Lab Guide

OCR with IBM RPA

Nigel T. Crowther
ncrowther@uk.ibm.com

Jukka Juselius
jukka.juselius@fi .ibm.com

Hands-on Lab

Version 1.0 for General Availability

NOTICES

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

TRADEMARKS

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a Registered Trade Mark of AXELOS Limited.

ITIL is a Registered Trade Mark of AXELOS Limited.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

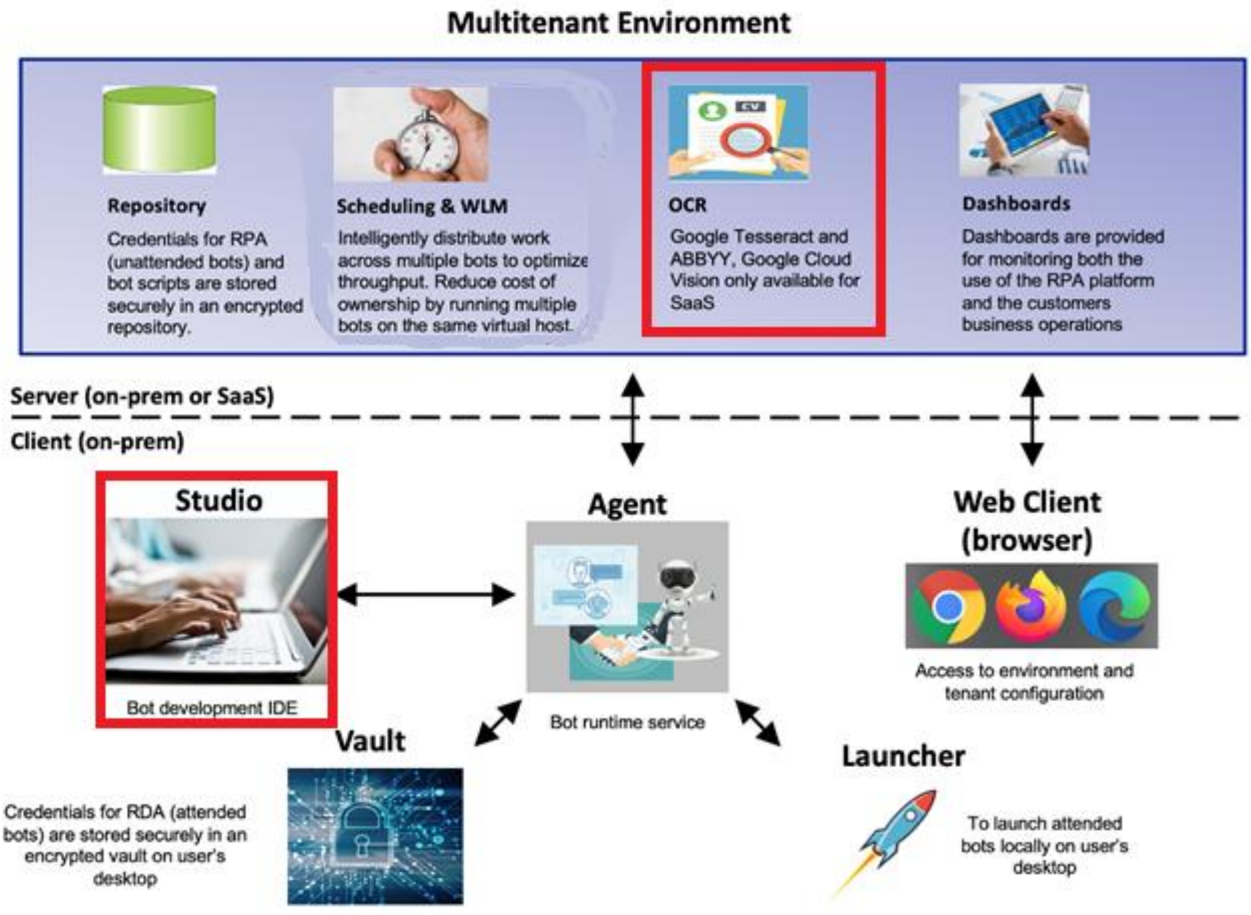
UNIX is a registered trademark of The Open Group in the United States and other countries.

Table of Contents

1 Introduction

This lab we will demonstrate IBM RPA's Imaging capabilities. Imaging solves a wide variety of problems including PDF data extraction and surface automation. We will demonstrate these capabilities through two real-world use cases

Note that for OCR we will be using Google Tessaract. See below.


Multitenant Environment
Repository
Credentials for RPA (unattended bots) and bot scripts are stored securely in an encrypted repository.
Scheduling & WLM
Intelligently distribute work across multiple bots to optimize throughput. Reduce cost of ownership by running multiple bots on the same virtual host.
OCR
Google Tesseract and ABBYY, Google Cloud Vision only available for SaaS
Dashboards
Dashboards are provided for monitoring both the use of the RPA platform and the customers business operations
Server (on-prem or SaaS)
Client (on-prem)
Studio
Bot development IDE
Agent
Bot runtime service
Web Client (browser)
Access to environment and tenant configuration
Vault
Credentials for RDA (attended bots) are stored securely in an encrypted vault on user's desktop
Launcher
To launch attended bots locally on user's desktop

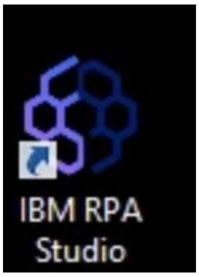
1.1 What is the difference between image processing and OCR?

We deliberately kept OCR out of the title and used *image processing*. Image processing is **the ability to recognize images and perform actions based on the content of these images.** Optical character recognition (OCR) is a subset of image processing. It is defined as **the process of classifying optical patterns in a image and converting these images to text**."

For an overview of OCR within IBM RPA see the following video: https://www.youtube.com/watch?v=_CfB-YtwawI

