

PUBLIC

Build a Workflow from Scratch with SAP Cloud Platform Workflow Management

DEV163

Exercise 2.2 | Integrate Business Rules with Workflow
Archana Shukla / SAP

TABLE OF CONTENTS

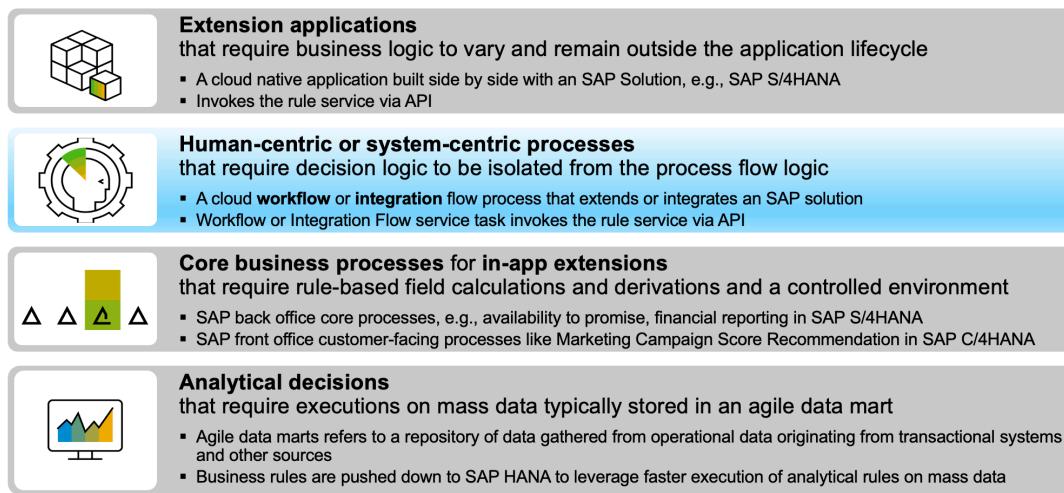
OVERVIEW	3
INTEGRATE BUSINESS RULES WITH WORKFLOW	4
EXECUTE AND MONITOR WORKFLOW.....	9
APPENDIX	11
Import Sample Workflow.....	11

OVERVIEW

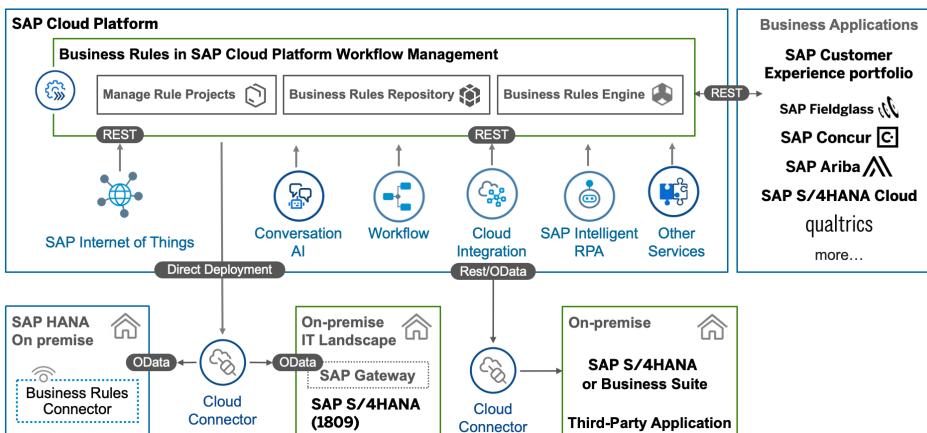
Estimated time: 30 minutes

Prerequisite: You have to run Exercise 2.1 before running this exercise.

Business Rules in SAP Cloud Platform Workflow Management enables business users & application developers to automate their decisions and enhance their decision-making experience. Integration and consumption are the key. There are 4 **consumption patterns** that you can use to leverage the power of business rules capabilities in your applications/solutions.

