



## **Outlook best practices (calendar)**

Project name: Outlook (mail) Best practices - Cloud Studio

Package version: 1.0.1

Version	Date	Description
1	November 22, 2021	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
Datatypes .....	7
<i>CalendarItem</i> .....	7
Automations.....	7
<i>Create and Send Calendar Item</i> .....	7
<i>Delete old Calendars</i> .....	8
<i>Respond to a calendar item</i> .....	9
<i>Save attachments</i> .....	9
<i>Search Calendar Items</i> .....	10
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 **Outlook (mail) Best practices - 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.

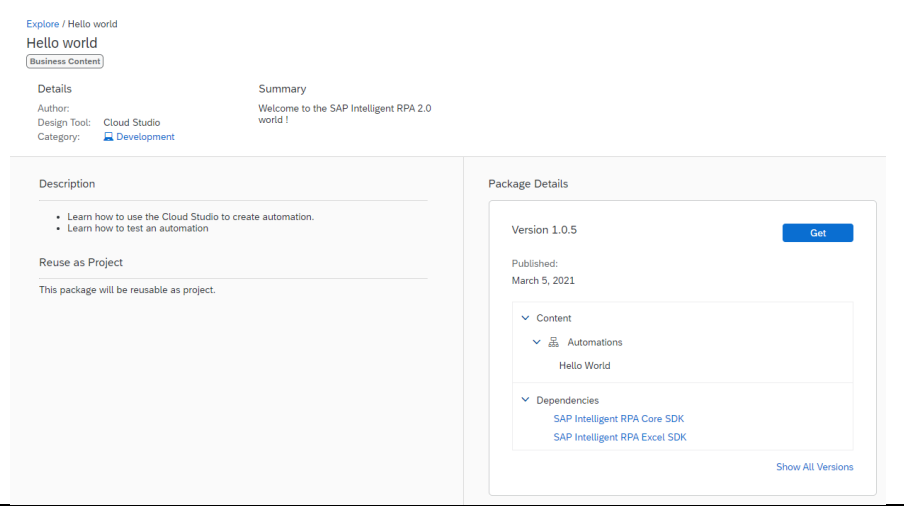
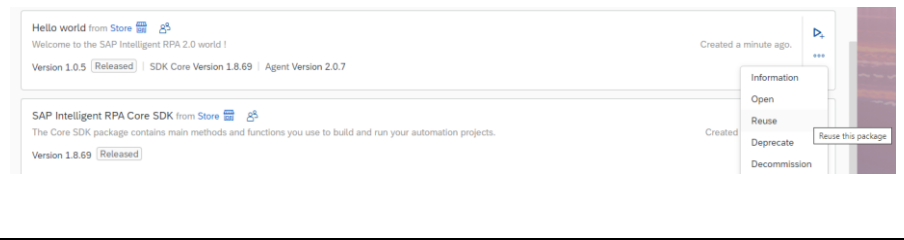
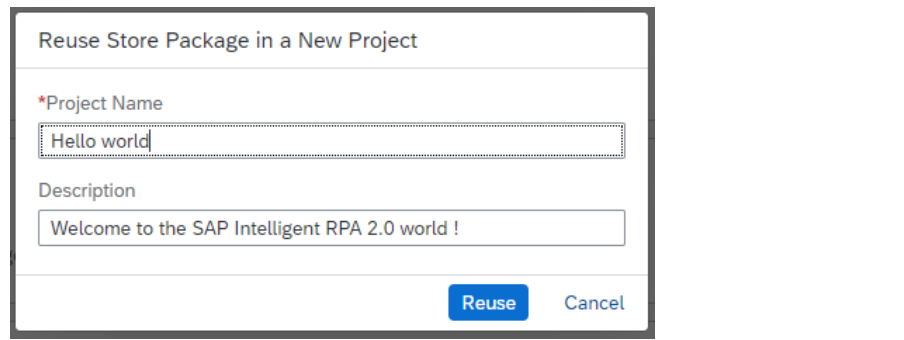
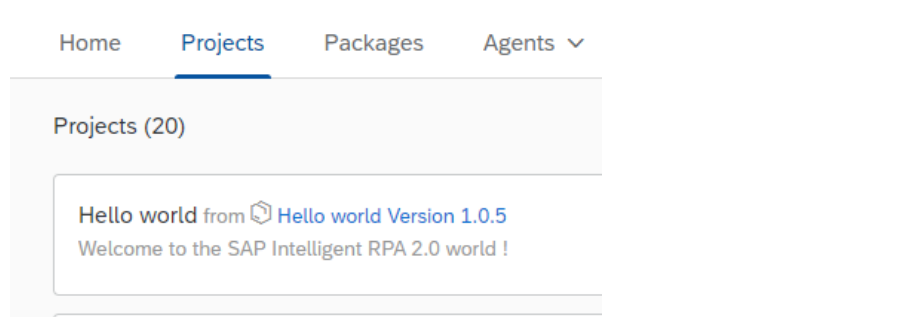
# IMPORTANT RECOMMENDATION

