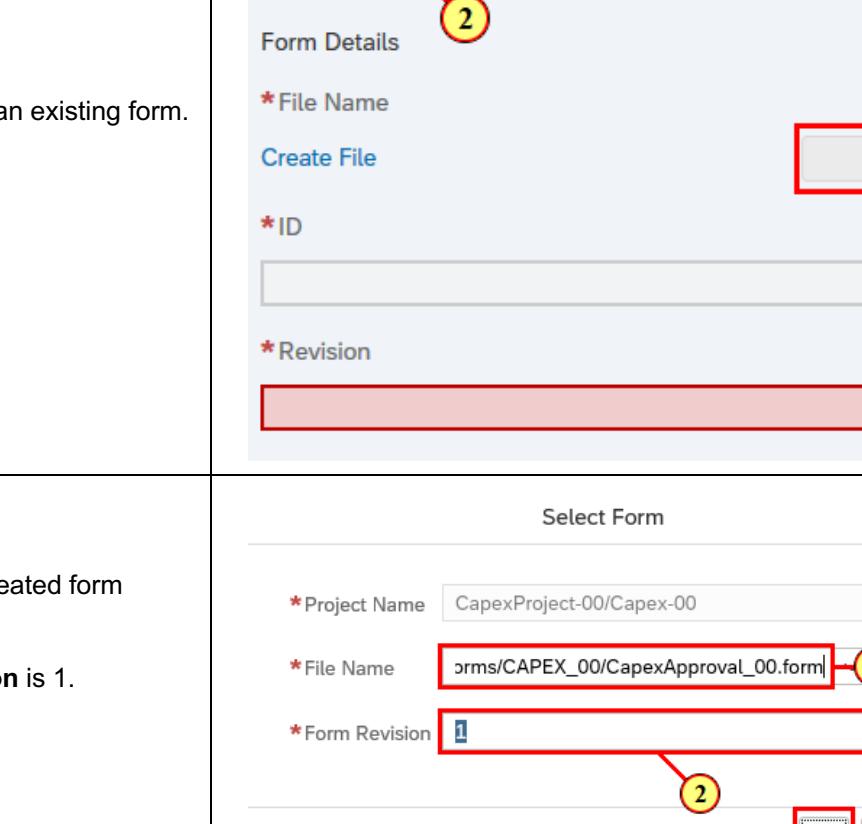
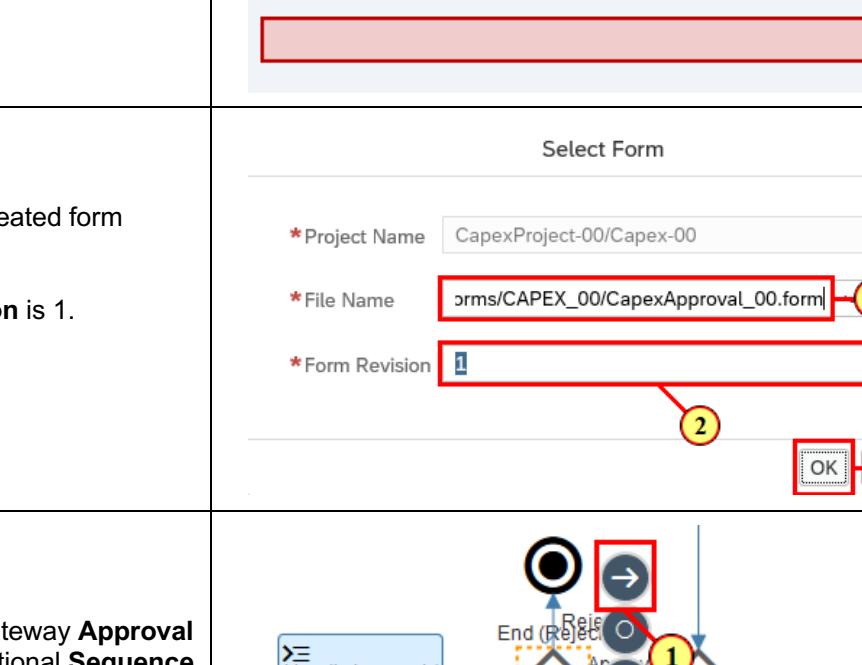
Explanation	Screenshot
<p>93. Select the script task Handle Approval (Local Manager).</p>	<pre> graph LR A[Approval Task (Local Manager)] --> B[HandleApproval (Local Manager)] B --> C{Approval result?} B -- X --> C </pre>
<p>94. In the Script Task Properties, select Create File.</p>	<p>* Script File Create File 1</p>
<p>95. Enter HandleApprovalLocalManager as Name of the new file.</p> <p>96. Select Create.</p>	<p>Create New File</p> <p>* Name HandleApprovalLocalManager 1</p> <p>Create 2 Cancel</p>
<p>97. Replace the default content with the HandleApprovalLocalManager.js 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 Add Event.</p> <p>100. Select the Terminate End Event .</p>	
<p>101. Place the End Event above the exclusive gateway named Approval result?</p> <p>Note: The yellow guiding lines help you position it nicely.</p>	
<p>102. Enter End (Rejection) as Name.</p> <p>103. Ensure that Terminate is checked.</p>	

Explanation	Screenshot
<p>104. Select the exclusive gateway with name Approval result?</p> <p>105. Drag the sequence flow connector onto the end event with name End (Rejection)</p>	
<p>106. Enter Reject as Name.</p> <p>107. Select the checkbox for Default condition.</p>	<p>Sequence Flow Properties</p> <p>ID: sequenceflow11</p> <p>* Name: Reject (highlighted with a red box)</p> <p>Documentation: (empty)</p> <p><input checked="" type="checkbox"/> Default (highlighted with a red box)</p>
<p>108. Select the sequence flow connector between the gateways named Approval result? and join.</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 Approve as Name.</p> <p>110. Enter <code>#{context.decision == "approve"}</code> as Condition.</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 Add Task .</p> <p>112. Select User Task .</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 User Task below the script task Prepare Data.</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' has a vertical orange guiding line extending downwards. A blue script task labeled 'Prepare Data' is positioned to the right of the line. To its right is another blue script task labeled 'Update Process Visibility'. A red box highlights a user task placed below the 'Prepare Data' task, with a yellow circle containing the number 1 indicating its position relative to the guiding line.</p>