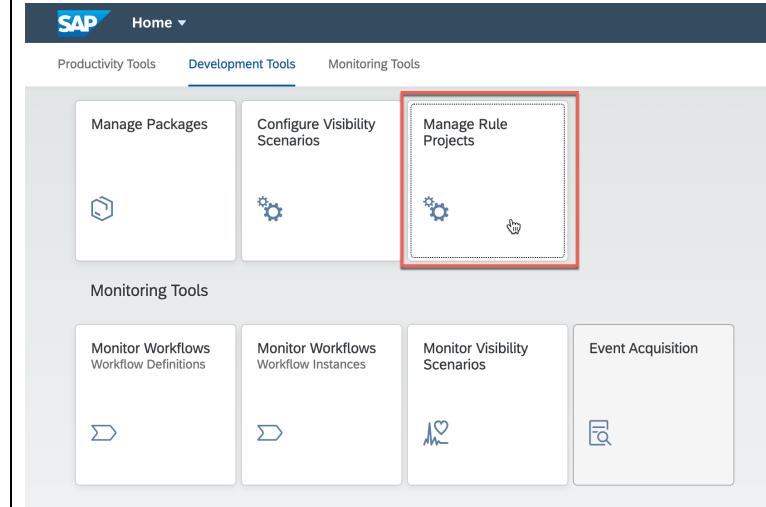
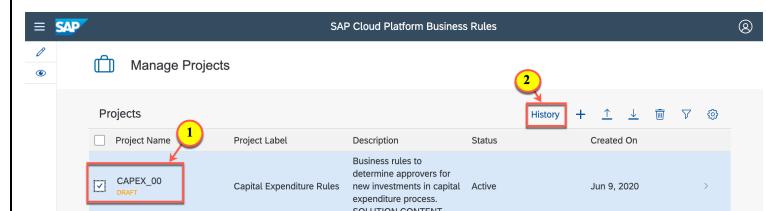
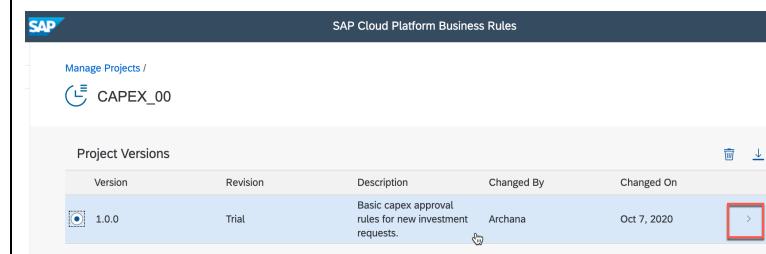
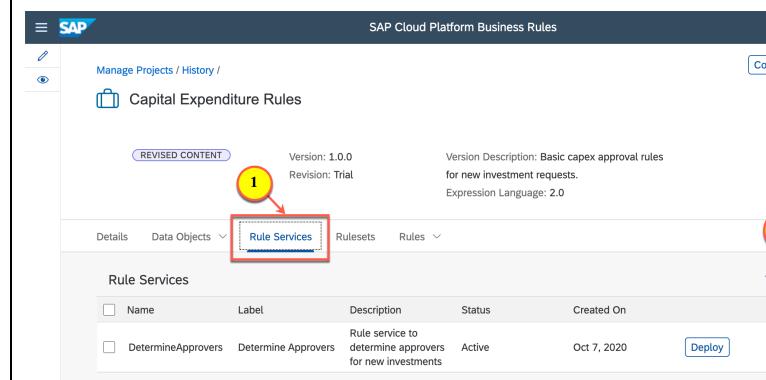