## 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 <b>Get</b> button.</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 <b>Reuse</b>.</p>	
<p>Set a name for the project to be created.</p>	
<p>Open the project that has just been created.</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*

Name	Description	Type
DownloadFolderPath	Location of the folder where attachments should be downloaded	String
EmailAddress	Used when multiple accounts are connected to Outlook	String
Forward	Recipient for whom the calendar item is forwarded to	String
messageContentPath	Location of the file used to generate the content of the mail	String

#### *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.

N/A

### 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.

#### *CalendarItem*

Name of attribute	Type	Description
Recipients	String	Addresses of the recipient of the mail to be sent. If multiple recipients, use semicolon ; to separate them
Name	String	Name of the recipient(s)
Subject	String	Subject of the calendar item
Start	Date/Time	Date of the calendar item
Duration	Number	Duration in minutes of the calendar item

### Automations

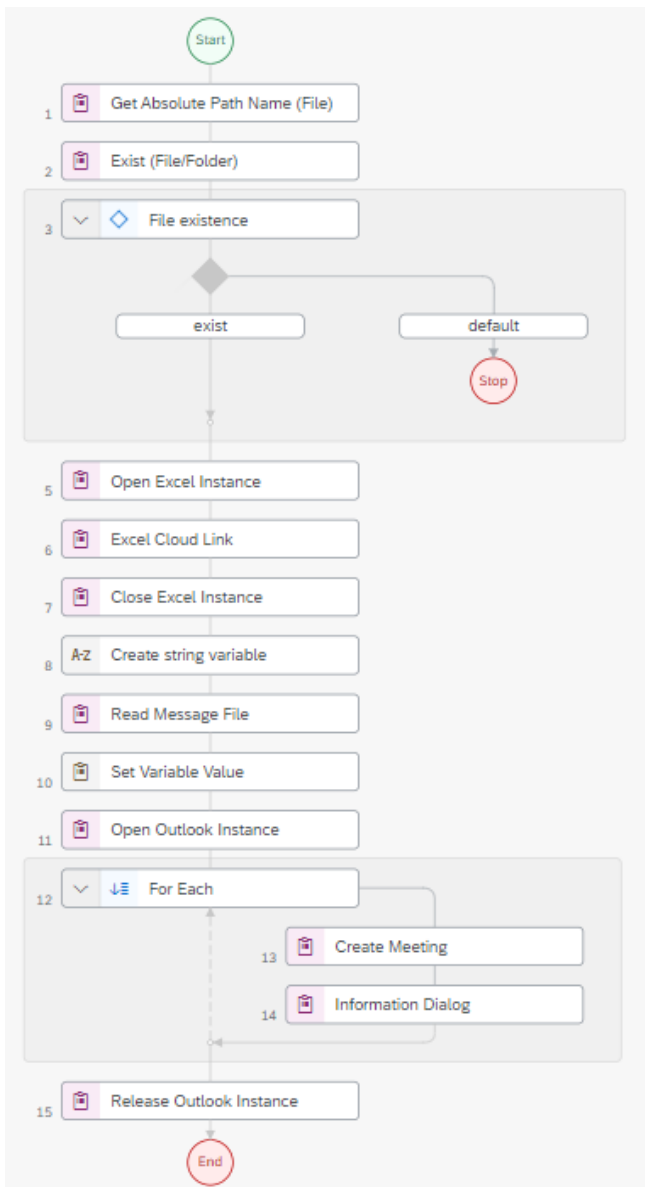
**Note:** all the automations contain Try/Catch blocks. For readability reasons they were not included in the screenshots of the automations in this document.

#### *Create and Send Calendar Item*

**Type:** Attended

**Input:** Excel file path

**Output:** None



The Excel file (input parameter) is opened. Agent will create a list of Calendar Item to create and send each one of them to the correct recipient.

The body of the meeting request is written in a .txt file used as a template.