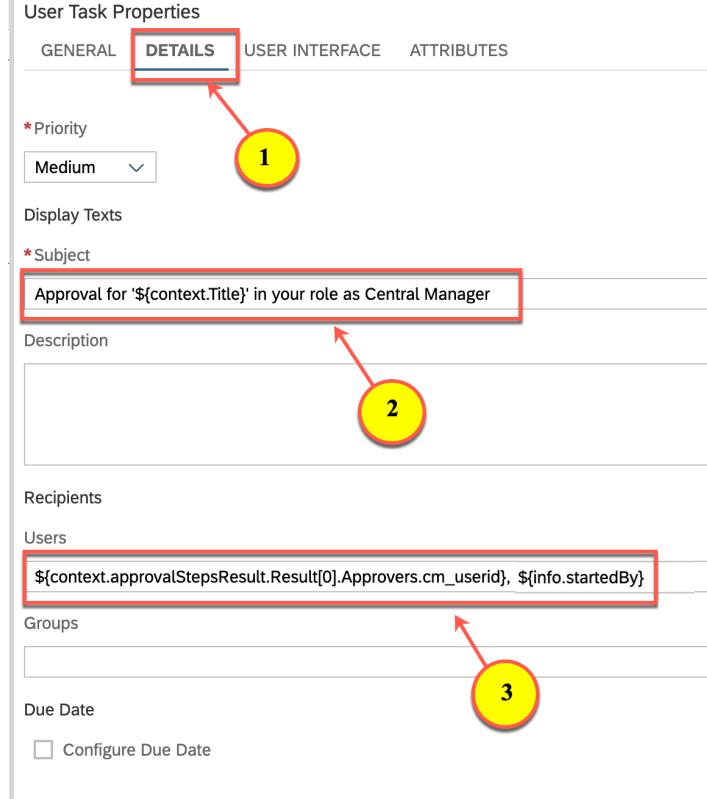
Explanation	Screenshot
<p>114. Verify that the ID is <code>usertask2</code>. If it differs, the references in the script file Prepare Data.js needs to be adapted accordingly.</p> <p>115. Enter Rework Task as Name.</p>	 <p>User Task Properties</p> <p>GENERAL DETAILS USER INTERFACE ATTRIBUTES</p> <p>ID usertask2</p> <p>*Name Rework Task</p> <p>Documentation</p> <p><input checked="" type="checkbox"/> Show in inbox workflow log.</p>
<p>116. Select the DETAILS tab.</p> <p>117. Select the Priority as High.</p> <p>118. Enter Rework for '\${context.Title}' as Subject.</p> <p>119. Enter \${info.startedBy} as Recipients - User.</p> <p>120. Select Configure Due Date.</p> <p>121. Enter 2 as Duration and Days from the dropdown list Unit of Time, to mark the task as being due 2 days after it started.</p> <p>⚠️ 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</p> <p>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>

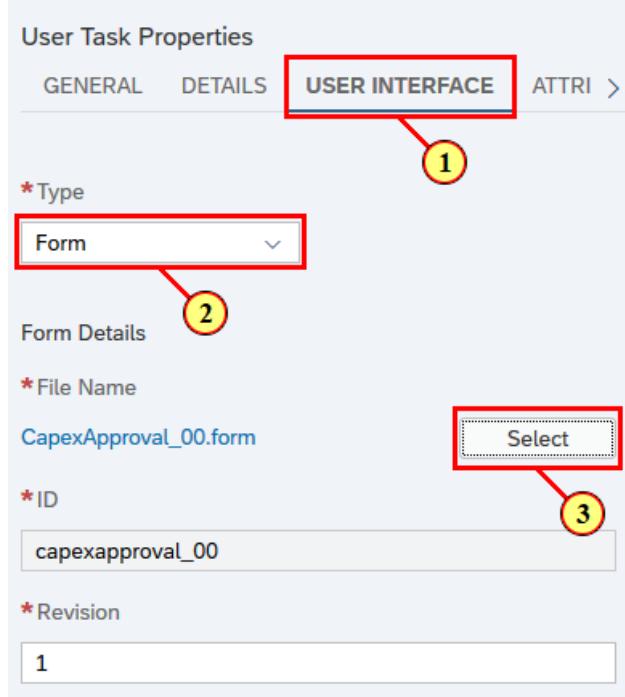
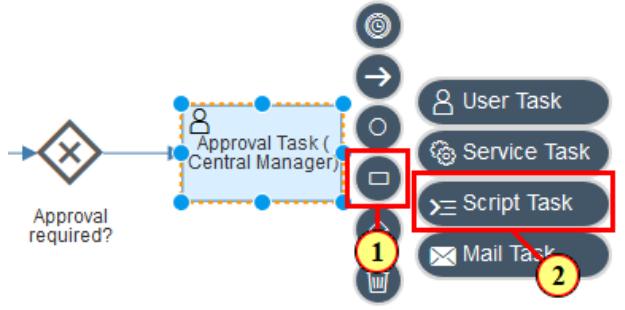
Explanation	Screenshot
<p>122. Select the USER INTERFACE tab.</p> <p>123. Select Form as Type.</p> <p>124. Click Select to choose an existing form.</p>	 <p>The screenshot shows the 'User Task Properties' dialog with the 'USER INTERFACE' tab selected (indicated by a red box and yellow circle 1). The 'Type' dropdown is set to 'Form' (indicated by a red box and yellow circle 2). A 'Select' button is visible on the right (indicated by a red box and yellow circle 3).</p>
<p>125. Select the previously created form CapexApproval_00.</p> <p>126. Ensure that the Revision is 1.</p> <p>127. Select OK.</p>	 <p>The screenshot shows the 'Select Form' dialog. The 'File Name' field contains 'orms/CAPEX_00/CapexApproval_00.form' (indicated by a red box and yellow circle 1). The 'Form Revision' field contains '1' (indicated by a red box and yellow circle 2). The 'OK' button is highlighted with a red box and yellow circle 3.</p>
<p>128. Select the Exclusive Gateway Approval result? and drag an additional Sequence Flow Connector...</p>	 <p>The screenshot shows a BPMN diagram fragment. A blue rounded rectangle labeled 'HandleApproval (Local Manager)' has a sequence flow pointing to an exclusive gateway (a diamond with outgoing flows). One of the gateway's outgoing flows is highlighted with a red box and yellow circle 1, representing the 'Approval result?' connector. Other symbols include an end event, a join, and various decision and activity nodes.</p>

