Hierarchies

1. Replace the Rent Scooter with a 'Process Task'

Currently the rental of our scooter task is a simple Human Task. But now we have the process which is taking care about the rental. Let's replace first the rent scooter task with a process task.

1. Click on the 'Rent Scooter' Human Task.
2. Locate the wrench icon directly below the task (hover text is 'Transform')
3. Select Tasks and then Process Task
4. Save to validate the result.

2. Link the existing process to the case

In the last exercise we created a process. Now let's use the process inside the case. Let's add the 'Process reference' and select a reference.

1. Click on the Process Task in the diagram.
2. In the property list, search for 'Process reference' which is located in the 'General' section.
3. Click on 'link to an existing one'.
4. Search 'FLW-P001 Rent Scooter' in the list.
5. Click on 'FLW-P001 Rent Scooter' to select the item.
6. Save the model to validate the result.

3. Adapt the max speed condition to use the parent variable

The scope of the variables has changed and the process itself no longer has a variable 'maxSpeed'. Change the condition to read the variable from the parent or root instead.

1. Click on the Sequence Flow with the condition (after the gateway).
2. Locate in the properties on the right side the option for the 'Condition expression'.
3. Click on 'Condition expression' and then click on the the 'Condition Builder' icon on the top right.
4. Search for the property root.maxSpeed in the 'Select variable...' field.
5. Select 'Greater than or equal' for the operator.
6. Enter the value 15 into the 'Enter number value...' field.
7. Press 'Ok' to add the condition.
8. Save the model to validate the result.

4. Publish the changes

To publish your app, click on the cloud icon featuring a small arrow to the top.

1. Locate the top menu bar which above the canvas.

2. Locate the publish button in the toolbar. This is the cloud symbol with a little arrow inside.
3. Press this button and a popup window will appear.
4. In the popup window, confirm the deployment target and the app. In this environment you do not really have a choice and you can simply say 'Publish'.
5. Once you completed this exercise you will be redirected automatically to Flowable Work. In a real application you would now need to switch to 'Flowable Work' manually.

5. Create a scooter and rent it twice

In Flowable Work, create a new case and rent the scooter twice. Once you have done that, decommission the scooter to complete the exercise.

1. Select 'New' below 'Work' or 'Work' below 'New'
2. Select the case 'FLW-C001 Scooter'.
3. In the Start Form, enter any data you want.
4. Press 'Start' and run through the tasks.
5. Once ready, press the rent scooter button.
6. Complete the rent scooter process.
7. Once done, execute the rent scooter one more time.
8. Complete the rent scooter process.
9. Once you then decommission your scooter, the exercise will be done.

6. Create a Form for the 'Return Scooter' task

Create a form and reference it as the task form of the 'Return Scooter' task within the process. Simply click on the task and select 'Form reference'. The form must have the key 'FLW-F013'.

1. Click on the process task and open the process.
2. Click on the task 'Return Scooter'.
3. In the property list, search for 'Form reference' which is located in the 'General' section.
4. Enter 'FLW-F013 Return Scooter' as name.
5. Enter as key 'FLW-F013' as key.
6. Press the 'Create' button to create the form.
7. Save the model to validate the result.

7. Create a single select and name it 'Damages'

The next step is to create a single select and name it 'Damages'. Press save to update the validation list on the right.

1. Look for the 'Select (Single)' component in the palette on the left side.
2. Drag and drop the Select (Single) into the form model.

3. Enter the name 'Damages'.
4. Save the model to validate the result.

8. Provide the list of items

As a next step, we need to add the items to the select. Therefore you can search for 'Items' and add 'None', 'Small', and 'Large'. The text should be the name as provided and the value the same in lowercase.

1. Select the 'Damages' field.
2. Search the properties on the right side for 'Items'.
3. Press the 'Add item' button.
4. Write 'none' in the field 'Value' and 'None' in the field 'Text'.
5. Press 'Add item' again and repeat for 'small' / 'Small'.
6. Press 'Add item' again and repeat for 'large' / 'Large'.
7. Press the 'OK' button.
8. Save the model to validate the result.