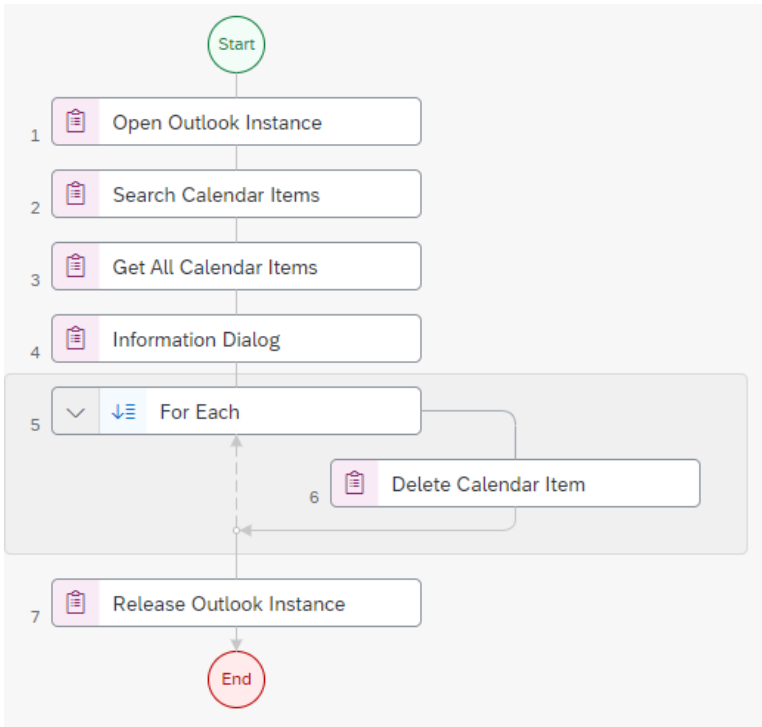
### **Delete old Calendars**

**Type:** Attended

**Input:** excelFilePath is the path of the Excel file used to send mails to multiple recipients

**Output:** None





Agent will look for calendar items, using criteria which are set in the Search Calendar Items activity.

Then it will delete each one of them.

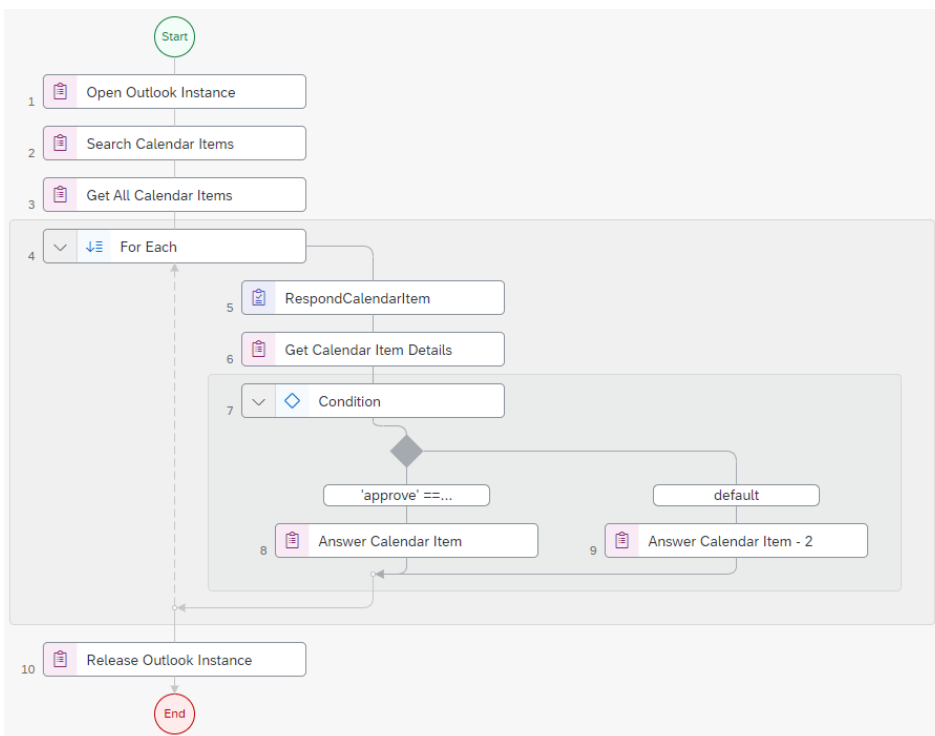
**Important note** : You need to make sure the criteria will not make the bot delete real meetings in your calendar.

### Respond to a calendar item

Type: Attended

Input: None

Output: None



Bot will look for calendar items using the criteria set in the Search Calendar Items activity.

For each item, it will display a user task where the user is asked where to give an answer (approve, decline).