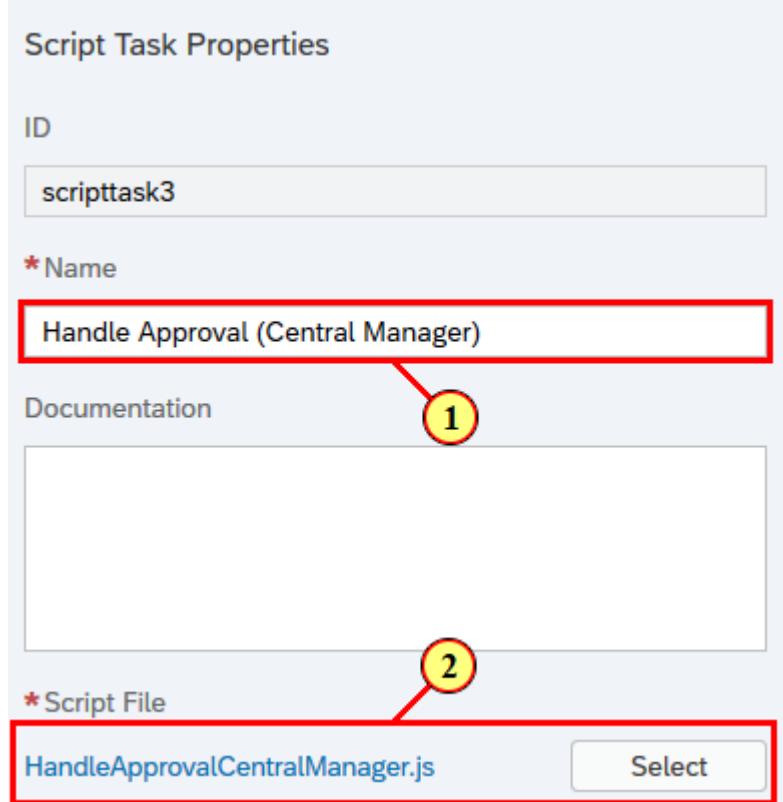
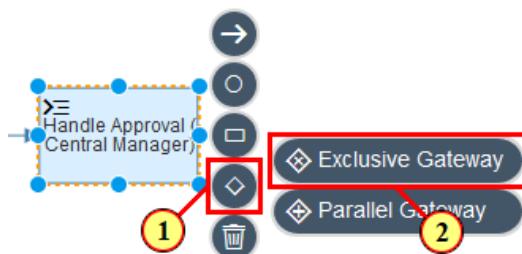
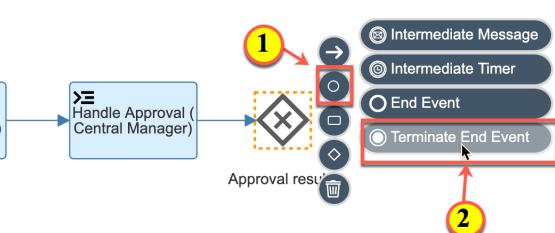
Explanation	Screenshot
<p>...onto the User Task Rework Task. Position it onto the highlighted middle dot.</p>	<pre> graph LR Start(()) -- Capex Request --> Prep[Prepare Data] Prep --> Ret[Retrieve Approval Steps] Ret --> Decision{Approval required?} Decision -- yes --> ApprovalTask[Approval Task (Local Manager)] ApprovalTask --> Handle1[Handle Approval (Local Manager)] Decision -- no --> Join(()) Handle1 --> EndReject((End (Reject))) EndReject -- Reject --> Decision EndReject -- Approve --> ApprovalResult{Approval result?} ApprovalResult -- join --> Join </pre>
<p>129. Click on the Connection Dot 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 Rework 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 Rework Task</p> <p>134. Select Sequence Flow Connector 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 Prepare Data.</p>	<p>Capex Request</p> <p>Sequence flow connected from Rework Task to Prepare Data task.</p>

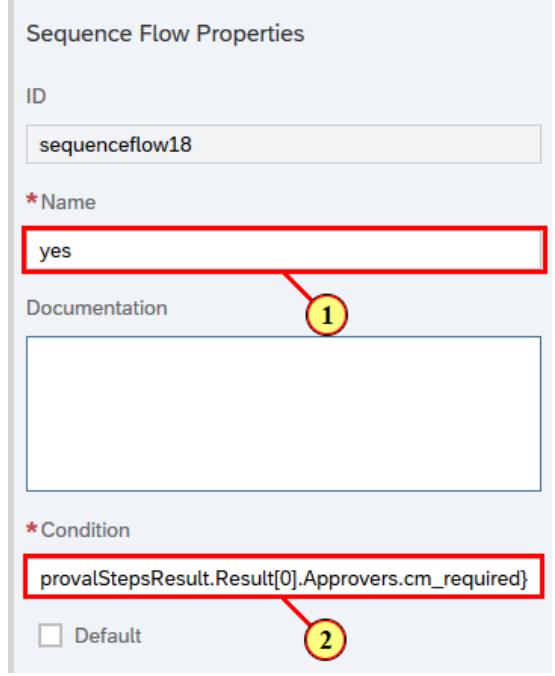
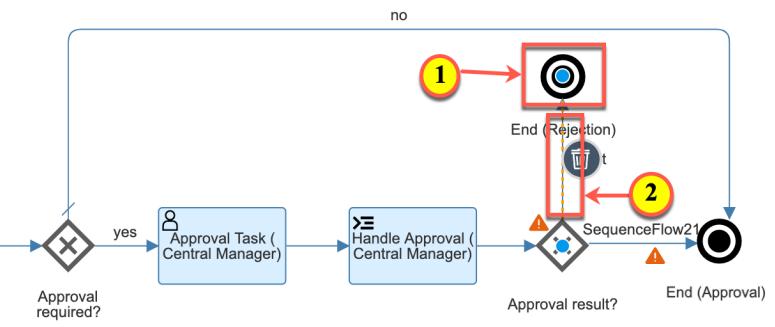
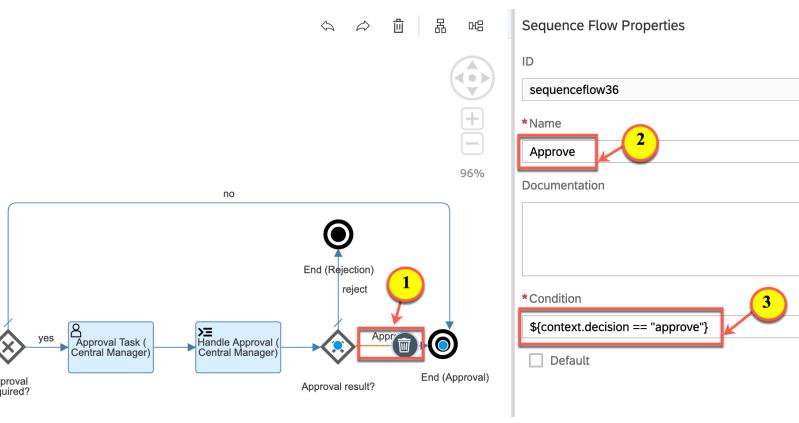
Explanation	Screenshot
<p>Ensure your workflow looks like this.</p>	<pre> graph LR Start(()) --> PrepareData[Prepare Data] PrepareData --> RetrieveSteps[Retrieve Approval Steps] RetrieveSteps --> Decision{Approval required?} Decision -- yes --> ApprovalTask[Approval Task (Local Manager)] ApprovalTask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