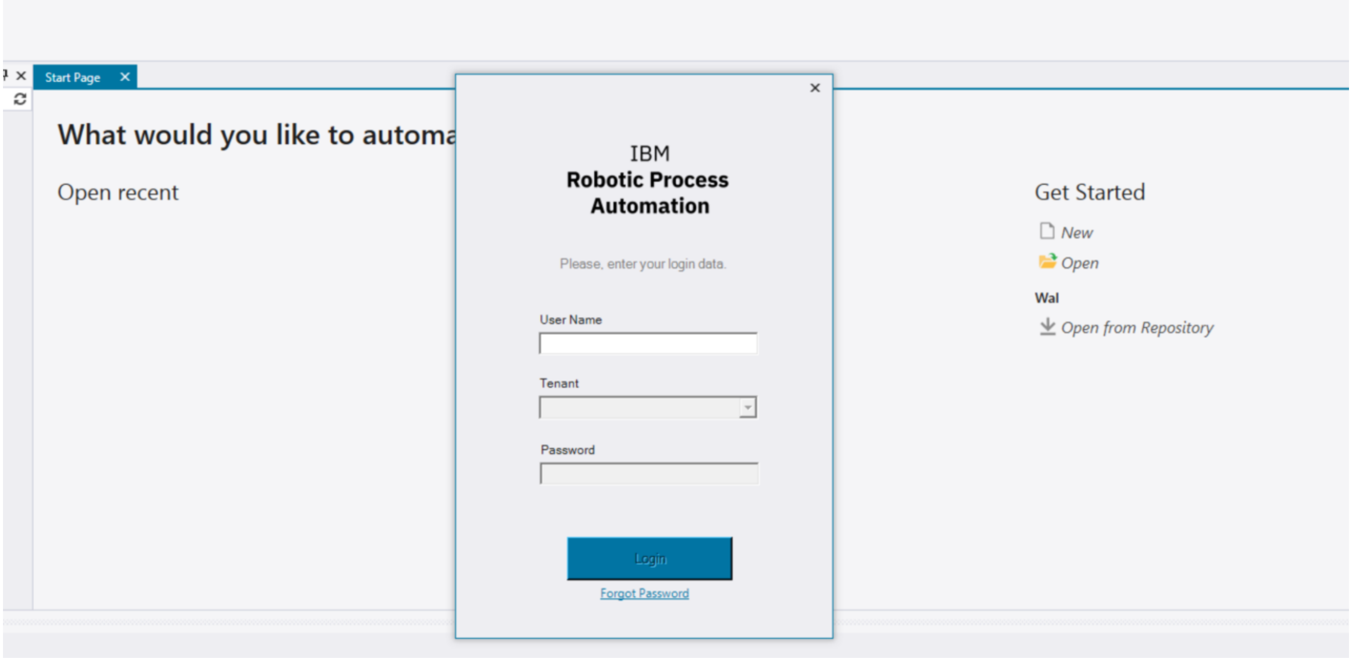
1.2 Getting Started

On your desktop find the IBM RPA Studio icon and launch it.



1.3 Log In

Provide your credentials to log into the studio



Please, enter your login data.

Once the tenant is retrieved, enter your password.

Click login again to finish logging into the client.

2 Scenario 1 – Extracting Data from PDFs

We will begin by creating a bot to read data from PDFs using OCR.

2.1 Scenario Description

Mary is a bookkeeper who spends most of her time copying data from invoices to spreadsheets. You are tasked with automating this.

2.2 Open a work in progress script

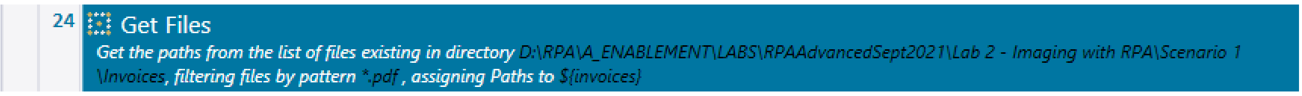
Start IBM RPA Studio and open:

Lab 2 - Imaging with RPA/Scenario 1. Scenario 1 Started.wal.

Your script will open in Studio.

2.3 Correct the folder

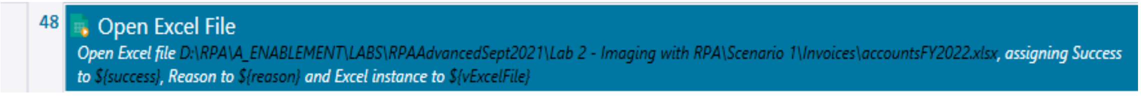
Navigate to subroutine processInvoices. Edit the path to point to the *Lab 2 - Imaging with RPA/Scenario 1/invoices* folder. See below:



Get Files

*Get the paths from the list of files existing in directory D:\RPA\A_ENABLEMENT\LABS\RPAAdvancedSept2021\Lab 2 - Imaging with RPA\Scenario 1\Invoices, filtering files by pattern *.pdf , assigning Paths to ${invoices}*

Navigate to subroutine *openAccountSpreadsheet*. Edit the path to point to *Lab 2 - Imaging with RPA/Scenario 1/invoices/accountsFY2022.xlsx* See below:

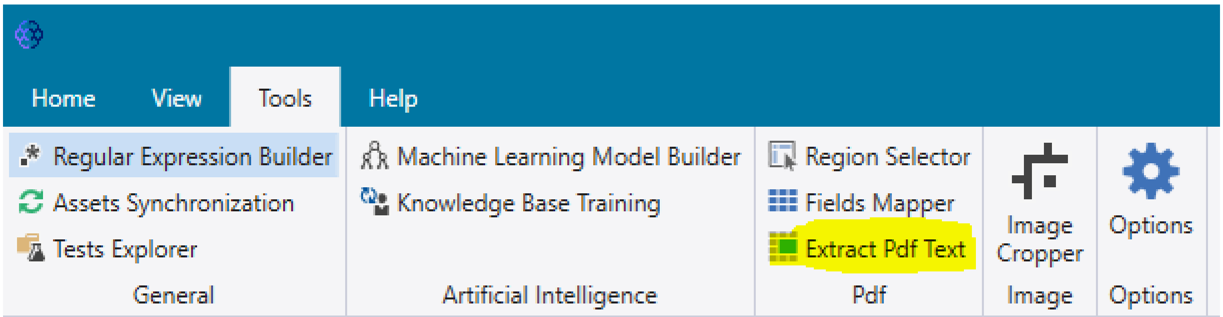


Open Excel File

Open Excel file D:\RPA\A_ENABLEMENT\LABS\RPAAdvancedSept2021\Lab 2 - Imaging with RPA\Scenario 1\Invoices\accountsFY2022.xlsx, assigning Success to ${success}, Reason to ${reason} and Excel instance to ${vExcelFile}

