How to pilot Calculator

Project name: Calculator

Package version: 1.1.0

THE BEST RUN SAP

Version of document	Date	Description
1	October 14, 2020	Document created
1.1	March 05, 2021	Change template Add information about SDK dependencies
1.2	August 12, 2021	Changing version of the package
1.3	December 17, 2021	Add automation to present synchronous clicks

TABLE OF CONTENTS

INTRODUCTION

This document describes the SAP Build Process Automation sample **Calculator** and provides the following information:

- Description (functional and technical)
- Version used to generate this sample

It also contains information on prerequisites, such as the steps to follow before launching the agent.

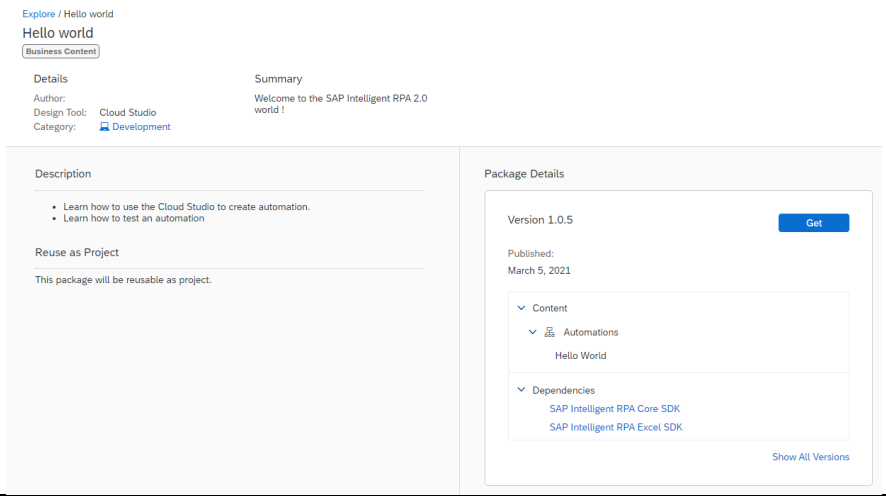
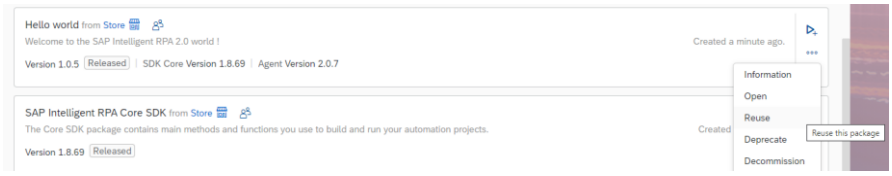
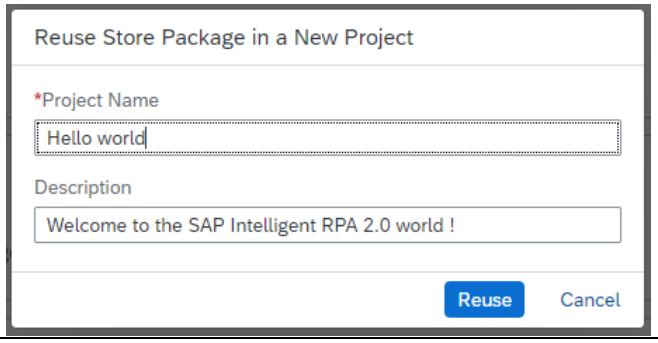
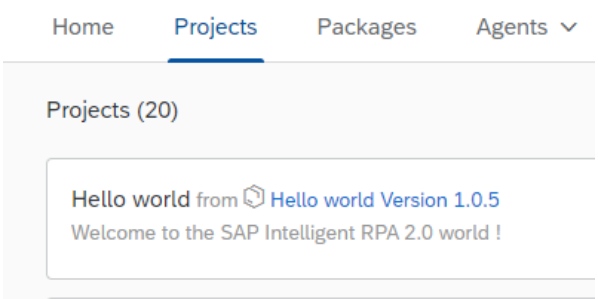
IMPORTANT RECOMMANDATION

General

To use this sample, you need to have a basic knowledge and understanding of SAP Build Process Automation tool. At the very least you need to know how to build an automation, add and modify activities and generate a package.

Reuse the sample as a new project

Note: screenshot might display a different name than the one of this sample.

From the Cloud Factory, open the Store tab and select the sample you want to retrieve. Click on the **Get** button.	
Once the package is retrieved, open the Packages tab of the Cloud Factory. Click on the Options button of the package you just retrieved and select the option **Reuse**.	
Set a name for the project to be created.	
Open the project that has just been created.	
If needed, update the content of this project, and generate a new package from it.	

You need to execute this procedure to be able to open the project and see all its content (the captured applications, the declared items, the automations, etc.).

DESCRIPTION

This package contains captures, datatype and automations that are described below. See chapter Version for more details about the version of the Desktop Agent and the SDK dependencies.

Settings

This section describes the settings of the project such as environment variables or dependent packages that are used in the automation.

Environment variables
N/A

Dependent packages
N/A

Captures

This section describes the captures which were made to pilot the application in this sample. It will also describe the different methods which were used to capture the pages and declare the items.

Calculator
Type: UI Automation
Framework: None

Screen	Description
Calculator	All buttons of the calculator were simply declared using the AutomationId as criterion. The *NormalOuput* item uses the **Must Exist** option. The *num6Button* item uses the Must Exist option. As the UI Automation connector is in use, we recommend checking that the option **Refresh screen recognition** is set to **At a fixed time interval**. Note: This value means that the agent will periodically check if the screen can be identified or not (this mechanism is also known as *polling*). It reduces a bit the performances of the bot but it makes sure that elements (screens or items) will be identified even if they are loaded dynamically.