Explanation	Screenshot
<p>142. Select the Details tab.</p> <p>143. Enter Approval for '\${context.Title}' in your role as Central Manager as Subject.</p> <p>144. Enter <code> \${context.approvalStepsResult.Result[0].Approvers.cm_userid}, \${info.startedBy}</code> as Recipients - Users.</p> <p>⚠️ 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>

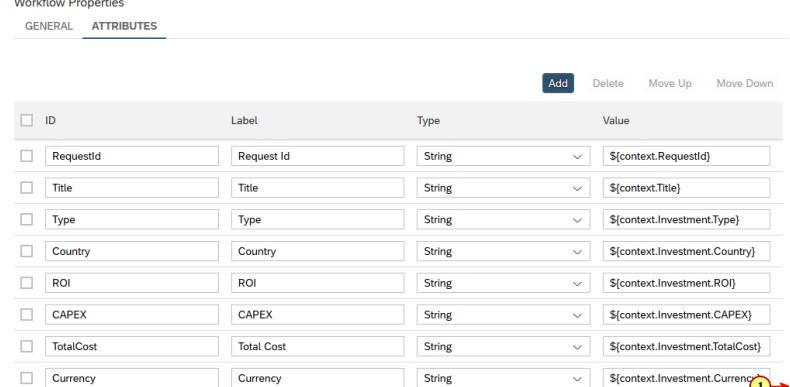
Explanation	Screenshot
<p>145. Select the User Interface tab.</p> <p>146. Select Form as Type.</p> <p>147. Use the Select button to choose the previously created form capexapproval_00.</p>	 <p>User Task Properties</p> <p>GENERAL DETAILS USER INTERFACE ATTRI ></p> <p>* Type: Form</p> <p>Form Details</p> <p>* File Name: CapexApproval_00.form Select</p> <p>* ID: capexapproval_00</p> <p>* Revision: 1</p>
<p>148. Select Add Task on the central manager approval.</p> <p>149. Select the Script Task.</p>	 <p>Approval required?</p> <p>Approval Task (Central Manager)</p> <p>1 Script Task</p> <p>2 User Task</p> <p>3 Service Task</p> <p>4 Mail Task</p>

Explanation	Screenshot
<p>150. Enter Handle Approval (Central Manager) as Name.</p> <p>151. Click Create File to create a new file and Name it HandleApprovalCentralManager.</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 Task ID (usertask3) multiple times. This is the id of the User Task Approval Task (Central Manager) 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 Handle Approval 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 Add Gateway .</p> <p>154. Select the Exclusive Gateway .</p> <p>155. Name it as Approval result?</p>	 <p>The screenshot shows a workflow diagram. An 'Exclusive Gateway' is highlighted with a red box. A yellow circle with the number '1' points to the gateway. A yellow circle with the number '2' points to a 'Parallel Gateway' icon.</p>
<p>156. Click Approval result? and Select Add Event.</p> <p>157. Select the Terminate End Event. Name the event as End (Approval).</p>	 <p>The screenshot shows a workflow diagram. A 'Terminate End Event' is highlighted with a red box. A yellow circle with the number '1' points to the event. A yellow circle with the number '2' points to the 'Approval result?' gateway.</p>

Explanation	Screenshot
<p>158. On the previous exclusive gateway with name Approval required? add a new Sequence Flow Connector and drag it onto the newly added Terminate End Event.</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 no as Name.</p> <p>160. Select it as Default.</p>	<p>Sequence Flow Properties</p> <p>ID: sequenceflow23</p> <p>* Name: no (circled with yellow circle 1)</p> <p>Documentation: (empty field)</p> <p>Default: checked (circled with yellow circle 2)</p>
<p>161. Select the Sequence Flow Connector between Approval required? and Approval Task (Central Manager).</p>	