2.3.1 Get PDF Text by OCR Command

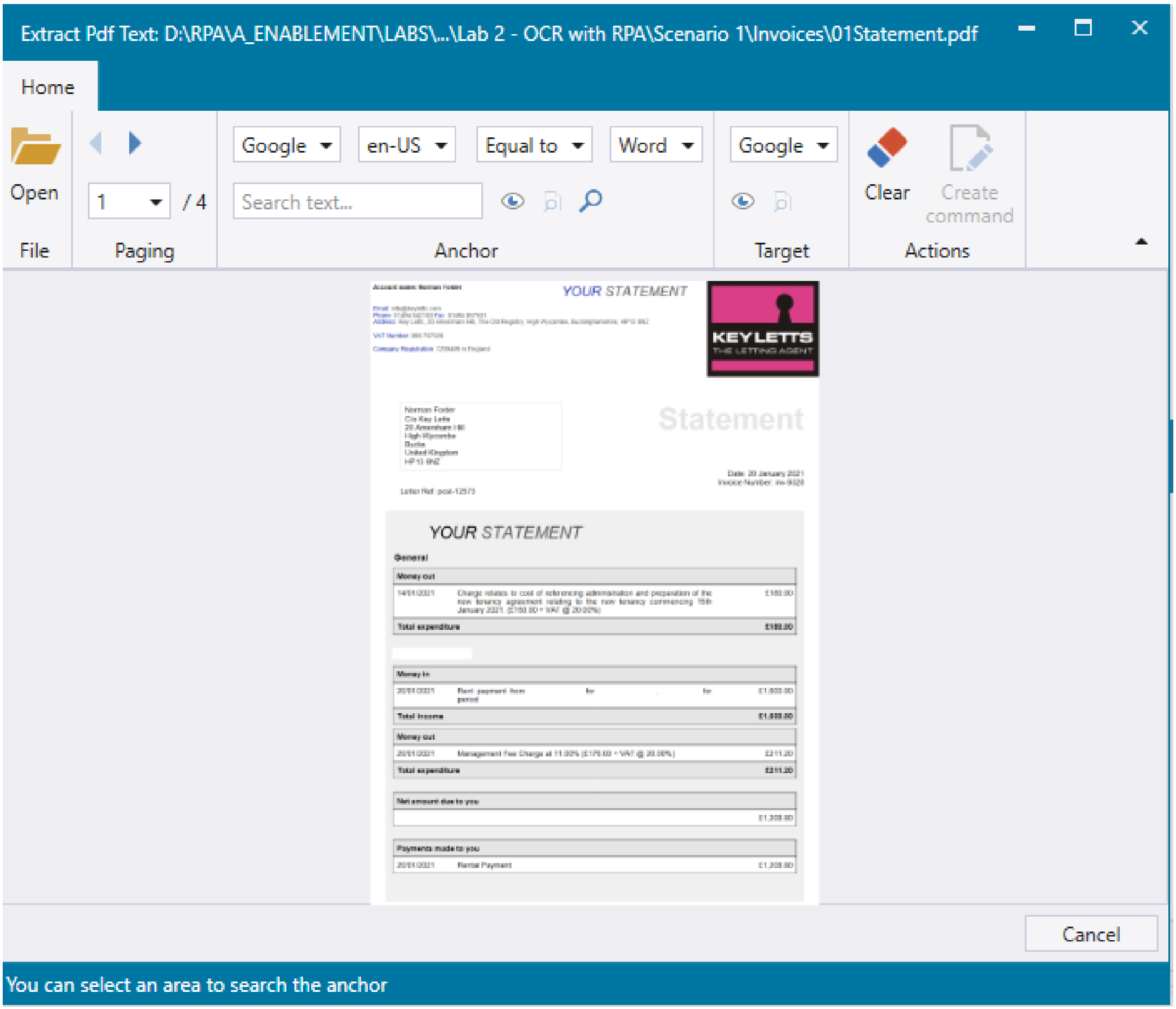
Under the *tools* tab, select *Extract Pdf Text*:



The extract Pdf Text dialog box should appear. Open

Lab 2 - Imaging with RPA\Scenario 1\Invoices\ 01Statement.pdf

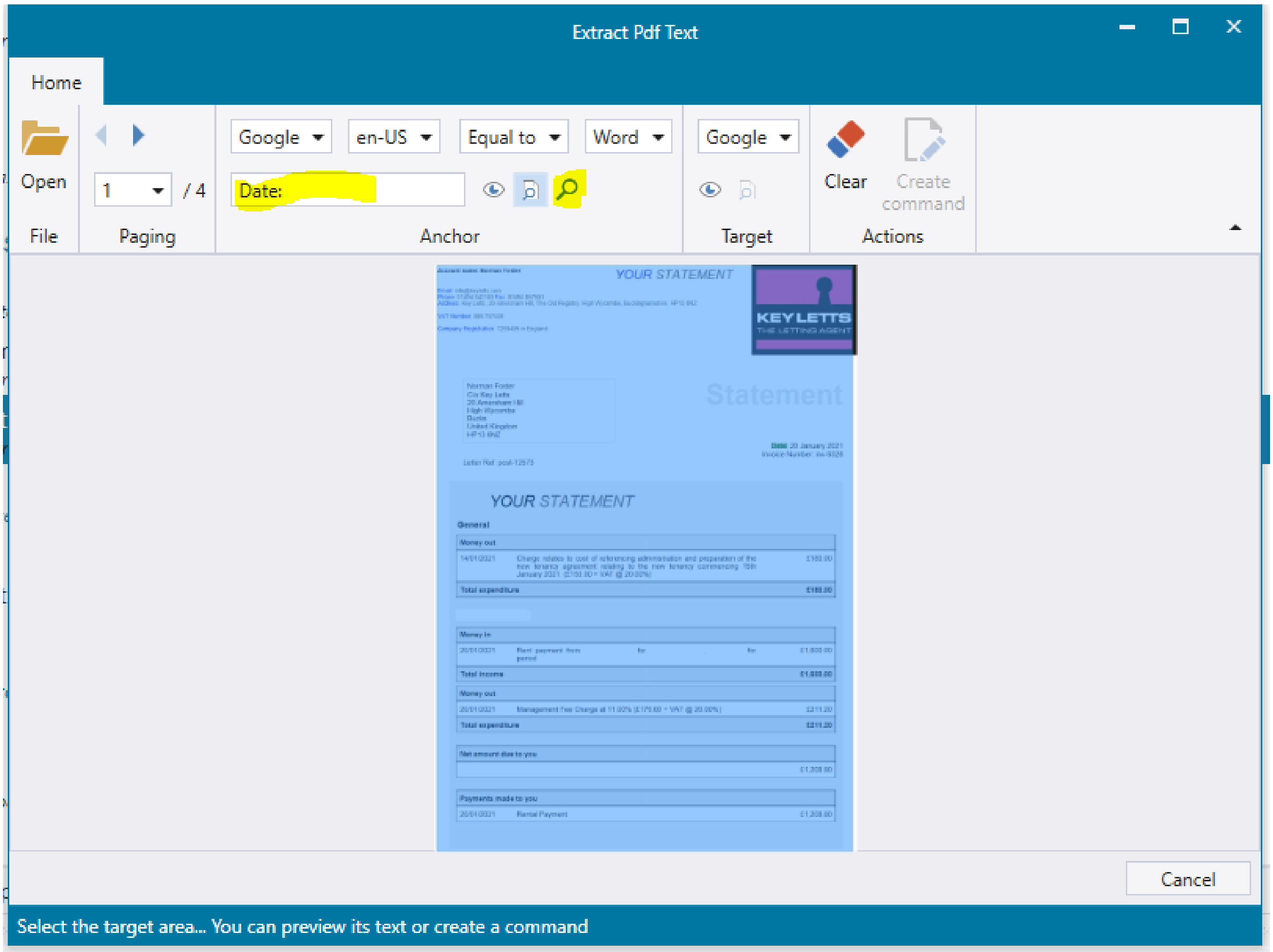
You should see the following:



Extract Pdf Text: D:\RPA\A_ENABLEMENT\LABS\...\Lab 2 - OCR with RPA\Scenario 1\Invoices\01Statement.pdf