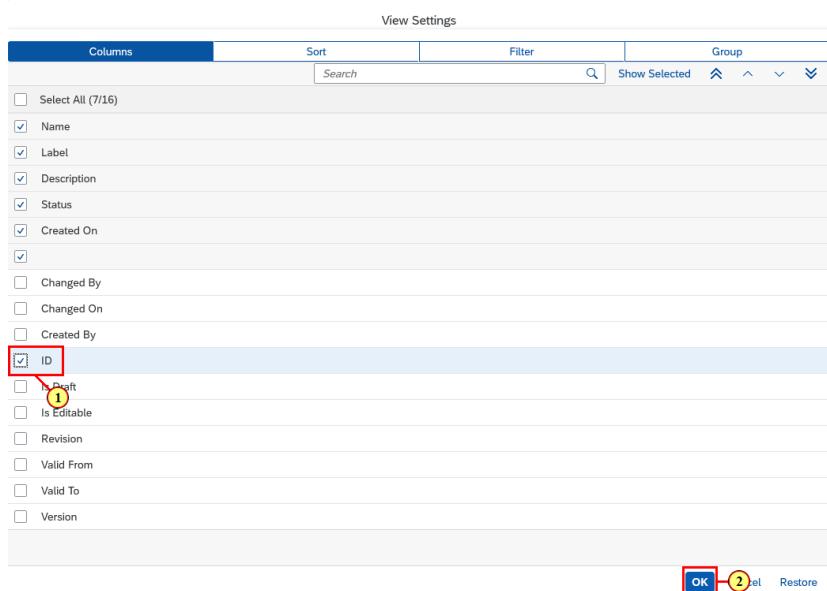
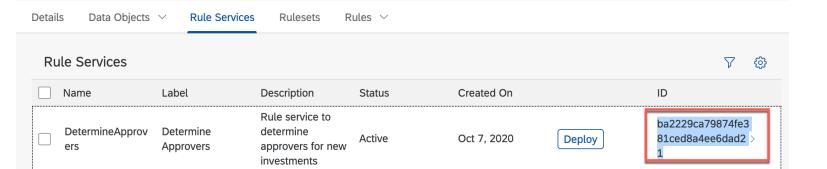
Pattern 2 talk about how business rules can be used to abstract the decision logic in process orchestration scenarios. **Workflow** and **Business Rules** work harmoniously to provide an unmatched support to automate decisions in workflow such that the business users can easily modify the rules without changing the process flow. **Business rule service can be accessed in Workflow using a Service Task**.



In this exercise, you will integrate the approver determination rules you build in earlier exercise with the capex approval workflow. For that you will configure the Service Task with the details of the Business Rules API.

INTEGRATE BUSINESS RULES WITH WORKFLOW

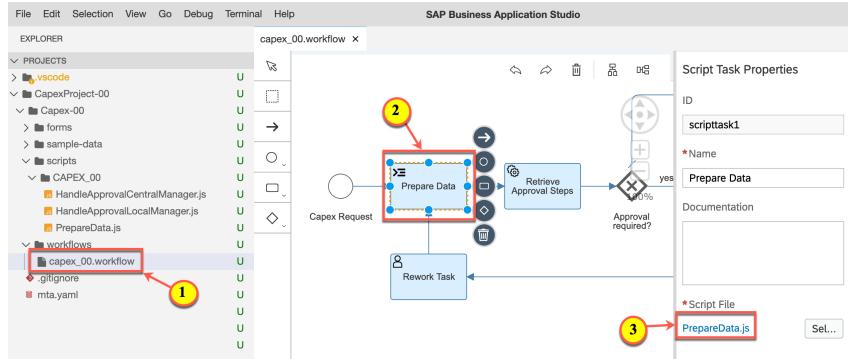
Explanation	Screenshot
<p>There are two kind of rule invocation APIs (a) versioned and (b) working-set. In this exercise, we will integrate business rules with workflow using versioned invocation API. For that, you have to get the rule service ID of the versioned rule service.</p> <ol style="list-style-type: none"> From Workflow Management Launchpad Home page, open Manage Rule Projects application. 	
<ol style="list-style-type: none"> Select your business rule project and click on History. 	
<ol style="list-style-type: none"> Click to navigate into the version 1.0.0. 	
<ol style="list-style-type: none"> Click Rule Services tab. Select Settings . 	