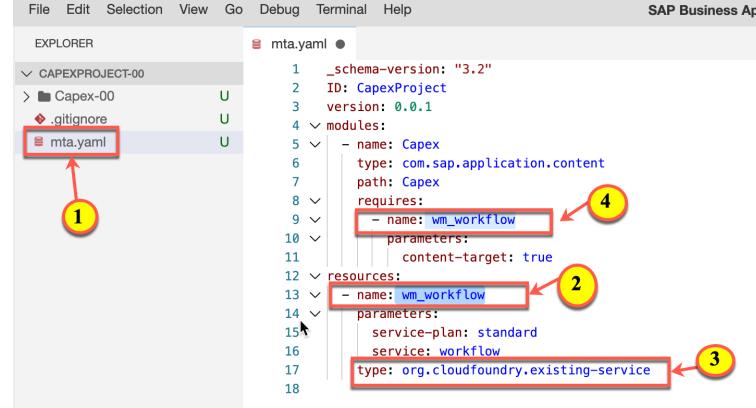
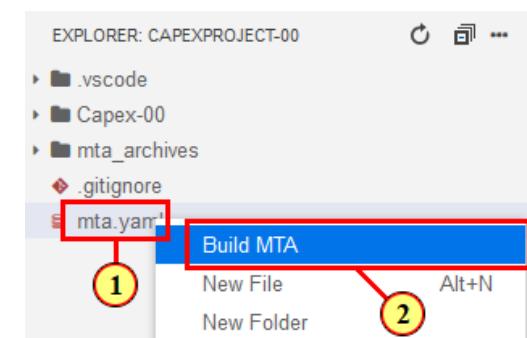
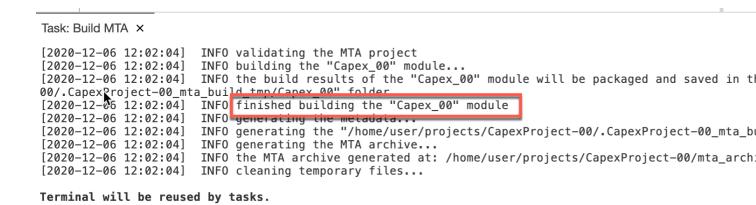
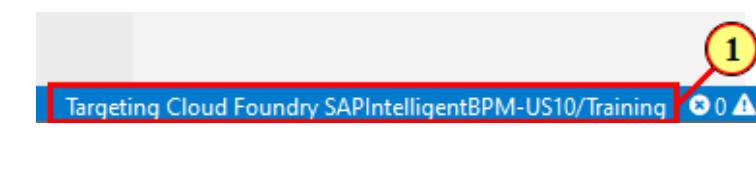
Explanation	Screenshot
<p>162. Enter Yes as Name.</p> <p>163. Enter <code>#{context.approvalStepsResult.Result[0].Approvers.cm_required}</code> as Condition.</p> <p>⚠️ Be careful during copying. There are no spaces in the Condition text. Sometimes during copy, spaces are added automatically. In this 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 Terminating End Event and place it above the Approval result? gateway. Name it End (Rejection).</p> <p>165. Drag a new Sequence Flow Connection from Approval result? onto the End (Rejection) event. Name it Reject and select Default as condition.</p>	 <p>The diagram illustrates the workflow logic. It starts with a decision diamond 'Approval required?'. If 'no', it leads to the end event 'End (Approval)'. If 'yes', it flows through an 'Approval Task (Central Manager)' and a 'Handle Approval (Central Manager)' activity. This leads to a gateway. From the gateway, one path leads to the terminating end event 'End (Rejection)' (labeled 1), and another path leads to 'End (Approval)' (labeled 2). A sequence flow labeled 'reject' connects the gateway to 'End (Rejection)'.</p>
<p>166. Select the Sequence Flow Connection from Approval result? onto the End (Approval)</p> <p>167. Name it Approve and enter <code>#{context.decision == "approve"}</code> as condition.</p>	 <p>The diagram shows the continuation of the workflow after the rejection path. It includes an 'Approval required?' decision diamond, an 'Approval Task (Central Manager)' activity, a 'Handle Approval (Central Manager)' activity, and a gateway. From the gateway, one path leads to 'End (Rejection)' (labeled 1) and another to 'End (Approval)' (labeled 2). A sequence flow labeled 'Approve' connects the gateway to 'End (Approval)'. The 'Approve' sequence flow is selected in the properties dialog, which shows the 'Name' field containing 'Approve' and the 'Condition' field containing 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 Approval result? onto the Rework Task.</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 Rework 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 Workflow Properties.</p> <p>172. In Workflow Properties, enter Subject as CAPEX Approval for '#{context.Title}'</p> <p>173. Enter Business Key as <code>#{context.RequestId}</code></p> <p>174. Select the ATTRIBUTES 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 Workflow Properties 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 Add.</p> <p>176. Enter Currency as ID.</p> <p>177. Enter Currency as Label.</p> <p>178. Enter Value: \${context.Investment.Currency} as Value.</p>	
<p>179. In the same way, add the following fields:</p> <p>ID: TotalCost Label: Total Cost Value: \${context.Investment.TotalCost}</p> <p>ID: CAPEX Label: CAPEX Value: \${context.Investment.CAPEX}</p> <p>ID: ROI Label: ROI Value: \${context.Investment.ROI}</p> <p>ID: Country Label: Country Value: \${context.Investment.Country}</p> <p>ID: Type Label: Type Value: \${context.Investment.Type}</p> <p>ID: Title Label: Title Value: \${context.Title}</p> <p>ID: RequestId Label: Request Id Value: \${context.RequestId}</p>	
<p>Note: Currently only String type is supported.</p> <p>⚠️ Be careful during copying. There are no spaces in the Value 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(()) --> Prep[Prepare Data] Prep --> Ret[Retrieve Approval Steps] Ret --> Decision{Approval required?} Decision -- yes --> Local[Approval Task Local Manager] Local --> LocalEnd[End (Reject) / Approve] LocalEnd -- no --> Central[Approval Task Central Manager] Central --> CentralEnd[End (Reject) / Approve] CentralEnd -- no --> End(()) LocalEnd -- yes --> Decision CentralEnd -- yes --> Decision LocalEnd -- join --> Central LocalEnd -- reject --> Rework[Rework Task] CentralEnd -- reject --> Rework </pre>
<p>Note: If you did not use Auto Save, 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 below to this highlighted item.</p>

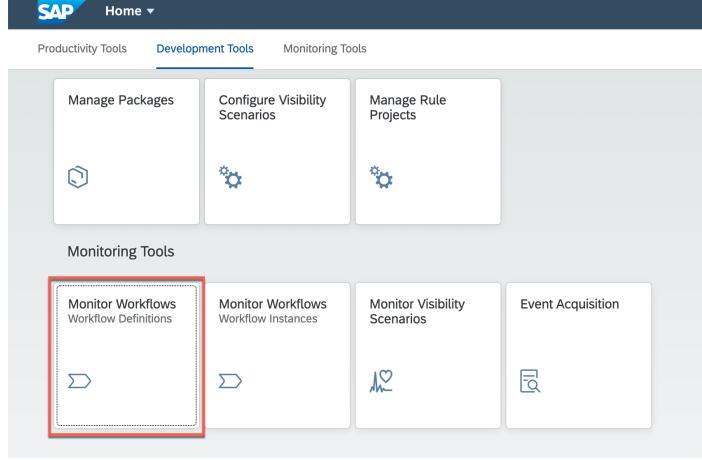
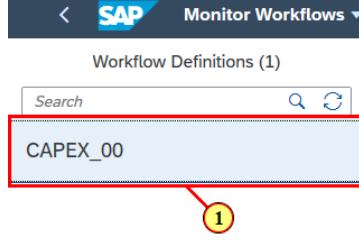
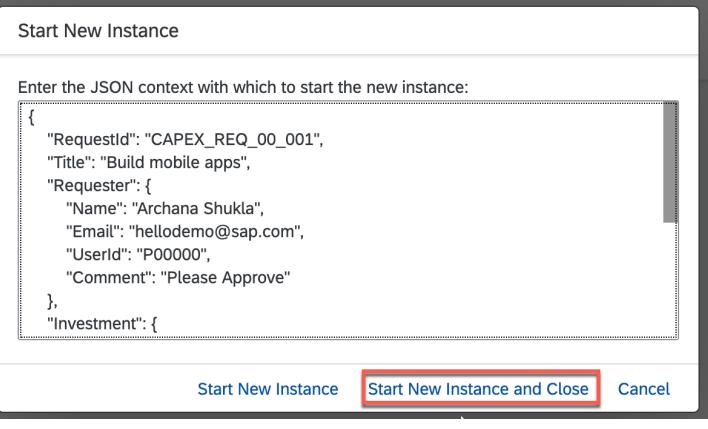
BUILD AND DEPLOY WORKFLOW

