



How to manage errors

Project name: Error management

Package Version: 1.0.1

Version of document	Date	Description
0	20 June 2022	Document created

TABLE OF CONTENTS

INTRODUCTION	4
IMPORTANT RECOMMENDATION	5
General	5
Reuse the sample as a new project	5
DESCRIPTION	7
Settings	7
<i>Environment variables</i>	7
<i>Dependent packages</i>	7
Captures	7
<i>Duckduckgo</i>	7
Datatypes	7
Alerts	7
Automations.....	7
<i>Scenario to execute</i>	7
<i>Retry scenario if error occurs</i>	8
<i>Get business data in OnError branch</i>	9
VERSION	11
SAP Build Process Automation	11
Target application	11
PREREQUISITES.....	12
Global setup	12
Specific steps to follow before launching the agent	12
EXPECTED OUTPUT	13

INTRODUCTION

This document describes the SAP Build Process Automation sample **Read Write CSV JSON XML – Cloud Studio** 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.

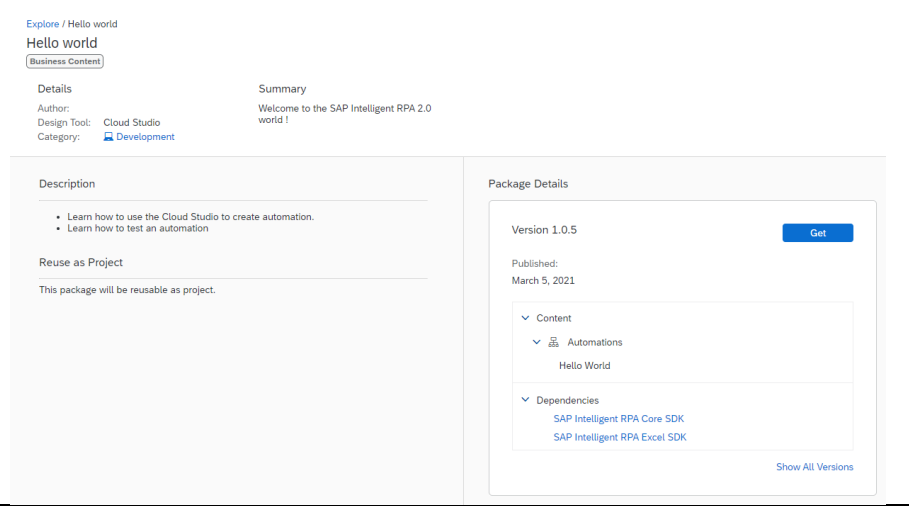
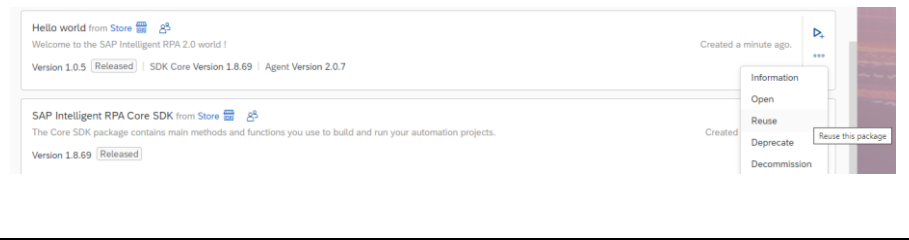
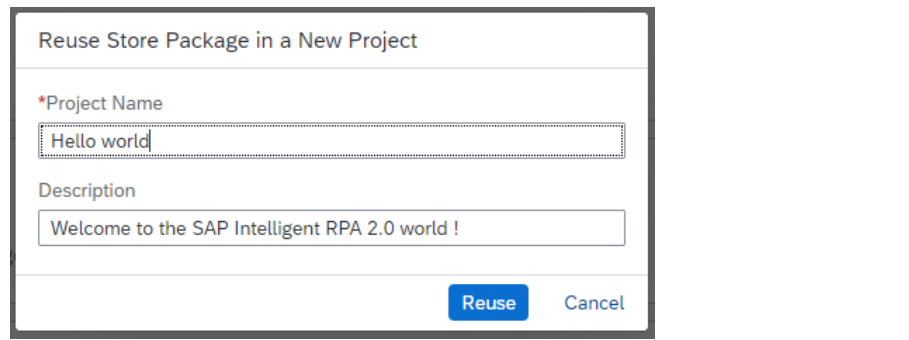
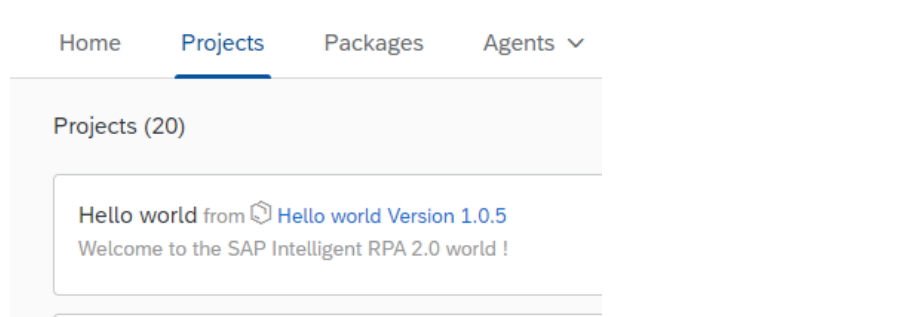
IMPORTANT RECOMMENDATION