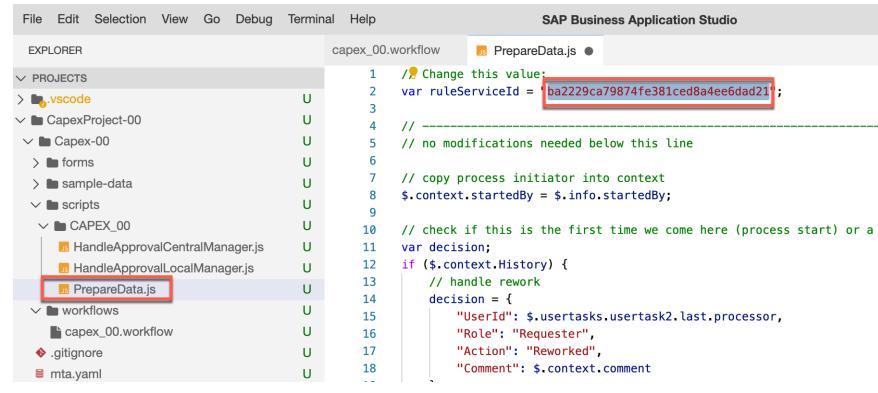
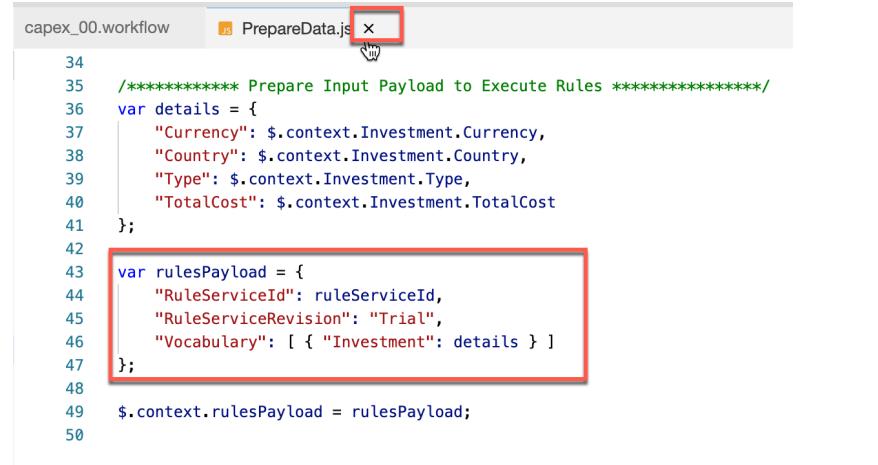
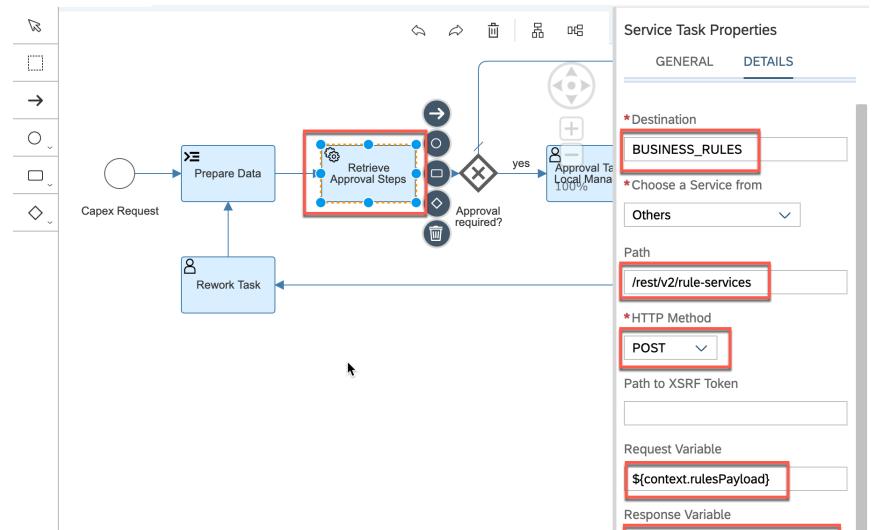
Explanation	Screenshot
<p>6. Select the ID column which will display the ID of the rule service.</p> <p>7. Select OK.</p>	
<p>8. Copy the Rule Service ID and paste it in a notepad.</p>	

Now we will use the copied rule service information to configure the workflow.

If you have not executed *Exercise 1 on Workflow* and are coming directly to execute *Exercise 2* then you have to first import the workflow in SAP Business Application Studio.

To do so, follow the Appendix | Import Sample Workflow.