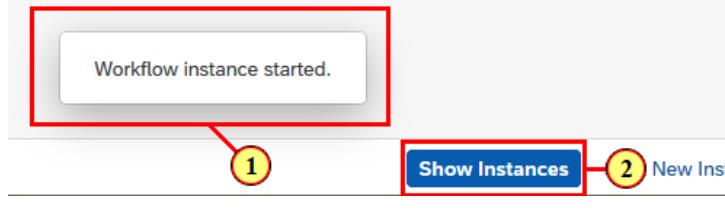
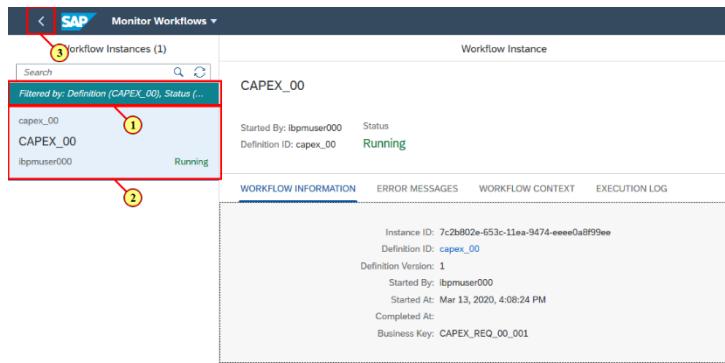
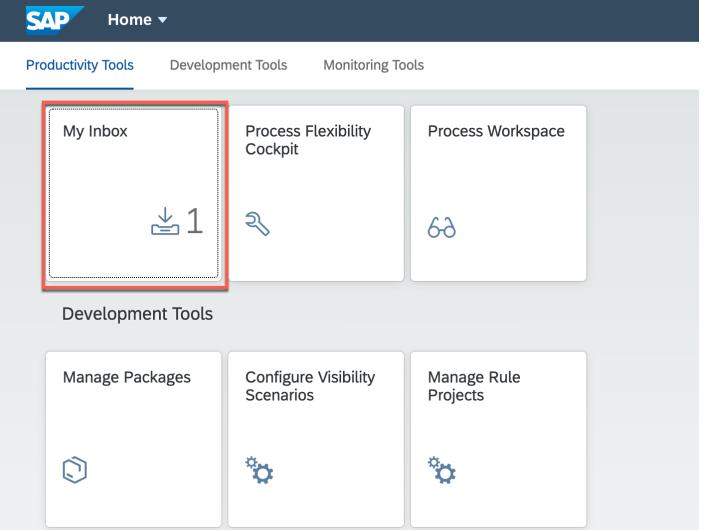
Explanation	Screenshot
<p>180. Double-click mta.yaml to open it.</p> <p>Note: The default MTA will mention a reference to a new workflow service instance. When you ran the Booster in the setup section, the workflow service instance is already created and so change the resource to the existing instance name.</p> <p>181. Change the resource name to wm_workflow.</p> <p>182. Change the resource type to org.cloudfoundry(existing-service).</p> <p>183. Change the resource reference to wm_workflow.</p> <p>184. Close the tab and ensure the file is saved.</p>	 <pre> File Edit Selection View Go Debug Terminal Help SAP Business Application Studio EXPLORER CAPEXPROJECT-00 Capex-00 .gitignore mta.yaml mta.yaml 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 View → Problems menu option to see the modelling errors. Fix those first.</p> <p>185. To build the project, right-click on mta.yaml to open the context menu.</p> <p>186. Select Build MTA.</p>	
<p>187. Ensure that the Build MTA 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/.tmp/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.archive" [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_archive [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>	

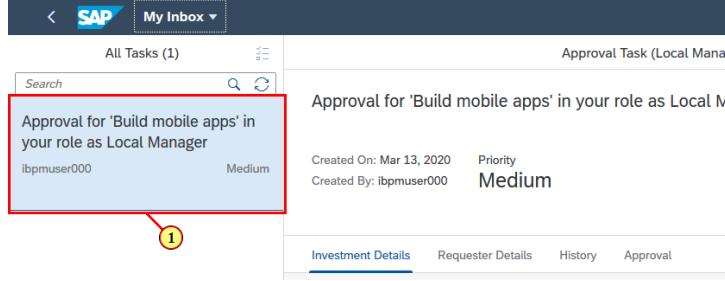
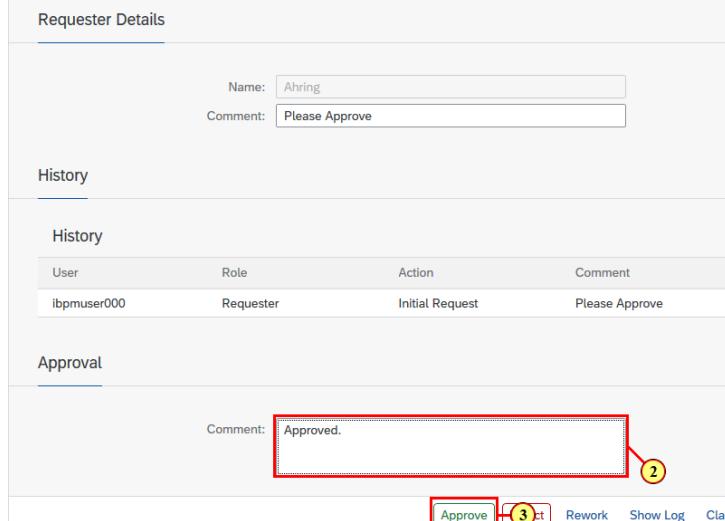
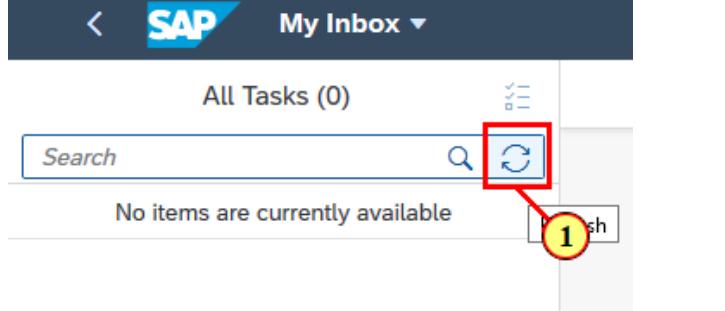
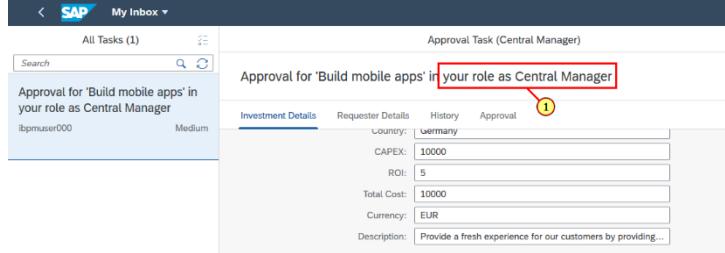
Explanation	Screenshot				
<p>You can get the need information about the Login URL, organization name and space name from you trial account.</p> <p>You have to login with your trial user email and password.</p>	<p>SAP Cloud Platform Cockpit</p> <p>Subaccount: trial</p> <p>Subdomain: db73ff38trial ID: bc6567da-cabb-49f2-998d-6177896d0d93</p> <p>Cloud Foundry Environment Entitlements</p> <p>Org Name: db73ff38trial Org ID: 20e51be8-b1ed-4788-9146-89deff83faac Members: 1 API Endpoint: https://api.cf.eu10.hana.ondemand.com</p> <p>Spaces (1)</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Applications</th> </tr> </thead> <tbody> <tr> <td>dev</td> <td>0</td> </tr> </tbody> </table> <p>Disable Cloud Foundry</p> <p>Entitlements</p>	Name	Applications	dev	0
Name	Applications				
dev	0				
<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"> open a New Terminal window and enter command cf login --sso You will then be given a URL to get Temporary Authentication Code 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). If the code is correct, you will be prompted to select the organization else it will throw Invalid passcode. For latter case, get the passcode again, paste and click Enter key. Enter the organization index number or the organization name from the list. If you have only 1 space then it will automatically be selected, else it will ask you to enter the space as well. When your connection is done, you will see the connected cloud account information in the blue-bar. 	<pre>user@workspaces-ws-6229h-deployment-7bffbcb8494-r4fc5: ~/projects x 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): ■</pre> <pre>user@workspaces-ws-6229h-deployment-7bffbcb8494-r4fc5: ~/projects x 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</pre> <p>Org (enter to skip): 11 Targeted org db73ff38trial Targeted space dev</p> <p>API endpoint: https://api.cf.eu10.hana.ondemand.com (API version: 3.91.0) User: [REDACTED]@trial Org: db73ff38trial Space: dev user: projects \$ ■</p> <p>APPLICATION MODELER Targeting CF db73ff38trial/dev</p>				