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.

<p>From the Cloud Factory, open the Store tab and select the sample you want to retrieve.</p> <p>Click on the Get button.</p>	 <p>Explore / Hello world Hello world (Business Content)</p> <p>Details Author: Cloud Studio Design Tool: Cloud Studio Category: Development</p> <p>Summary Welcome to the SAP Intelligent RPA 2.0 world !</p> <p>Description</p> <ul style="list-style-type: none"> Learn how to use the Cloud Studio to create automation. Learn how to test an automation <p>Reuse as Project This package will be reusable as project.</p> <p>Package Details</p> <p>Version 1.0.5 Get</p> <p>Published: March 5, 2021</p> <p>Content</p> <ul style="list-style-type: none"> Automations <ul style="list-style-type: none"> Hello World <p>Dependencies</p> <ul style="list-style-type: none"> SAP Intelligent RPA Core SDK SAP Intelligent RPA Excel SDK <p>Show All Versions</p>
<p>Once the package is retrieved, open the Packages tab of the Cloud Factory.</p> <p>Click on the Options button of the package you just retrieved and select the option Reuse.</p>	 <p>Hello world from Store</p> <p>Welcome to the SAP Intelligent RPA 2.0 world !</p> <p>Version 1.0.5 (Released) SDK Core Version 1.8.69 Agent Version 2.0.7</p> <p>SAP Intelligent RPA Core SDK from Store</p> <p>The Core SDK package contains main methods and functions you use to build and run your automation projects.</p> <p>Version 1.8.69 (Released)</p> <p>Created a minute ago</p> <p>Information Open Reuse Deprecate Decommission</p> <p>Reuse this package</p>
<p>Set a name for the project to be created.</p>	 <p>Reuse Store Package in a New Project</p> <p>*Project Name Hello world</p> <p>Description Welcome to the SAP Intelligent RPA 2.0 world !</p> <p>Reuse Cancel</p>
<p>Open the project that has just been created.</p>	 <p>Home Projects Packages Agents</p> <p>Projects (20)</p> <p>Hello world from Hello world Version 1.0.5 Welcome to the SAP Intelligent RPA 2.0 world !</p>
<p>If needed, update the content of this project, and generate a new package from it.</p>	

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.

Duckduckgo

Type: WEB

Screen	Description
Main	The <i>searchField</i> item uses the Must Exist option.
Results	The <i>resultContainer</i> item uses the Must Exist option.

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.

Alerts

This section describes the alerts used in this sample. It describes the structure of the alert.