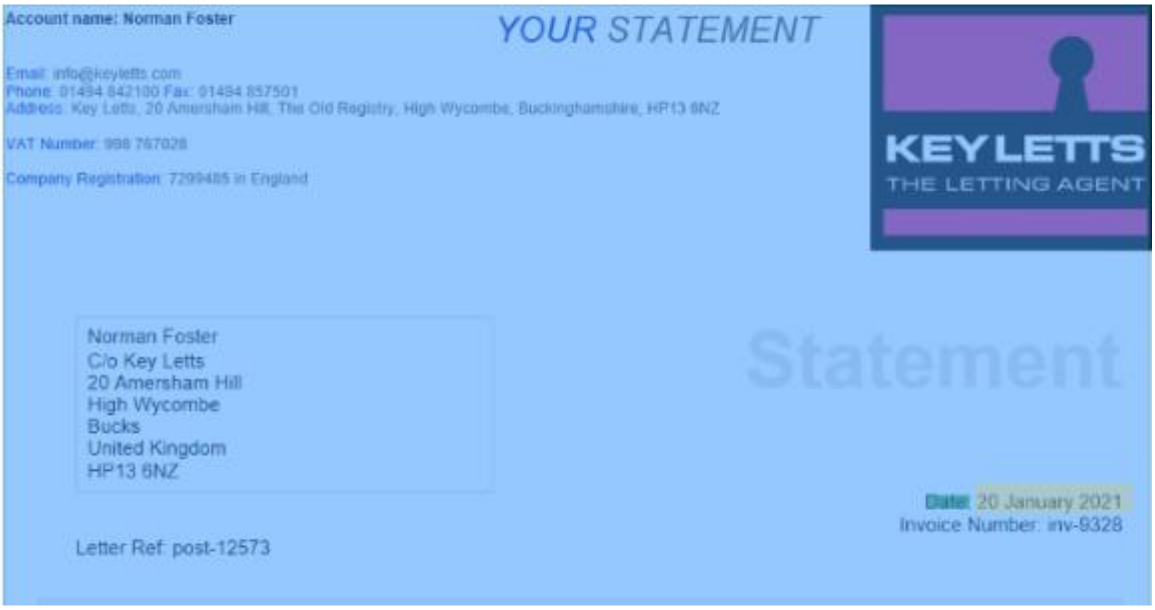
You can select an area to search the anchor

Using your mouse, select the entire page so that it turns blue. If you make a mistake, press the *Clear* button and try again. In the *Search Text* field enter '*Date:*'. This is the keyword starting point from which the OCR will extract a value. Make sure you type is as shown below, including the colon. Press the magnifying glass. This will invoke the OCR reader to find 'Date:' within the selected area. When found it is highlighted in green.



Select the target area... You can preview its text or create a command

Now specify the area containing the actual date. Using your mouse, select *20-January-2021*. You may need to zoom in to do this. This will be highlighted in yellow:



Account name: Norman Foster

YOUR STATEMENT

Email: info@keyletts.com
Phone: 01494 842100 Fax: 01494 857501
Address: Key Letts, 20 Amersham Hill, The Old Registry, High Wycombe, Buckinghamshire, HP13 6NZ

VAT Number: 998 767028

Company Registration: 7299485 in England

KEY LETTS
THE LETTING AGENT

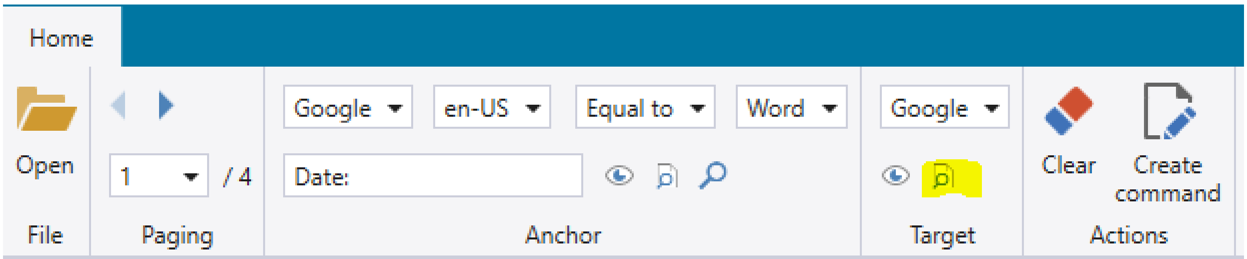
Norman Foster
C/o Key Letts
20 Amersham Hill
High Wycombe
Bucks
United Kingdom
HP13 6NZ

Statement

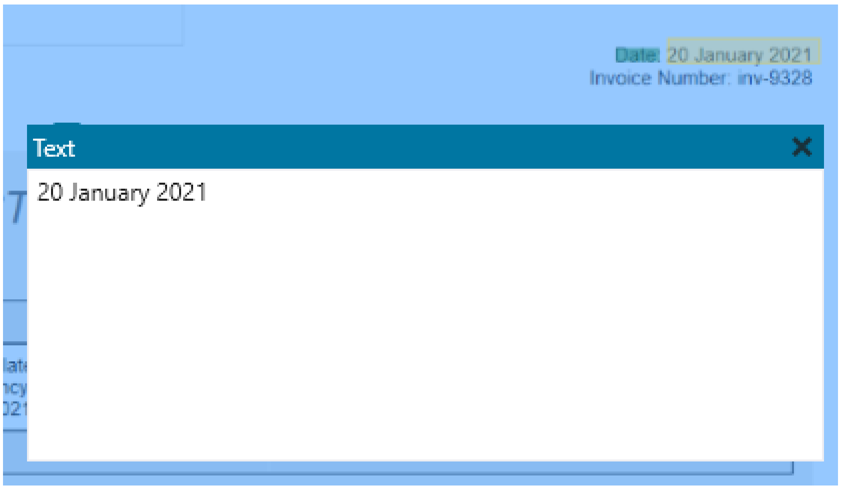
Date: 20 January 2021
Invoice Number: inv-9328

Letter Ref: post-12573

Now press *Preview the text obtained the second selected area...* button:



A dialog box will pop up showing the value extracted by OCR:

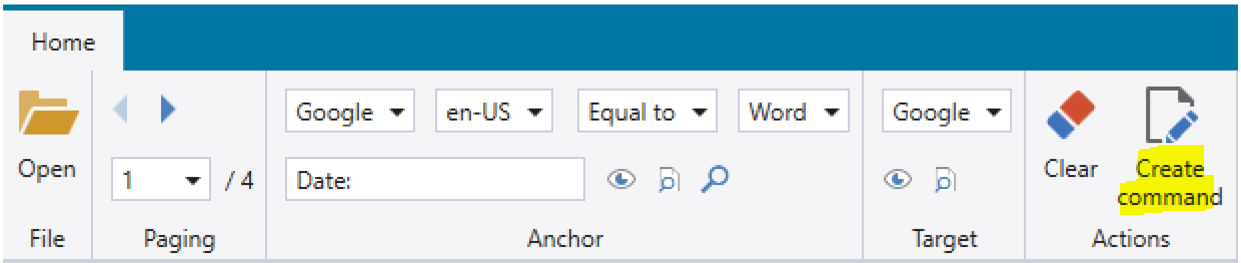

Date: 20 January 2021
Invoice Number: inv-9328

Text

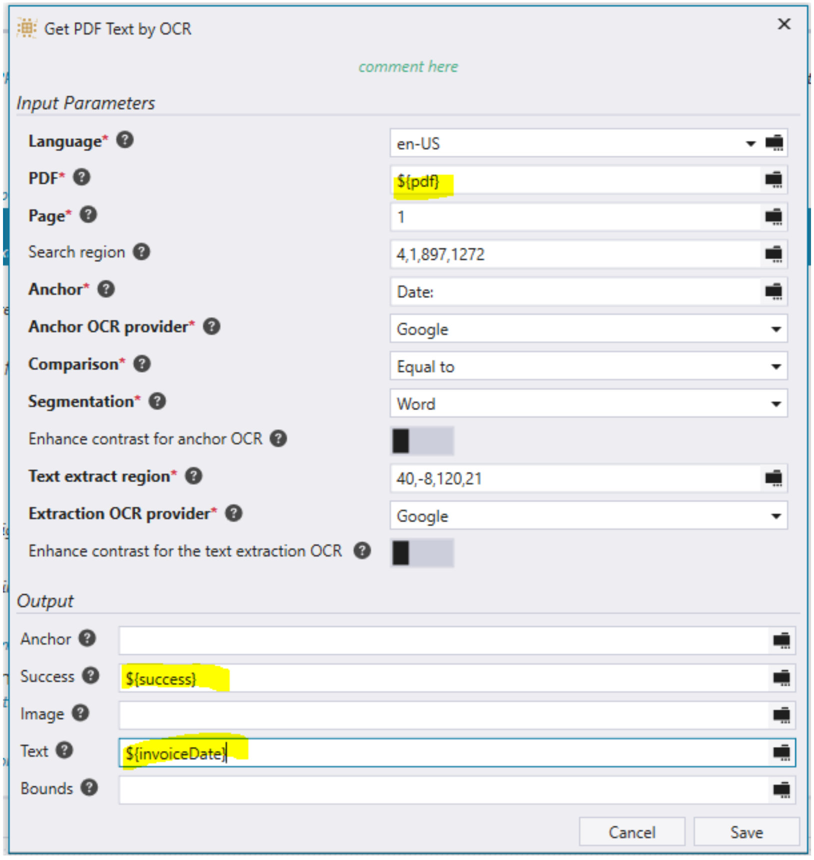
20 January 2021

If you do not see this, then the area has not been highlighted correctly. Re select it.

If you do see the date, press the *Create command* button:



The *Get PDF Text by OCR* command dialog box pops up. Change the PDF text field to point to *pdf* (not *pdf1*). Add the *Success* and *InvoiceDate* variables. The variables have already been defined, so you do not need to create them again. See changes highlighted below:



Get PDF Text by OCR

comment here

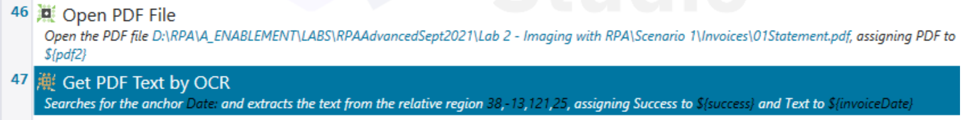
Input Parameters

Language*	en-US
PDF*	${pdf}
Page*	1
Search region	4,1,897,1272
Anchor*	Date:
Anchor OCR provider*	Google
Comparison*	Equal to
Segmentation*	Word
Enhance contrast for anchor OCR	
Text extract region*	40,-8,120,21
Extraction OCR provider*	Google
Enhance contrast for the text extraction OCR	

Output

Anchor	
Success	${success}
Image	
Text	${invoiceDate}
Bounds	

Press *Save*. Close the *Extract Pdf Text* window so that you are back in Studio looking at the script. You should see that two commands have been automatically added to the end of the script:



Open PDF File
Open the PDF file D:\RPA\A_ENABLEMENT\LABS\RPAAdvancedSept2021\Lab 2 - Imaging with RPA\Scenario 1\Invoices\01Statement.pdf, assigning PDF to ${pdf2}

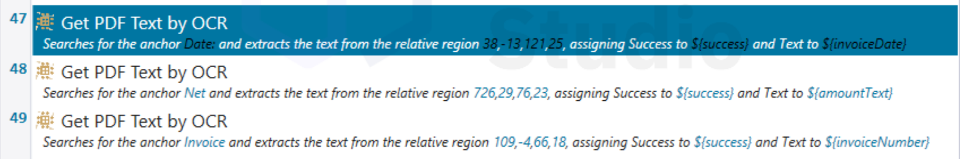
Get PDF Text by OCR
Searches for the anchor Date: and extracts the text from the relative region 38,-13,121,25, assigning Success to ${success} and Text to ${invoiceDate}

Delete the line containing *Open PDF File* as it is not required.

Repeat from the beginning of this section, adding two new OCR commands:

- Keyword: ***Net***, variable: ***amountText***
- Keyword: **Invoice**, *variable:* ***invoiceNumber***

When finished you should see the three *Get PDF Text by OCR* commands as shown below:

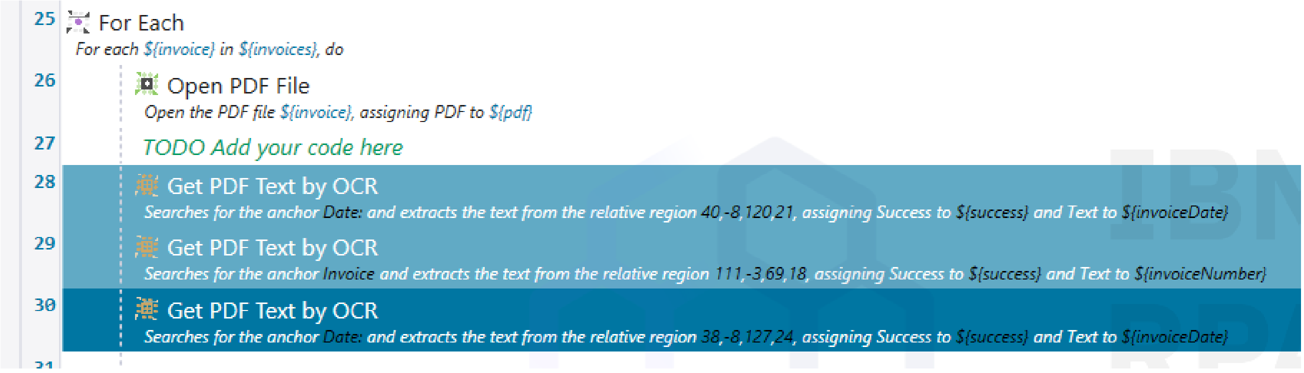


Get PDF Text by OCR
Searches for the anchor Date: and extracts the text from the relative region 38,-13,121,25, assigning Success to ${success} and Text to ${invoiceDate}