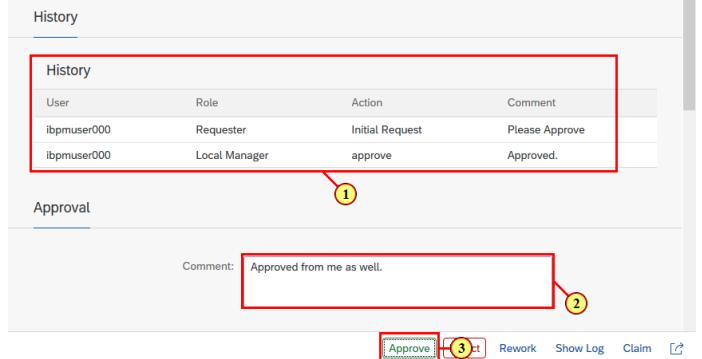
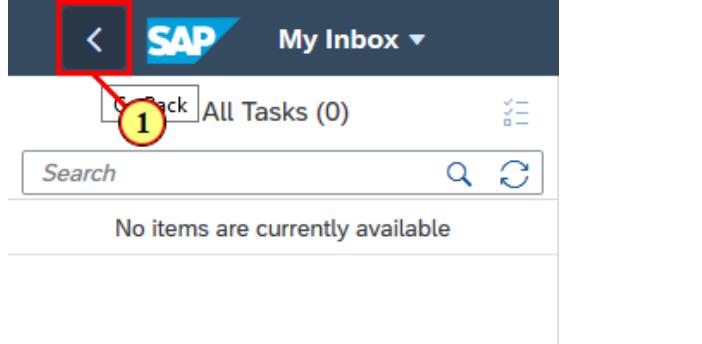
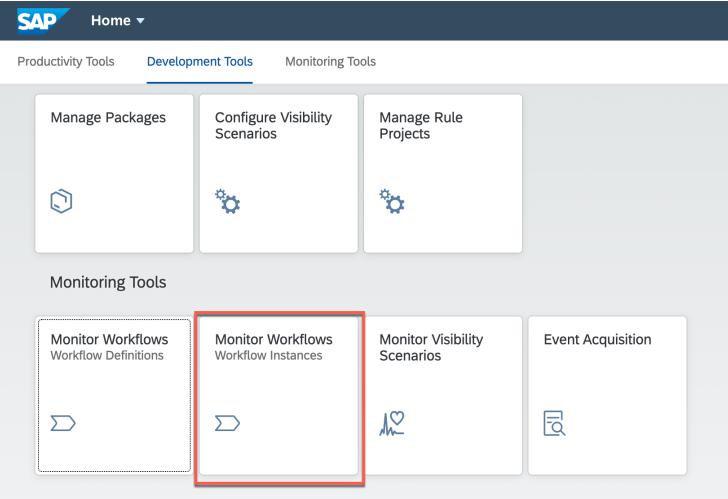
Explanation	Screenshot
<p>189. Select the newly created directory mta_archives to open it.</p> <p>190. Right-click on the created file CapexProject-00_0.0.1.mtar to open the context menu.</p> <p>191. Select Deploy MTA Archive.</p>	
<p>192. Ensure that the Deploy MTA Archive task exits with the successful message indicating that the deployment has finished.</p> <p>Note: Your deployment console should look similar to this with the "Process finished" 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>> Executing task: cf deploy /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar < Deploying multi-target app archive /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar in org db73ff38trial / space dev as jsp.com... There is an ongoing operation for multi-target app CapexProject-00. Do you want to abort it? (y/n)> 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 version: "3" 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"... Service key "Capex_00-wm_workflow-credentials" for service "wm_workflow" already exists Some services are deployed to multiple 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 "Content deployment for module Capex_00 failed", 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 y if you get option saying: There is an ongoing operation for multi-target app CapexProject-00. Do you want to abort it? (y/n)?</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-9381-2b0287ee4cd 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" 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-9381-2b0287ee4cd 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" 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) 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. > Executing task: cf deploy /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar < Deploying multi-target app archive /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar in org db73ff38trial / space dev as ... There is an ongoing operation for multi-target app CapexProject-00. Do you want to abort it? (y/n)> y</pre>