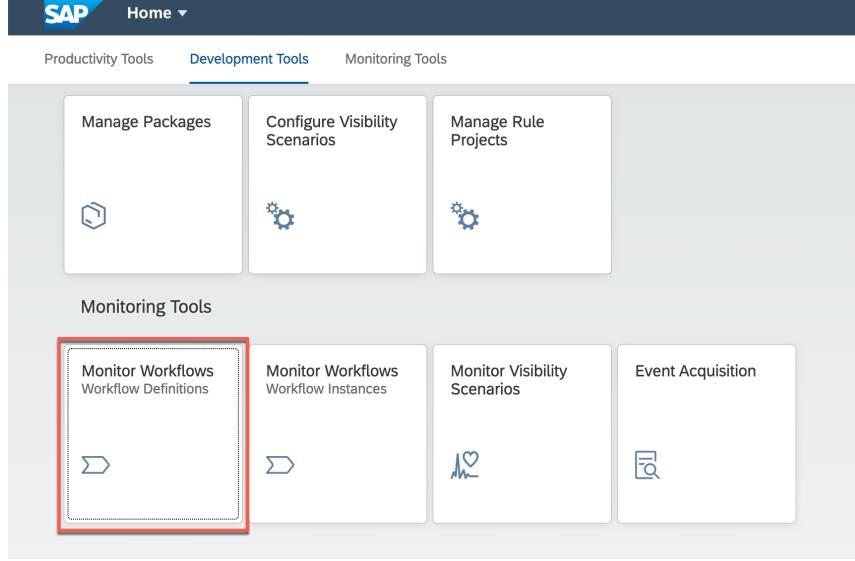
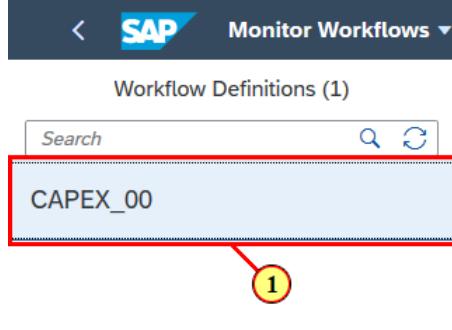
<p>9. Expand the imported workflow project and click to open the workflow and select Prepare Data script task.</p> <p>10. From the service task properties, select the script file PrepareData.js to open it.</p>	
---	--