9. Create an exclusive gateway after the scooter return and name it 'Requires Repair?'

Go back to our process. Then the next step is to create another Exclusive Gateway after the Return Scooter task. It must be linked with a Sequence Flow.

1. Increase the space between the User Task and the end event.
2. Locate the exclusive gateway in the palette on the left side.
3. Drag and drop the exclusive gateway from the palette onto the sequence flow.
4. Let it go once the sequence flow is green.
5. Save the model to validate the result.

10. Create a merging exclusive gateway after the 'Requires Repair?' and don't give it a name

The next step is to create another Exclusive Gateway after the 'Requires Repair?' gateway. It must be linked with a Sequence Flow. The purpose is to have a merging gateway after the conditions.

1. Increase the space between the last gateway and the end event.
2. Locate the exclusive gateway in the palette on the left side.
3. Drag and drop the exclusive gateway from the palette onto the sequence flow.
4. Let it go once the sequence flow is green.
5. Save the model to validate the result.

11. Change the sequence flow between the gateways to the default flow and label it 'No'

Let's use the sequence flow between the two gateways as a default Flowable in case of no repairs are required. Therefore, we are going to mark it as a default flow and add the label 'No'.

1. Locate the new sequence flow between the two gateways.
2. Click on the sequence flow to select it.
3. Search the properties on the right side for 'Default flow' and check it.
4. Change the name of the Sequence Flow to 'No'.
5. Save the model to validate the result.

12. Add an alternative path for small repairs

Create an alternative path between the exclusive gateways for small repairs. Name the sequence flow outgoing from the gateway 'Small damages' and create a User Task 'Conduct Small Repair'. The sequence flow following that task should then merge back into the final exclusive gateway. Don't forget to set the condition on the sequence flow after the splitting gateway

1. Select the exclusive splitting gateway.
2. Locate the quick draw right of the gateway.
3. Take the User Task and drag and drop it above the existing sequence flow.
4. Change the name of the User Task to 'Conduct Small Repair'
5. Double click on the newly created sequence flow and give it the name 'Small damages'
6. Locate the attribute 'Condition expression' in the attribute panel
7. Click on 'Condition expression' and then click on the the 'Condition Builder' icon on the top right.
8. Select the variable 'damages'
9. Choose the operator 'Equals'
10. Select the value 'Small' from the list.
11. Click on the User Task and locate the quick draw menu.
12. Take the sequence flow and drop it on the merging gateway.
13. Press save to validate the result.

13. Add an alternative path for large repairs

Create an alternative path between the exclusive gateways for large repairs. Name the sequence flow outgoing from the gateway 'Large damages' and create a call activity 'Start Repair Process'. The sequence flow following that task should then merge back into the final exclusive gateway. Again, don't forget the condition.

1. Select the exclusive splitting gateway.
2. Locate the quick draw right of the gateway.

3. Select the 'Task' symbol on the top and select 'Call activity'.
4. Double click on the newly created sequence flow and give it the name 'Large damages'
5. Locate in the properties on the right side the 'Condition expression'
6. Click on 'Condition expression' and then click on the the 'Condition Builder' icon on the top right.
7. Select the variable 'damages'
8. Choose the operator 'Equals'
9. Select the value 'Large' from the list.
10. Click on the call activity and locate the quick draw menu.
11. Take the sequence flow and drop it on the merging gateway.
12. Press save to validate the result.

14. Create the Repair Scooter Process

Now we need to create a new process for the repair. Therefore select the call activity and create a new process with the name 'FLW-P002 Repair Scooter Process' and key 'FLW-P002'. Save all models to validate the result.

