UTM
UNIVERSITI TEKNOLOGI MALAYSIA

Malaysia-Japan International Institute of Technology (MJIIT)

SMJE 4383 - ADVANCED PROGRAMMING

SEMESTER 1 SESSION 2022/2023

ASSIGNMENT 1

AUTOMATE THE CSV GENERATION PROCESS USING ROBOTICS PROCESS AUTOMATION (RPA) & PYTHON

NAME	MATRIC NO
MA ATHIRAH BINTI SAPAWI	A19MJ3030
AMIRAH BATRISYIA BINTI AMINNUDIN	A19MJ0011
SECTION	1 & 2
LECTURER NAME	ASSOC PROF IR DR ZOOL HILMI BIN ISMAIL
GITHUB LINK	

TABLE OF CONTENT

1.0 INTRODUCTION

The automation of business processes is an important area of focus for organizations looking to optimize their operations. In this context, robotics process automation (RPA) has emerged as a key technology for automating repetitive and time-consuming tasks. RPA enables the creation of software robots that can perform tasks like data entry, file management, and report generation, among others. By using RPA, organizations can achieve significant cost savings, improved accuracy, and faster processing times.

One area where RPA can be particularly effective is in automating the generation of CSV files. CSV files are a common format used to store and exchange data, but creating them manually can be a time-consuming process. With RPA, organizations can automate the entire process of generating CSV files, from extracting data from various sources to formatting and exporting the final file. This can help organizations save time and reduce errors, freeing up staff to focus on higher value tasks.

Python is a popular programming language used extensively in data analysis and automation. Combining Python with RPA can enable even greater automation capabilities, allowing organizations to build customized software robots that can perform complex tasks like data manipulation and analysis. By automating the CSV generation process using RPA and Python, organizations can achieve greater efficiency, accuracy, and cost savings, while freeing up staff to focus on more strategic tasks.

In this report, we'll discuss the procedures needed to use Python and RPA to automate the CSV generating process. We'll go over RPA software installation and configuration, writing Python scripts to automate data copying, and incorporating these scripts into an RPA work flow.

2.0 OBJECTIVES

The objective of this project is:

a. To create a sample of CSV file and save it in the same directory
b. To write a python code that can automatically copy the data from the CSV file into new CSV file and then send notification to email once the process is completed.

3.0 METHODOLOGY

Below is the process flow of this experiment

Figure 1 - Process flow of project

First step of this project is a CSV file need to be created using CSVpad software. CSVpad is a handy free CSV (Comma-separated values) editor.It supports Unicode and it is a portable application.CSVpad can manipulate columns and rows. For this project the CSV file is contained students' grade for communication subject. After the CSV file was created, the file was name as Student grade communication.

	A	B	C	D	E	F	G	H	I
1	Name	Matris No	Test1 (10%)	Test2 (10%)	Assignment1 (15%)	Assignment2 (15%)	Final (50%)	Total	Grade
2	Amirah Batrisyia	A19MJ001	10	10	15	13	49	97	A+
3	Ma Athirah	A19MJ303	10	10	15	13	49	97	A+
4	Siti Nur Izzati	A19MJ306	10	10	15	14	40	89	A
5	Amirah Syahmina	A21MJ208	10	5	14	14	40	83	A
6	Tariq Muhammad	A18MJ423	10	9	13	10	30	72	B
7	Siti Qaleesya	A20MJ230	9	6	12	10	35	72	B
8	Muhammad Haziq	A19M0073	8	10	15	12	42	87	A
9	Muhammad Ikhwa	A18MJ986	7	4	13	5	33	62	C
10	Izwan Afif	A17MJ003	5	10	15	12	22	64	C
11	Ahmad	A20MJ098	10	3	13	5	30	61	C
12	Siti Nur Fathimah	A17MJ560	2	10	10	8	35	65	C+
13	Siti Haziqah	A20MJ456	5	10	14	8	45	82	A
14	Batrisyia Amalina	A21MJ234	2	10	12	10	20	54	D
15	Nur Balqis Amani	A19MJ303	6	9	11	10	40	76	B+
16	Ilham Afiq	A20MJ000	8	8	11	12	40	79	B+
17	Muhammad Khairi	A21MJ300	10	5	12	12	42	81	A
18									