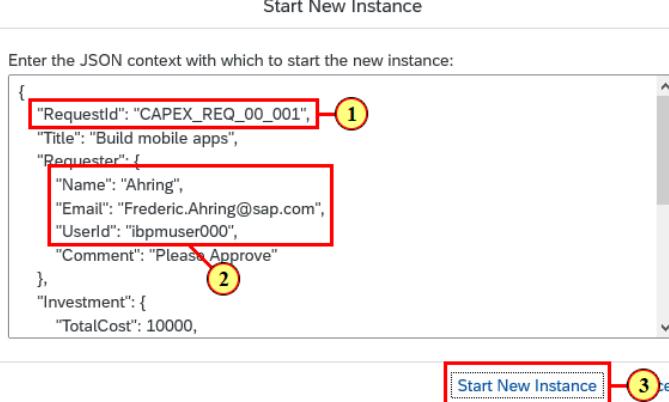
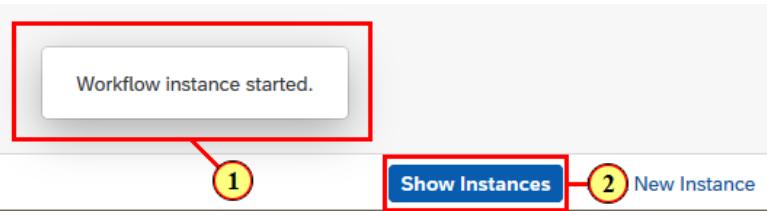
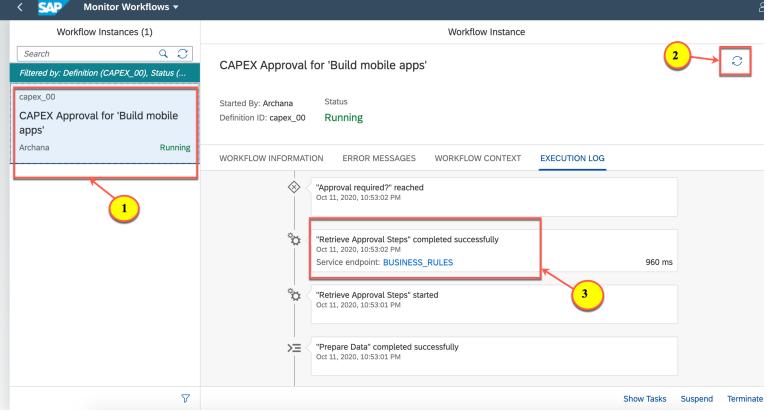
Explanation	Screenshot
<p>11. Paste the Rule Service ID into the variable.</p>	 <pre data-bbox="608 420 1462 747"> File Edit Selection View Go Debug Terminal Help capex_00.workflow PrepareData.js EXPLORER PROJECTS > capex_00 CapexProject-00 Capex-00 forms sample-data scripts CAPEX_00 HandleApprovalCentralManager.js HandleApproval.localManager.js PrepareData.js workflows capex_00.workflow .gitignore mta.yaml 1 // Change this value! 2 var ruleServiceId = "ba2229ca79874fe381ced8a4ee6dad21"; 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 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 ... </pre>
<p>Scroll down in the javascript file and notice how the input payload is constructed to call the business rules API in the workflow script task.</p> <p>12. Save and close ✕ the file.</p>	 <pre data-bbox="608 813 1462 1229"> 34 ***** 35 var details = { 36 "Currency": \$.context.Investment.Currency, 37 "Country": \$.context.Investment.Country, 38 "Type": \$.context.Investment.Type, 39 "TotalCost": \$.context.Investment.TotalCost 40 }; 41 42 var rulesPayload = { 43 "RuleServiceId": ruleServiceId, 44 "RuleServiceRevision": "Trial", 45 "Vocabulary": [{ "Investment": details }] 46 }; 47 48 \$.context.rulesPayload = rulesPayload; 49 50 </pre>
<p>13. Click to open the workflow again and select Retrieve Approval Steps service task.</p> <p>Notice the service task properties:</p> <ul style="list-style-type: none"> BUSINESS_RULES destination is created when you ran the Booster. This destination contains the credentials to run business rules invocation API. Path is the relative path of the versioned API Request variable is the input payload as constructed in the script task before. Response variable is where the output of the rules will be appended. 	 <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>

Explanation	Screenshot
<p>14. From the workflow model, select Approval Task (Local Manager) user task.</p> <p>In User Task Properties, notice the Users section. It is directly consuming the output from the business rules execution from the response variable.</p>	
<p>15. Save your changes, build the workflow by right-click on mta.yaml to open the context menu and click Build MTA.</p>	
<p>16. Ensure that the Build MTA task exits with code 0, indicating the build is completed.</p>	
<p>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 click on the blue-bar and follow the wizard to enter the details.</p>	