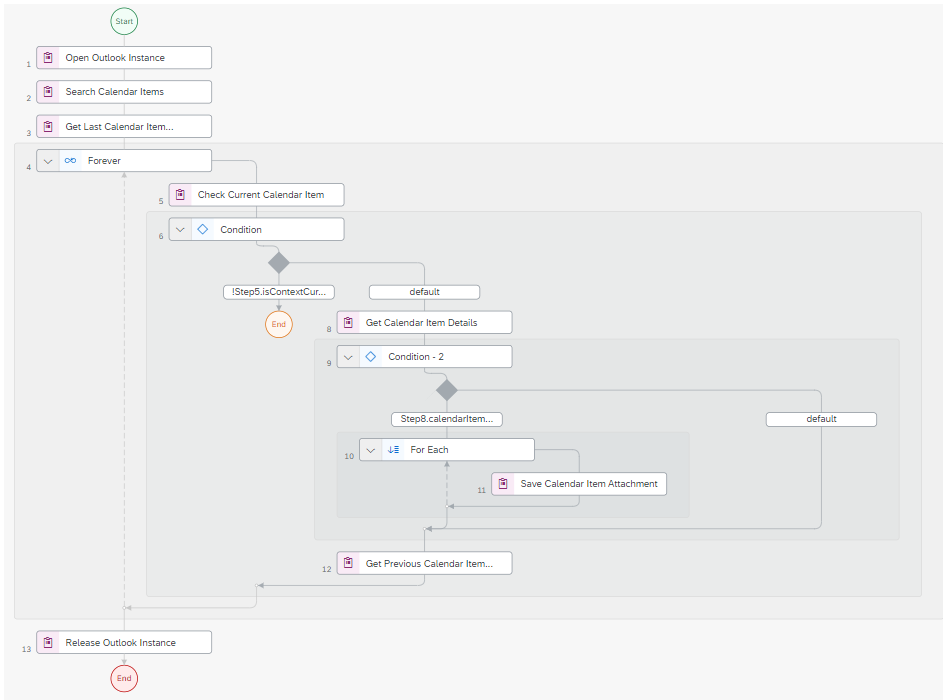
Then the bot will send the answer automatically.

### Save attachments

Type: Attended

Input: None

Output: None



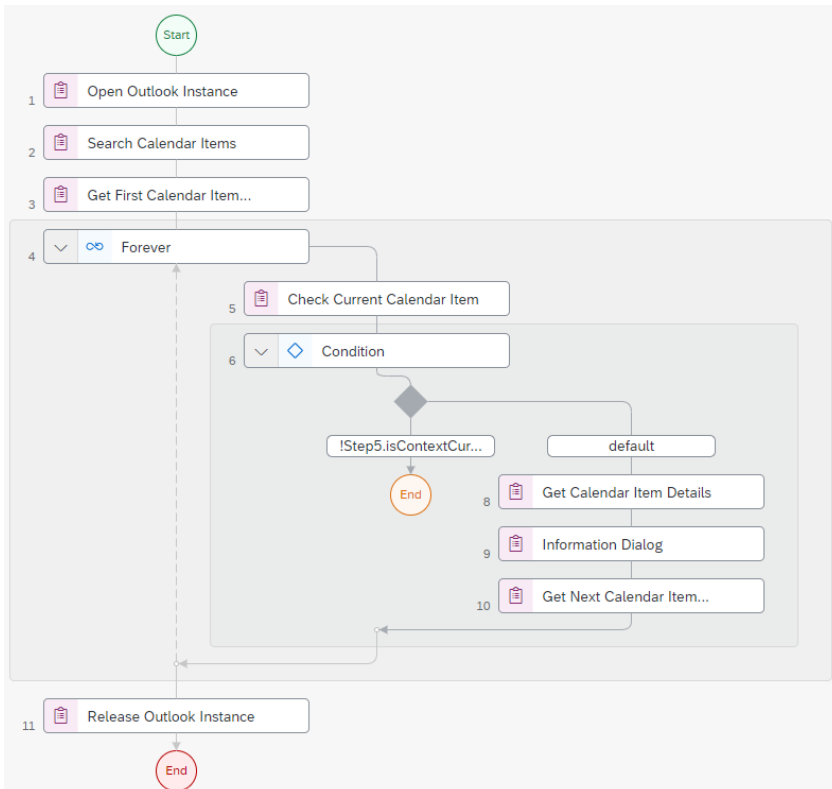
Agent retrieves the list of calendar items. Then it will iterate over this collection to save all the attachments if they exist.

### Search Calendar Items

Type: Attended

Input: None

Output: None



The agent gets a collection of calendar items (actually it is what we call a "context", and not a real list of items).

It will then display information for each elements in the list.

## **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 **2.0.17** or higher.

The following SDK dependencies were used to generate this sample: 1.17.36

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

### **Target application**

N/A

## 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

- Download the **samples.zip** archive from the sample and extract its content
- When you deploy your automation, set the environment variable **ContentMailFilePath** with the location of the text file **mailContent.txt**.
- Set a value to **DownloadFolderPath** variable

**EXPECTED OUTPUT**

N/A

[www.sap.com/contactsap](http://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](http://www.sap.com/copyright) for additional trademark information and notices.

**THE BEST RUN**