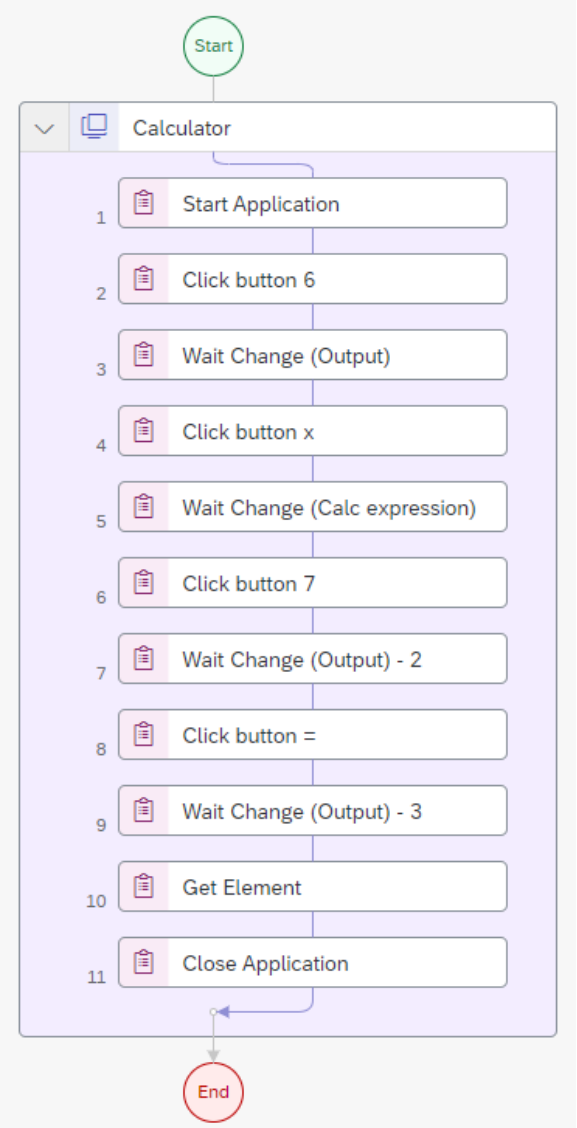
Datatypes

This section describes the datatype used in this sample. It describes the structure of the datatype and where it is used in the automations.

N/A

Automations

Calculate
Type: Attended
Input: None
Output: None

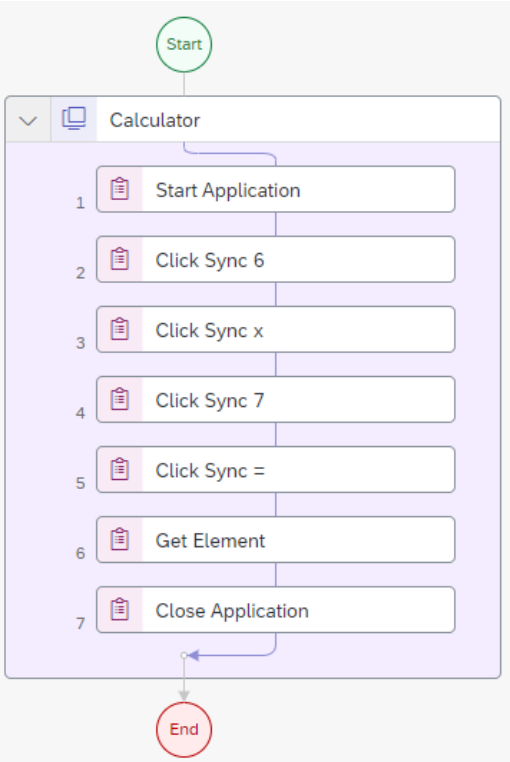


The agent starts the application then performs a simple calculation and displays the result in the console log.

Important:
The CLICK action is **asynchronous**. Consequently, it can be very difficult to pilot an application as you must be sure that the application is ready to perform the next actions. Otherwise, you will *lose* some actions in your automation. To detect if an application is ready to perform the next actions, you might want to detect a change in the application (for example a given value changes when you perform an action).
In this case, we look for a change on the *NormalOutput* item when we click on a number or the *equal* button. When we click on an operator, we look for a change on the *CalculatorExpression* item.

When it's not possible to use this technique, you need to insert a **Wait** instruction in the automation. Beware: it might be difficult to pilot an application using **Wait** activities (especially if the system is low to respond, then the defined delay might not be enough).

Calculate using synchronous clicks
Type: Attended
Input: None
Output: None



The agent starts the application then performs a simple calculation and displays the result in the console log.

Important:
The CLICK action is **synchronous** in this automation so there is no need to check if changes occurred in the application.

VERSION

The product versions used to generate this sample are detailed below. This sample is provided “as is”, with no warranty that it will work correctly with other versions. If some versions of your software are different (such as the tool version or the target application version), you may need to recapture the application and/or update the workflow activities.

SAP Build Process Automation

This sample targets the Desktop Agent **2.0.7** or higher.

The following SDK dependencies were used to generate this sample: 1.19.55

See documentation for more details about the compatibility between SDK version and Desktop Agent.

Target application

calc.exe	10.2008.2.0

PREREQUISITES

Global setup

SAP Build Process Automation must be installed in accordance with the installation guide available here. An SAP Build Process Automation Factory must be available with a suitable environment (containing an agent). All information can be found in the “Getting Started” section accessible via the above link.

Specific steps to follow before launching the agent

N/A

EXPECTED OUTPUT

At the end of the execution, the following output should be the result of the operation.

www.sap.com/contactsap

THE BEST RUN SAP®