Explanation	Screenshot
<p>17. You can get the needed information about the Login URL, organization name and space name from your 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>Cloud Foundry Environment</p> <p>Org Name: db73ff38trial (highlighted)</p> <p>Org ID: 20e51be8-b1ed-4788-9146-89def783faac</p> <p>Members: 1</p> <p>API Endpoint: https://api.cf.eu10.hana.ondemand.com (highlighted)</p> <p>Disable Cloud Foundry</p> <p>Organisation Name</p> <p>Spaces (1)</p> <p>Name Applications</p> <p>dev 0</p> <p>Login URL</p> <p>Space Name</p>
<p>18. Select the newly created directory mta_archives to open it.</p> <p>19. Right-click on the created file CapexProject-00_0.0.1.mtar to open the context menu.</p> <p>20. Select Deploy MTA Archive.</p>	<p>EXPLORER: CAPEXPROJECT-00</p> <ul style="list-style-type: none"> .vscode Capex-00 mta_archives (highlighted) <ul style="list-style-type: none"> CapexProject-00_0.0.1.mtar (highlighted) .gitignore mta.yaml <p>Deploy MTA Archive</p> <p>New File</p> <p>New Folder</p>
<p>21. Ensure that the Deploy MTA Archive task exits with code 0 indicating that the deployment has finished.</p> <p>Note: You can also click on the □ icon in the lower right corner to toggle display of the console panels.</p>	<p>Task 'Deploy MTA Archive (MTA)' has exited with code 0.</p> <p>Task 'Deploy MTA Archive (MTA)' has been started.</p> <p>Ln 18, Col 1 LF UTF-8 Spaces: 2 YAML (2)</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> ● Problems > Task: Deploy MTA Archive > Task: Build MTA Deploying multi-target app archive /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_iworkflow@gmail.com... Uploading 1 files... /home/user/projects/CapexProject-00/mta_archives/CapexProject-00_0.0.1.mtar OK Deploying in org "SAPIntelligentBPM-US10" and space "Training" Detected MTA schema version: "3" No deployed MTA detected - this is initial deployment Detected new MTA version: "0.0.1" Service key "Capex-00-BPMWorkflow-credentials" for service "BPMWorkflow" already exists Uploading content module "Capex-00" in target service "BPMWorkflow"... Deploying content module "Capex-00" in target service "BPMWorkflow"... Skipping deletion of services, because the command line option "--delete-services" is not specified. Process finished. Use "cf dmol -i c29f3b28-96ba-11ea-920e-eccc0a81d618" to download the logs of the process. </pre>

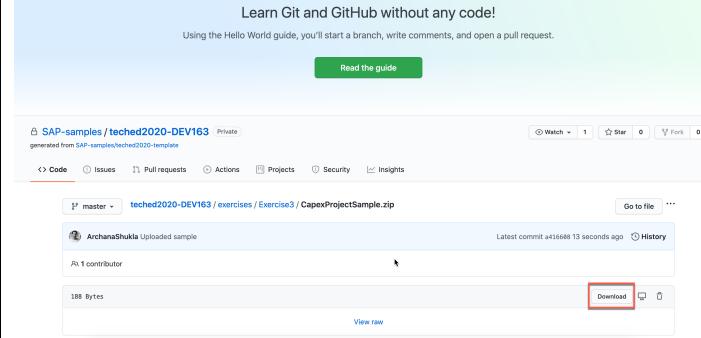
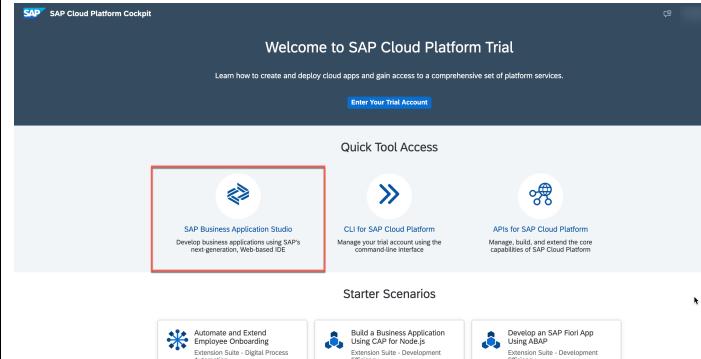
EXECUTE AND MONITOR WORKFLOW