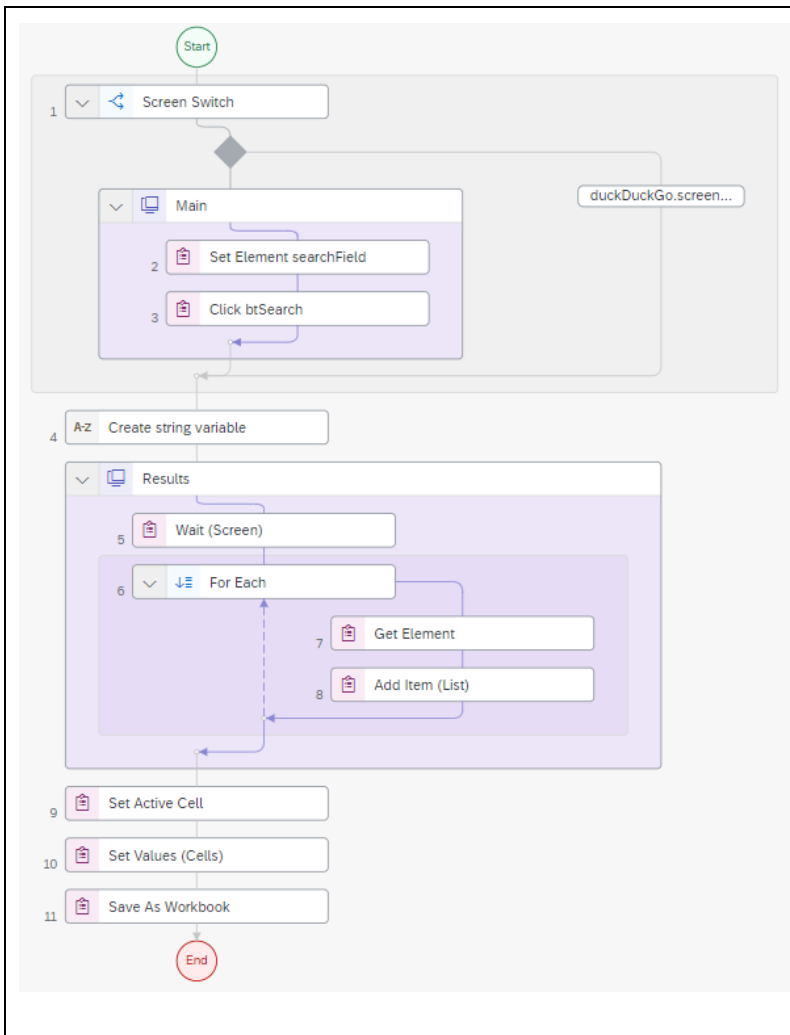
N/A

Automations

Scenario to execute

Input: outputFilePath (*String*)

Output: N/A



This scenario is a sequence of steps executed by the bot. This automation contains multiple errors so the bot will fail accordingly:

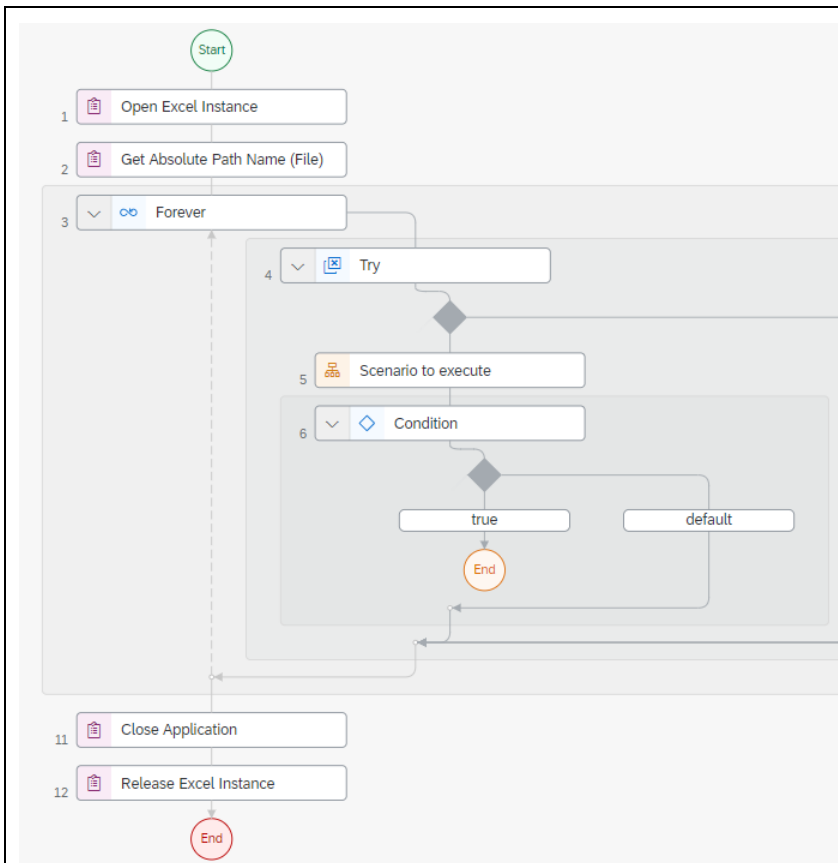
- The web application is not launched when the bot tries to set a value in a field
- The bot set the active cell in Excel but the workbook does not exist.

The OnError branch does not contain any activity. When an exception is launched, the error will automatically be thrown to the parent automation (if any) and this job will fail.