Figure 2 - CSV file created

In Figure 3 is the code in Python that will demonstrated how copy data from a CSV file and paste it in the same directory and send notification to email:

```
Open   assignment1.py  ~/Desktop   Save

1 import smtplib
2
3 # This part will open both fileSS
4 with open('Student_grades_Communication.csv','r') as firstfile,
5 open('Student_grades_Communication(final).csv','w') as secondfile:
6
7 # read content from first file
8         for line in firstfile:
9
10 # append content to second file
11                 secondfile.write(line)
12
13 #coding for sent notification to email
14 content = ("Hi, Want to tell you that the data from Students's Grade for
   Communication was succesfully update")
15 mail = smtplib.SMTP('smtp.gmail.com',587)
16 mail.ehlo()
17 mail.starttls()
18 mail.login('amirahbatrisyia70@gmail.com','wwqmkphegeqmaksx')
19 mail.sendmail('amirahbatrisyia70@gmail.com','maathirah@graduate.utm.my',content)
20 mail.close()
21 #notify that the email was succesfully sent
22 print("copy data process was successful and notification was sent to email")

Python 2   Tab Width: 8   Ln 20, Col 44   INS
```

Figure 3 - Coding for copy data and

In the red square shape is coding for copy data. This task of copy data can be done by using file mode. To copy the data, ‘r’ mode is for read only file, so it will open the file and read it and for executed the data to new CSV, ‘w’ mode is for write data. In the computer there is no file name for

\Student_grades_Communication(final).csv so because the file is no exist then the new file was created and opened. In the coding, ‘Student_grades_Communication.csv’ file was read and ‘Student_grades_Communication(final).csv’

In yellow square shape is consist of code for sent notification to email. After the data from CSV file has been copied to new CSV file, we need some notification to notify the user that the process was completed. So the method is by sending an email notification. To send email notification, the **smtplib** library in Python was used. The **smtplib** module defines an SMTP client session object that can be used to send mail to any internet machine with an SMTP. This method required recipient’s email address and apps passwords for login purpose, the sender’s email address and the email content as the arguments.

4.0 RESULT AND DISCUSSION

When the coding was executed, a “copy data process was successful and notification was sent to email” message will appear on the terminal after the file was copied. This will show that the data inCSV file was copied and the user had been notified. In Figure 4 show the text displayed on the terminal after the process was completed

```
mira@mira-virtual-machine:~/Desktop$ python3 assignment1.py
copy data process was successful and notification was sent to email
mira@mira-virtual-machine:~/Desktop$
```

Figure 4 - Text displayed after email notification was sen

In Figure 5 is the original CSV file (Student_grades_Communication.csv) and the copy file (Student_grades_Communication(final).csv)

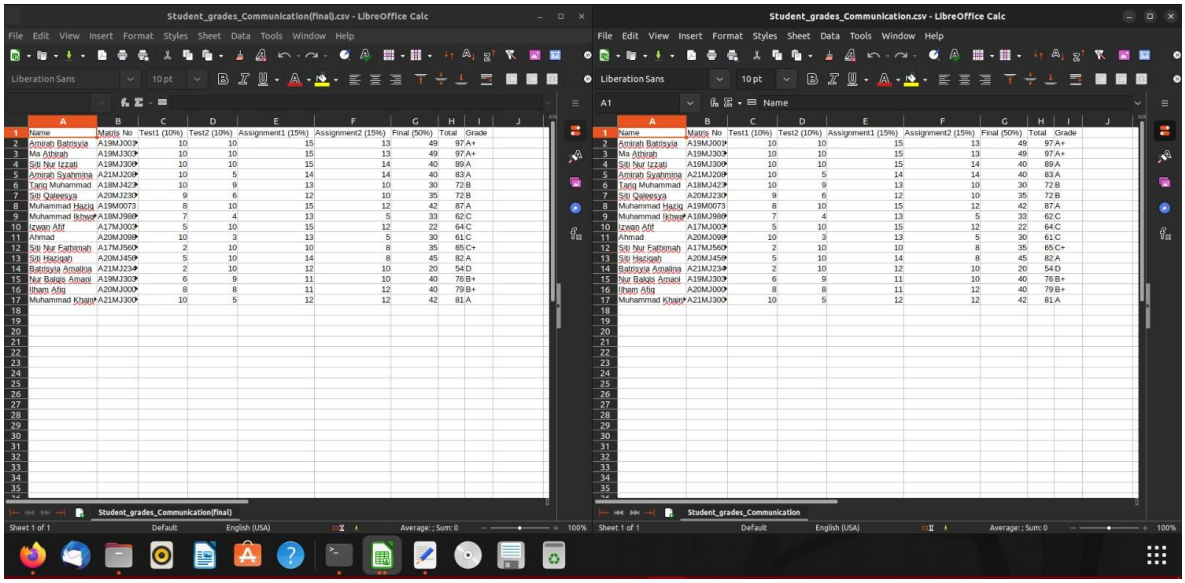

Figure 5 - Original CSV file (Student_grades_Communication.csv)(*right*) and the copy file (Student_grades_Communication(final).csv)(*left*)

In Figure 6 show the email notification that will be send to user when the process completed.