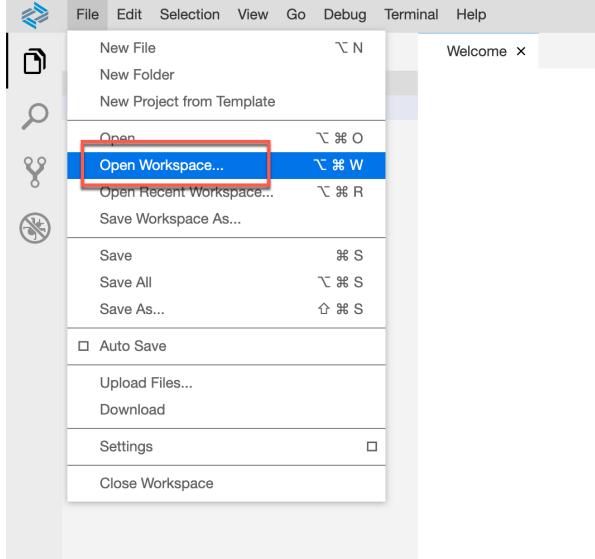
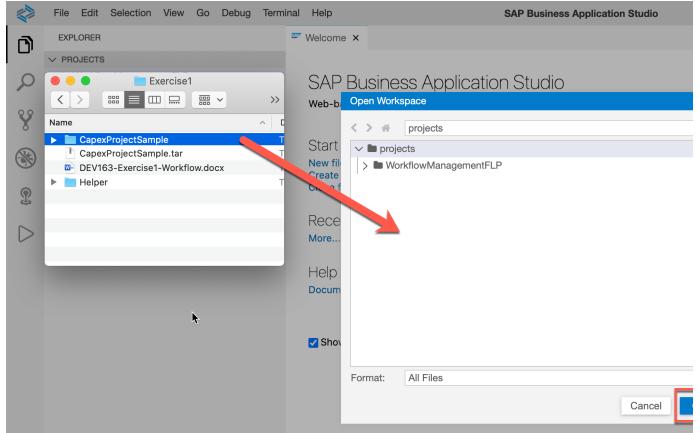
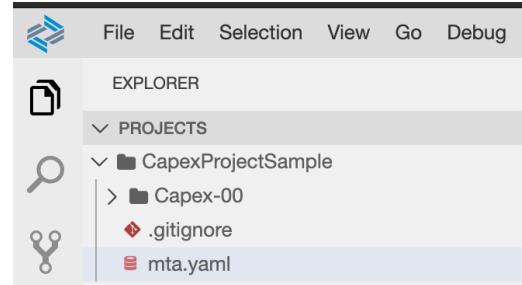
Explanation	Screenshot
<ol style="list-style-type: none"> 1. Open the Workflow Management Launchpad Home Page. 2. Log On with your trial username and password. 3. Select the tile Monitor Workflows - Workflow Definitions. 	
<p>Note: Monitor Workflows application has Master-Detail layout where you see 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. 	

Explanation	Screenshot
<p>6. Modify the RequestId in the JSON to any string.</p> <p>7. Adapt the following fields in the JSON element.</p> <p>Requester:</p> <ul style="list-style-type: none"> o Name can be anything o Email should be your trial user email ID o UserId is your trial user id <p>8. Select Start New Instance.</p>	
<p>A popup confirms that the workflow instance has been started.</p> <p>9. 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>10. Select the latest created workflow instance and click on Refresh button on top-right corner.</p> <p>11. From the EXECUTION LOG, you will notice that the service task Retrieve Approval Steps is completed successfully which executes the business rules service.</p>	
<p>You can also switch to WORFKLOW CONTEXT and see the output of the rule execution collected inside approvalStepResult element.</p>	<pre data-bbox="612 1679 954 1869"><code>"approvalStepsResult": { "Result": [{ "Approvers": { "cm_required": true, "cm_userid": "ibpmuser000", "lm_required": true, "lm_userid": "ibpmuser001" } }] }</code></pre>

APPENDIX

Import Sample Workflow

Explanation	Screenshot
<p>1. Download CapexProjectSample.zip project from given GitHub link in your local file system and extract the files.</p> <p>https://github.com/SAP-samples/teched2020-DEV163/blob/master/exercises/Exercise3/CapexProjectSample.zip</p>	
<p>2. Logon to SAP Cloud Platform Trial Home Page with your trial username and password.</p> <p>3. Choose SAP Business Application Studio.</p>	

Explanation	Screenshot
<p>4. Click on the Dev Space to go into the workspace and from File menu choose File Open Workspace.</p> <p>Note: If you have no dev space, click the Create Dev Space with:</p> <ol style="list-style-type: none"> 1. <i>Application Type</i> as SAP Fiori. 2. <i>Extension</i> as Workflow Management. <p>Else select PLAY  to start the space if you see the status as STOPPED.</p>	
<p>5. Drag and drop the extracted CapexProjectSample folder in the Open Workspace dialog and click Open.</p>	
<p>6. You will see that the CapexProjectSample project is imported into your workspace.</p>	

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.