Retry scenario if error occurs

Input: outputPath (String)

Output: N/A



This automation (which can be triggered) is the one calling the *Scenario to execute*.

First it opens the Excel instance, and then tries to execute the scenario. As planned, exceptions will be thrown as some conditions are not met.

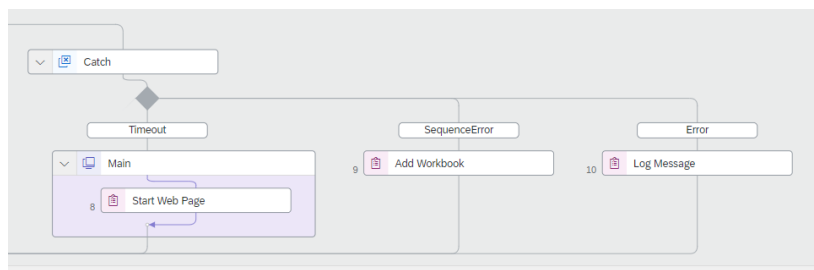
On a Timeout error (cause by the bot waiting too long to detect the web application), we make sure to start the web page.

When a SequenceError is raised (due to an incorrect sequence in the use of the Excel activities), we open a new workbook.

It is only when the sub-scenario has correctly been executed that the bot can end the automation.

Note: in real use cases, it might be interesting to get error details, to determine what went wrong and fix it accordingly.

The catch branch of the try/catch block is defined as below:

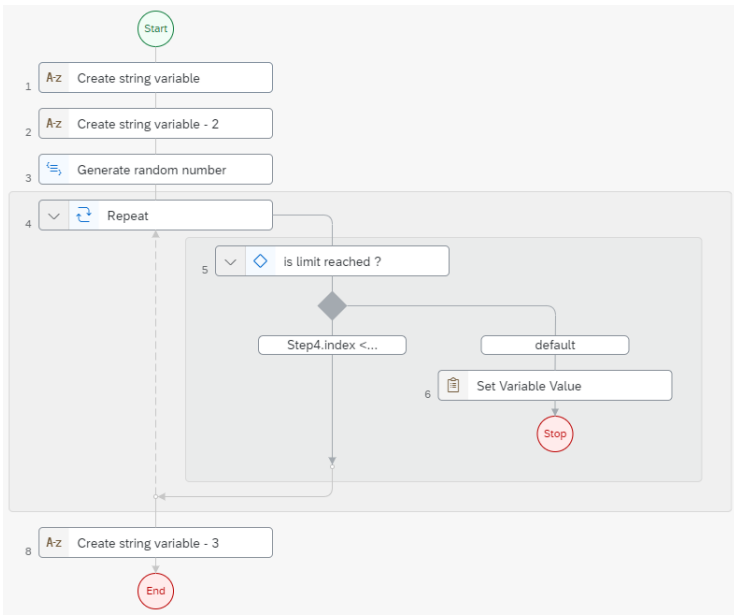


Get business data in OnError branch

Type: Unattended

Input: N/A

Output: N/A



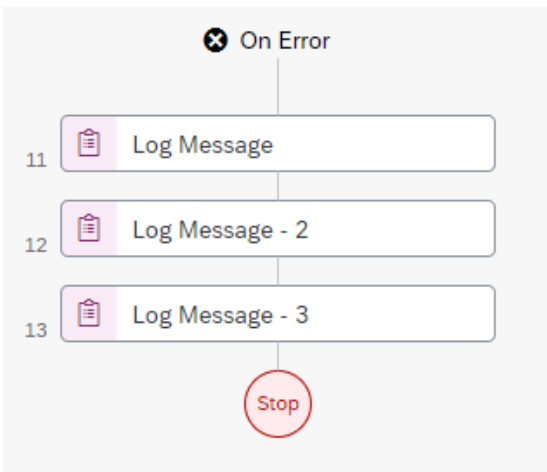
In this automation, a random number is generated (between 1 and 10).

In the loop, if the limit is reached then an exception is thrown.

In the OnError branch, it is possible to retrieve the value of the different variables used in the automation.

The only condition to read the value of a variable is that this variable must have been set. If an uncaught error was thrown before a variable was initialized, then its value is *null*.

In the **OnError** branch:



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 **2.0.23** or higher.

The following SDK dependencies were used to generate this sample: 1.26.51

See [documentation](#) for more details about the compatibility between SDK version and Desktop Agent.

Target application

Application	Version
duckduckgo.com	N/A Date of last test: June 20, 2022

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

www.sap.com/contactsap

© 2019 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.

THE BEST RUN

