There are many ways to store data in Orchestrator to smooth the way for your automation. Assets provide you with a simple way of centralizing and managing variable values.

Let’s take the project we created for the previous tutorial (Processes and Jobs). We want to type a specific message into a Notepad window. The message is not different for each job started so it doesn’t make much sense to add it as input parameter, but it can be subject to change, so it’s better to make sure it is easily editable.

In this case, we can use an asset to store the message we want to type into notepad.

Let’s go to Assets tab in Orchestrator and create a new asset.

Give it a name.. (NotepadMessage)

We can see that it can be of following types:

* Text
* Bool
* Integer
* Credential

Since we want to store string, let’s select the type “Text".

And add the value we want to store, let’s say “Hello! This message is stored in an asset!"

Now that we have the asset defined in Orchestrator, let’s modify our project so it uses the asset we just created.

We’ll need a “Get asset" activity .. Specify the name of the asset – this is the one we wrote in Orchestrator - and the variable where we want to store the value retrieved from Orchestrator (I’ll create a new variable and name it “NotepadMessage".

Now we only need to update the ‘type into’ activity in order to use the value of the variable we created.

And.. Let's run it!

Let’s assume now that we need the value of the asset to be different depending on the robot. In this case, let’s create another asset.. And use VALUE PER ROBOT.

Now we can select each robot.. And add a value for the asset.

For robot1 let’s write “Hello, Robot 1" and for robot 2 - “Hello, Robot2!".

Now let’s publish the project..

..update the process

And start a job on both robots.

Let’s check the results.. We can notice that on robot2 the message is “Hello, Robot2" and for Robot1 it’s “Hello, Robot1".

---

Another important type of asset is the Credential!

Let’s say we have the UiDemo application and we need to create a workflow for login. In this case, we can store the username and password into a credential asset.

Let’s create it: give an appropriate name, select the Credential type and write the username and password (user: admin, pass: password).

Note: After we save the credential, if we try to edit it, the password will not be visible anymore. Also, if the username is modified, the password must be typed again.

Now that we have our credential available in Orchestrator, let’s build the workflow.

We should create a new workflow and:

-Add a Get Credential Activity – create the variables for the username and password. Notice that the password is a SecureString.

-Add a Type into activity - indicate the username field and add Username variable as input for this activity.

(in the video, don’t forget to rename the activity. The part where the name is modified can be skipped in the video or very fast-forward but the activities should appear renamed. Another option would be to have the workflow partially built and only add the variables as input for the type activities)

-Add a Type secure activity - indicate the password field and add Password as input (notice that this activity only accepts SecureString variables)

-Click LOGIN (Indicate Login button)

And test the workflow