1. Click on the Call Activity model.
2. In the property list, search for 'Process reference' which is located in the 'General' section.
3. Enter 'FLW-P002 Repair Scooter Process' as name.
4. Enter 'FLW-P002' as key.
5. Click on 'Create' to create a new process.
6. Save all models to validate the result.

15. Create a User Task 'Write assessment'

As a first step in the new process we should write an assessment. Let's create a User Task with the associated sequence flow.

1. Click on the start event in the FLW-P002.
2. Locate the user symbol in the quick draw next to the start symbol.
3. Drag and drop the user symbol.
4. Rename the User Task to 'Write assessment'.

16. Change the assignee to '${root.initiator}'

Now the default assignee is ${initiator} and we can't use that inside the sub-process since the variable is not set. Let's change that variable to the initiator of our case instance which is root.initiator.

1. Click on the User Task 'Write assessment'.
2. In the properties on the right side, locate the field 'Assignee' which is located in the attribute group 'Assignment'.

3. Change it to '${root.initiator}'
4. Click the save button to save the process.

17. Create a splitting and a merging parallel gateway

Create two parallel gateways directly after the User Task. The first parallel gateway should split the sequence flow while the second one is merging the sequence flow again. Please also make sure that you have an end event after the last gateway.

1. Click on the User Task and locate the quick draw.
2. Take the end event and drag and drop it all the way to the right that you have some space in the middle.
3. Locate the parallel gateway in the palette on the left hand side.
4. Drag and drop two parallel gateways to the sequence flow.
5. Save the model to validate the result.

18. Create a parallel task 'Check Tires'

Between the parallel gateways there should be a task 'Check Tires' which is assigned to the case initiator.

1. Ensure you have enough space between the two parallel gateways.
2. Drag and drop a User Task on the sequence flow between the gateways.
3. Rename the User Task to 'Check Tires'.
4. Change the assignee to '${root.initiator}'
5. Save the model to validate.

19. Create a parallel task 'Check Electronics'

Between the parallel gateways there should be a task 'Check Electronics' which is assigned to the case initiator.

1. Move the existing User Task to the top.
2. Create a new sequence flow between the parallel gateways
3. Drag and drop a User Task on the sequence flow between the gateways.
4. Rename the User Task to 'Check Electronics'.
5. Change the assignee to '${root.initiator}'
6. Save the model to validate.

20. Create a parallel task 'Check Frame'

Between the parallel gateways there should be a task 'Check Frame' which is assigned to the case initiator.

1. Create a new sequence flow between the parallel gateways
2. Drag the sequence flow below the task.
3. Drag and drop a User Task on the sequence flow between the gateways.
4. Rename the User Task to 'Check Frame'.

5. Change the assignee to '${root.initiator}'
6. Save the model to validate.

21. Publish the changes

To publish your app, click on the cloud icon featuring a small arrow to the top.

1. Locate the top menu bar which above the canvas.
2. Locate the publish button in the toolbar. This is the cloud symbol with a little arrow inside.
3. Press this button and a popup window will appear.
4. In the popup window, confirm the deployment target and the app. In this environment you do not really have a choice and you can simply say 'Publish'.
5. Once you completed this exercise you will be redirected automatically to Flowable Work. In a real application you would now need to switch to 'Flowable Work' manually.

22. Create a scooter and rent it twice

In Flowable Work, create a new case and rent the scooter twice. For the first time select small repair, for the second time select large repair. Once you have done that, decommission the scooter to complete the exercise.

1. Select 'New' below 'Work' or 'Work' below 'New'
2. Select the case 'FLW-C001 Scooter'.
3. In the Start Form, enter any data you want.
4. Press 'Start' and run through the tasks.
5. Once ready, press the rent scooter button.
6. Ensure that for the return you select small damages
7. Complete the rent scooter process.
8. Once done, execute the rent scooter one more time.
9. Ensure that for the return you select large damages
10. Complete the rent scooter process.
11. Once you then decommission your scooter, the exercise will be done.