Get PDF Text by OCR
Searches for the anchor Net and extracts the text from the relative region 726,29,76,23, assigning Success to ${success} and Text to ${amountText}

Get PDF Text by OCR
Searches for the anchor Invoice and extracts the text from the relative region 109,-4,66,18, assigning Success to ${success} and Text to ${invoiceNumber}

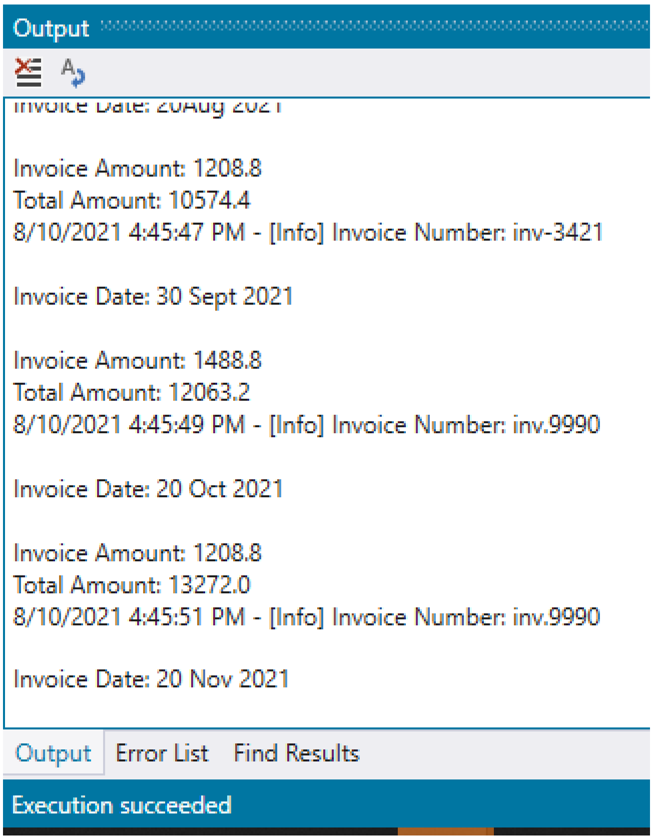
Copy the remaining three lines into subroutine *processInvoices*, directly beneath *Open PDF file* shown below:



```
For Each
For each ${invoice} in ${invoices}, do
    Open PDF File
    Open the PDF file ${invoice}, assigning PDF to ${pdf}
    TODO Add your code here
    Get PDF Text by OCR
    Searches for the anchor Date: and extracts the text from the relative region 40,-8,120,21, assigning Success to ${success} and Text to ${invoiceDate}
    Get PDF Text by OCR
    Searches for the anchor Invoice and extracts the text from the relative region 111,-3,69,18, assigning Success to ${success} and Text to ${invoiceNumber}
    Get PDF Text by OCR
    Searches for the anchor Date: and extracts the text from the relative region 38,-8,127,24, assigning Success to ${success} and Text to ${invoiceDate}
```

2.4 Test

Run the script with Ctrl+F5. In the output log you should see that each PDF has been scanned and its key values stored in the spreadsheet. See below:



```
Invoice Date: 20Aug 2021

Invoice Amount: 1208.8
Total Amount: 10574.4
8/10/2021 4:45:47 PM - [Info] Invoice Number: inv-3421

Invoice Date: 30 Sept 2021

Invoice Amount: 1488.8
Total Amount: 12063.2
8/10/2021 4:45:49 PM - [Info] Invoice Number: inv.9990

Invoice Date: 20 Oct 2021

Invoice Amount: 1208.8
Total Amount: 13272.0
8/10/2021 4:45:51 PM - [Info] Invoice Number: inv.9990

Invoice Date: 20 Nov 2021
```

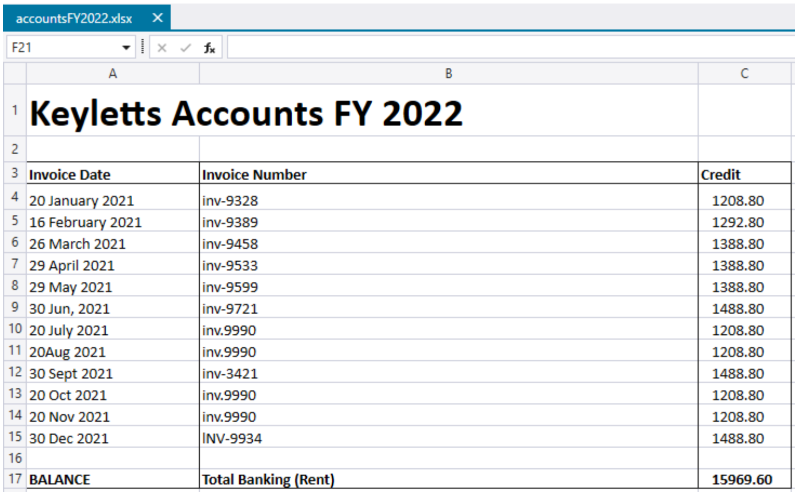
Execution succeeded

If you see anything different from the above result (for example, invoice number contains special characters), it's probably because the selector is not configured correctly.

From Studio, open the excel file:

Lab 2 – Imaging with RPA\Scenario 1\Invoices\ accountsFY2022.xlsx

Verify that rows 4-15 have been added from the PDFs:



	A	B	C
1	Keyletts Accounts FY 2022		
2			
3	**Invoice Date**	**Invoice Number**	**Credit**
4	20 January 2021	inv-9328	1208.80
5	16 February 2021	inv-9389	1292.80
6	26 March 2021	inv-9458	1388.80
7	29 April 2021	inv-9533	1388.80
8	29 May 2021	inv-9599	1388.80
9	30 Jun, 2021	inv-9721	1488.80
10	20 July 2021	inv.9990	1208.80
11	20Aug 2021	inv.9990	1208.80
12	30 Sept 2021	inv-3421	1488.80
13	20 Oct 2021	inv.9990	1208.80
14	20 Nov 2021	inv.9990	1208.80
15	30 Dec 2021	INV-9934	1488.80
16			
17	**BALANCE**	**Total Banking (Rent)**	**15969.60**

2.5 Bonus Exercise

Using the free PDF editor https://www.pdfescape.com create new invoices for a new Financial Year.

3 Scenario 2 – Surface Automation

3.1 Real World Alignment

Some applications cannot be automated by accessing GUI controls. The only way is by finding an image on the screen and clicking on it. This is known as surface automation.