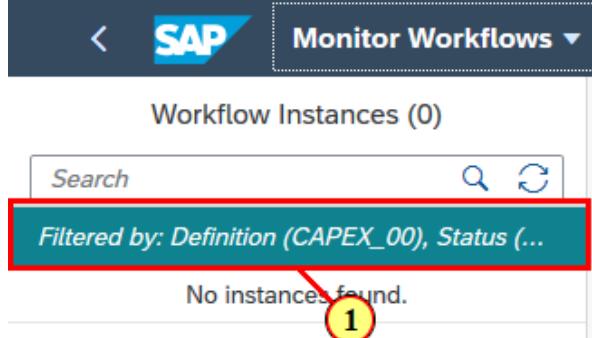
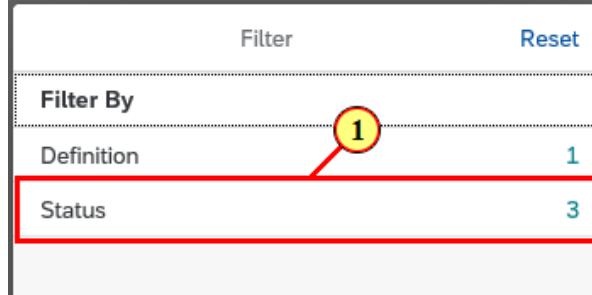
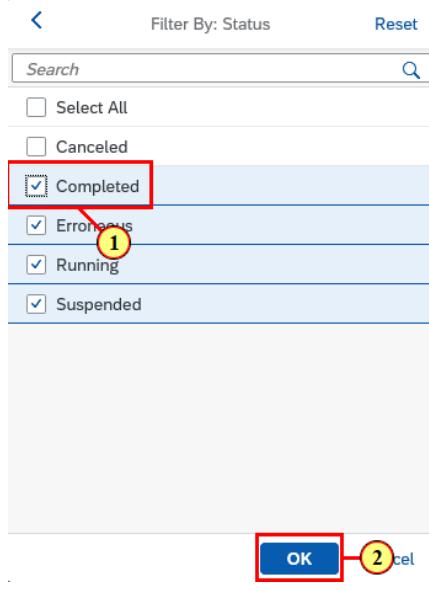
EXECUTE AND MONITOR WORKFLOW

Explanation	Screenshot
<ol style="list-style-type: none"> 1. Open the Workflow Management Launchpad 2. Log On with your trial username and password. 3. Select the tile Monitor Workflows - Workflow Definitions. 	
<p>Note: Monitor Workflows application has Master-Detail layout where you see all the workflows deployed on left. Ensure that the workflow you just deployed - CAPEX_00 - is existing.</p> <ol style="list-style-type: none"> 4. Select the workflow. 	
<ol style="list-style-type: none"> 5. Select Start New Instance. <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 Workflow API.</p>	
<ol style="list-style-type: none"> 6. Check the start payload 7. Click Start New Instance and Close <p>Note: The other <i>Start New Instance</i> option is to start multiple instances of the workflow from the same popup.</p>	 <pre data-bbox="763 1558 1073 1805"> { "RequestId": "CAPEX_REQ_00_001", "Title": "Build mobile apps", "Requester": { "Name": "Archana Shukla", "Email": "hello@demo.sap.com", "UserId": "P00000", "Comment": "Please Approve" }, "Investment": { ... } } </pre>

Explanation	Screenshot
<p>A popup confirms that the workflow instance has been started.</p> <p>8. Select Show Instances to switch to the Monitor Workflows - Workflow Instances application where you can view all the running, completed, suspended workflow instances and their details.</p>	
<p>Note: When coming from Show Instances, a Filter is automatically added to the specific Workflow Definition.</p> <p>9. Select the newly started workflow instance and look at its details.</p> <p>10. Select SAP icon to go back to the Home Page of the Workflow Management Launchpad.</p>	
<p>11. Select the tile for My Inbox. 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 Approval section.</p> <p>14. Enter a Comment of your choice.</p> <p>15. Click Approve.</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 Refresh .</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> <table border="1"> <tr> <td>Country: Germany</td> </tr> <tr> <td>CAPEX: 10000</td> </tr> <tr> <td>ROI: 5</td> </tr> <tr> <td>Total Cost: 10000</td> </tr> <tr> <td>Currency: EUR</td> </tr> <tr> <td>Description: Provide a fresh experience for our customers by providing...</td> </tr> </table>	Country: Germany	CAPEX: 10000	ROI: 5	Total Cost: 10000	Currency: EUR	Description: Provide a fresh experience for our customers by providing...		
Country: Germany									
CAPEX: 10000									
ROI: 5									
Total Cost: 10000									
Currency: EUR									
Description: Provide a fresh experience for our customers by providing...									

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

Explanation	Screenshot
<p>21. Be default, Monitor Instances application only shows Running, Suspended and Erroneous workflow instance. To see the completed workflow, click on the line showing the currently active Filters or select the Filter 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 and a refresh button. 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 ellipsis. Below this, it says "No instances found." with another yellow circle labeled "1" pointing to the text.</p>
<p>22. In the popup, select Status.</p>	 <p>The screenshot shows a "Filter By" dialog. It has a "Filter" button and a "Reset" button at the top. Below that is a "Filter By" section with a "Definition" dropdown containing the value "1" and a "Status" dropdown containing the value "3". A red box highlights the "Status" dropdown, and a yellow circle labeled "1" points to the number "1" next to the "Definition" dropdown.</p>
<p>23. Select Completed to also show already completed workflow instances.</p> <p>24. Select OK to activate the changed filter settings.</p>	 <p>The screenshot shows a "Filter By: Status" dialog. It has a "Search" bar and a "Reset" button at the top. Below is a list of filter options: "Select All" (unchecked), "Canceled" (unchecked), "Completed" (checked), "Erroneous" (checked), "Running" (checked), and "Suspended" (checked). A red box highlights the "Completed" checkbox, and a yellow circle labeled "1" points to it. At the bottom right is an "OK" button, which is also highlighted with a red box and a yellow circle labeled "2" points to it.</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>Congratulations! You completed the Workflow exercise.</p> <p>In this exercise, you learnt how to:</p> <ul style="list-style-type: none"> ○ Model workflow from SAP Business Application Studio. ○ Build & deploy workflow ○ Start workflow instance from Monitor Definitions ○ Track the workflow instances from Monitor Instances ○ Process your work items from My Inbox 	<p>The screenshot shows the SAP Monitor Workflows interface. In the center, there's a table of workflow instances. One instance, 'CAPEX_00', is highlighted with a red box and has a yellow circle with the number '1' pointing to it. To the right of the table, there's a detailed view of the 'CAPEX_00' instance. It shows the 'Status' as 'Completed' with a red box and a yellow circle with '2' pointing to it. Below the status, there's a summary of the instance: Instance ID: 7c2b802e-653c-11ea-9474-e0e0a08f99ee, Definition ID: capex_00, Definition Version: 1, Started By: ibpmuser000, Started At: Mar 13, 2020, 4:08:24 PM, Completed At: Mar 13, 2020, 4:12:35 PM, Business Key: CAPEX_REQ_00_001. Below this are sections for 'ERROR MESSAGES' (no messages) and 'WORKFLOW CONTEXT' (with some JSON code).</p>

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

© 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 for additional trademark information and notices.