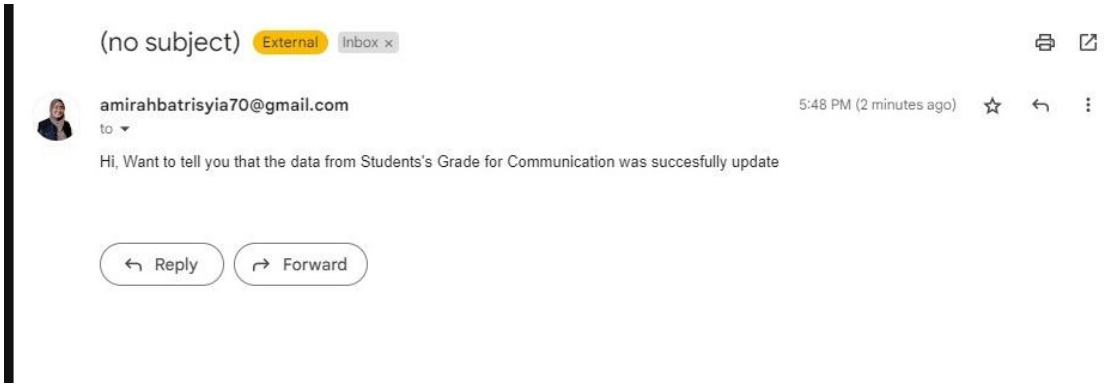

Figure 6 - Email notification sent when the process completed

Before process for the notification can be completed, an additional steps needed which is need to create the python apps for the sender email. It will give us 16 character password like a temporary password dor sender email to send email using Python. Figure 7 is the 16 character password that was generated.

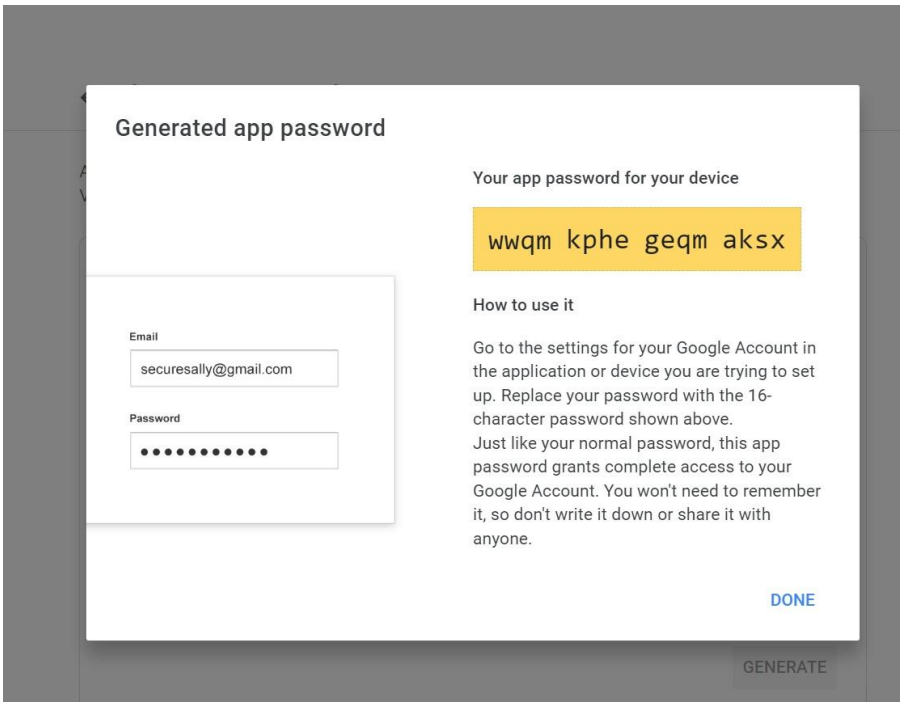

Figure 7 - 16 character password for python uses