3.2 Scenario Description

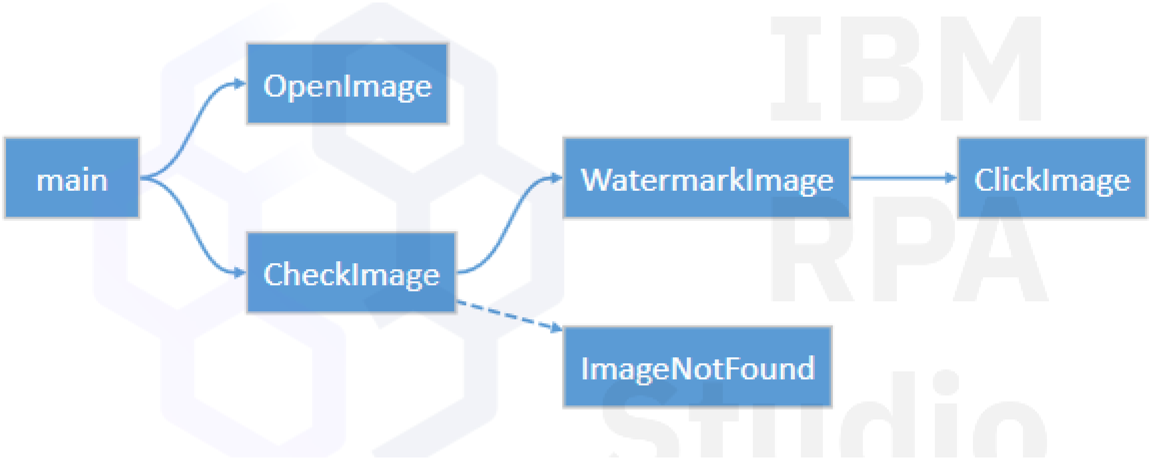
A legal company scans images for copyright. You are required to write a bot that scans images and identifies copyright infringement.

3.3 Open a work in progress script

Start IBM RPA Studio and open:

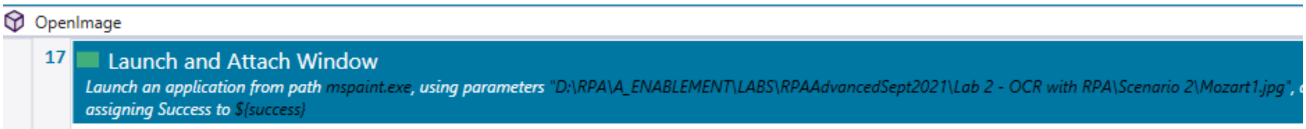
Lab 2 - Imaging with RPA/Scenario 2/Scenario 2 Started.wal.

Your script will open in Studio. Click on the *Call Graph* tab to view the flow. The bot calls *openImage* and then calls *checkImage*. The image is scanned with *ClickImage*. If the image is found it is watermarked. Otherwise, the error handler *ImageNotFound* is called.



3.4 Correct the folder

Navigate to subroutine *OpenImage*. On line 14 edit the path to point to your *Lab 2 - Imaging with RPA\Scenario 2\Mozart1.jpg*. Ensure the path is wrapped in quotes. See below:



OpenImage

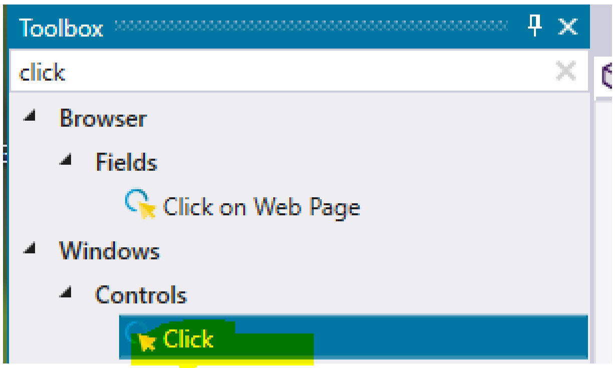
17 Launch and Attach Window

Launch an application from path mspaint.exe, using parameters "D:\RPA\A_ENABLEMENT\LABS\RPAAdvancedSept2021\Lab 2 - OCR with RPA\Scenario 2\Mozart1.jpg",
assigning Success to ${success}

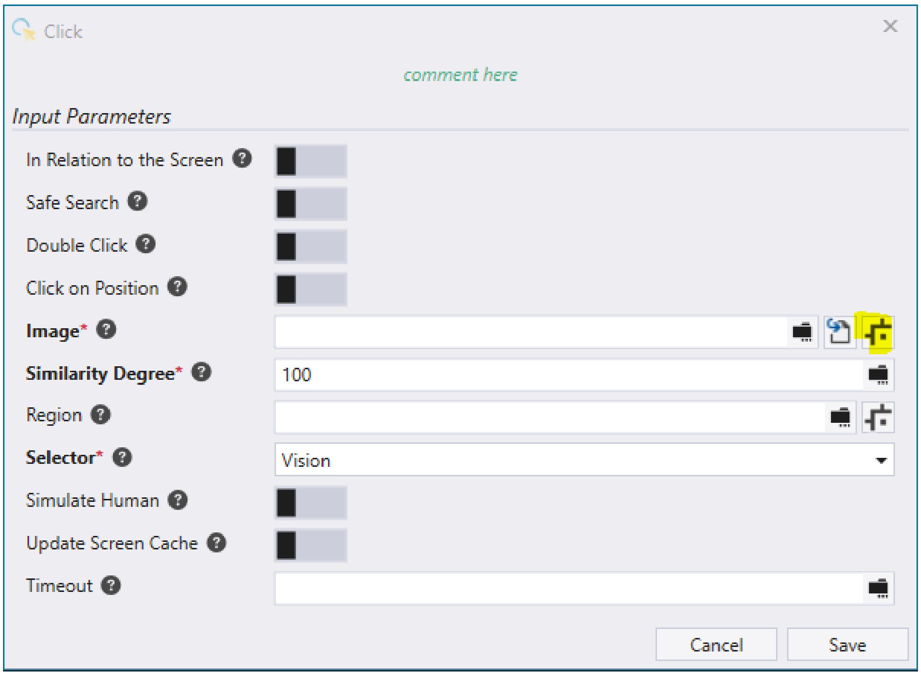
3.5 Open image

Open the image *Mozart1.jpg* with MS Paint.

In the Toolbox within RPA Studio, find the *Click* command and drag it to line 32 in subroutine *ClickImage*. .



Within the Click dialog, set *Selector* as *Vision* and select the *Region Selector* icon to the right of the *Image* text box:



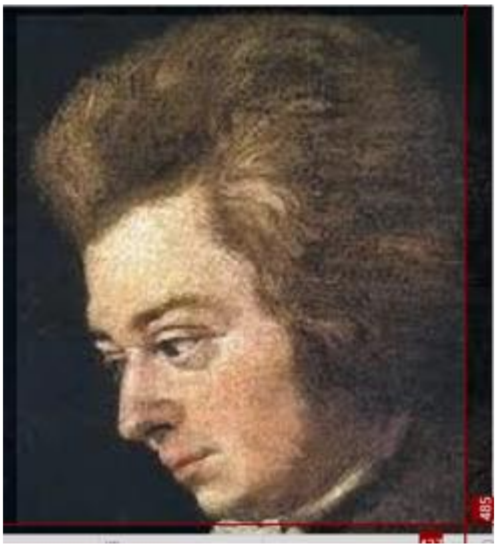
Click

comment here

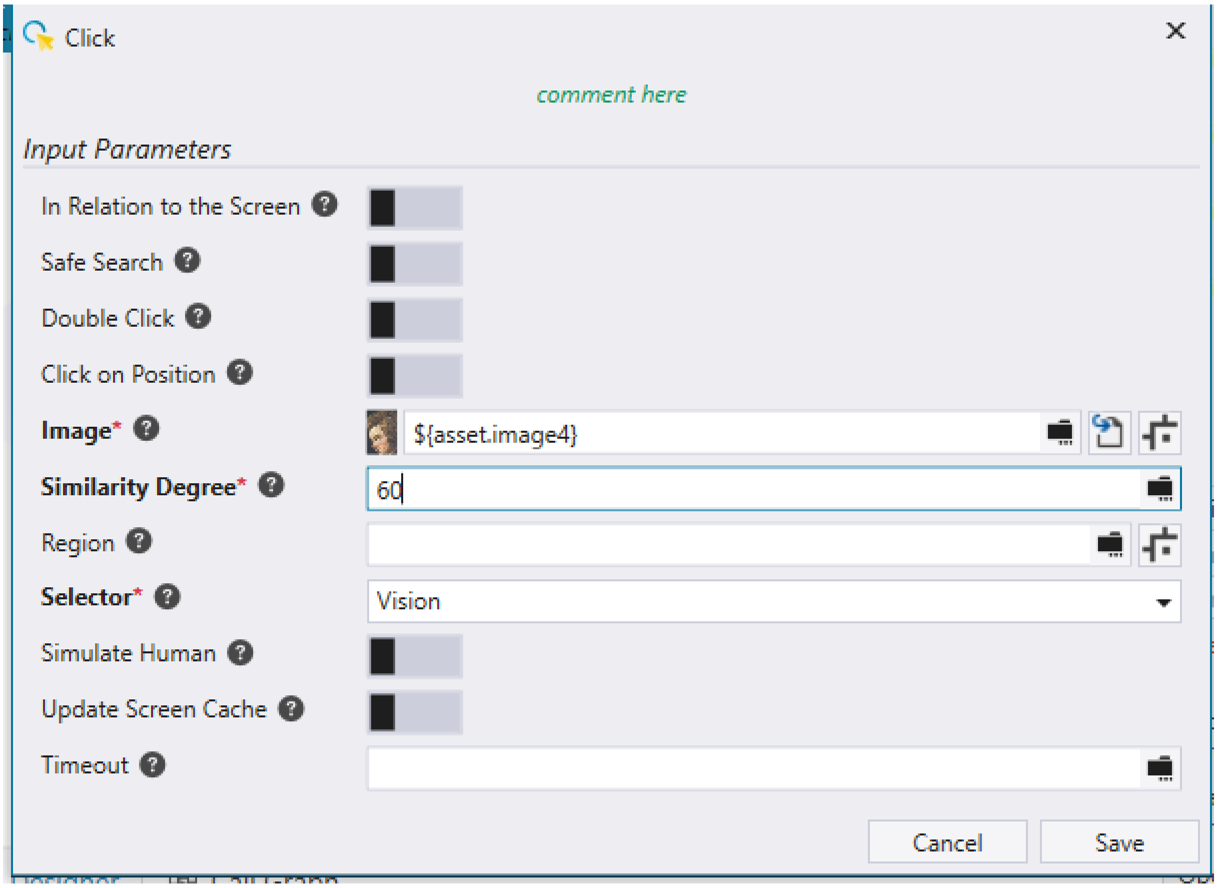
Input Parameters

In Relation to the Screen ?	
Safe Search ?	
Double Click ?	
Click on Position ?	
Image* ?	
Similarity Degree* ?	100
Region ?	
Selector* ?	Vision
Simulate Human ?	
Update Screen Cache ?	
Timeout ?	

Using the cross hatches, select the face of the composer:



Once selected, a thumbnail image of the selected area appears in the Control Image:



Click

comment here

Input Parameters

In Relation to the Screen

Safe Search

Double Click

Click on Position

Image* ${asset.image4}

Similarity Degree* 60

Region

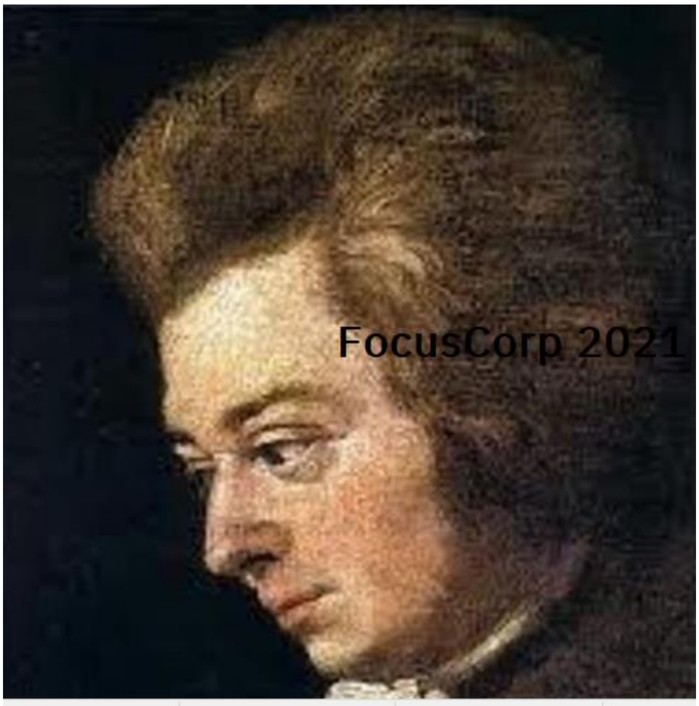
Selector* Vision

Simulate Human

Update Screen Cache

Timeout

Finally, set the Similarity Degree to 60. This will increase the tolerance to slight changes to the bitmap. Press Save. Run the script. The script should scan the bitmap for the image you selected and write a watermark. After a few seconds you will see the bitmap contain a watermark:



If you don't get this result, try reducing the *Similarity Degree* from 60 to 10 and run again.

Now repeat from section 3.4 but this time use the image:

Lab 2 - Imaging with RPA\Scenario 2\Mozart2.jpg

You must run the bot using Ctrl+F5 (no debug) to enable the error handler. Otherwise, you will get a run time error. This time the image is not watermarked. The output log contains

```
8/11/2021 11:14:13 AM - [Info] No infringement.
```

3.6 Advanced exercise

To complete the automation, iterate over a folder of several images, scanning each one in turn.

Nicely done! This concludes the lab.