

Configuration Manual

MSc Research Project
MSc Cloud Computing

Mukul Anaspure
Student ID: x21118141

School of Computing
National College of Ireland

Supervisor: Shivani Jaswal

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: Mukul Anaspure
Student ID: x21118141
Programme: MSc Cloud Computing **Year:** Jan 2022
Module: MSc Research Project
Lecturer: 15th December 2022
Submission Due Date:
Project Title: Automation in cloud environment using cloud services and python script
Word Count: **1057 Page Count: 7**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Mukul Anaspure

Date: 15th December 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Mukul Anaspure
x21118141

1 Pre-requisites and access required to cloud services

Anyone who is familiar with the fundamentals of cloud computing can comprehend the primary concept and follow the procedures without difficulty. It is advised to be familiar with the fundamentals of AWS so that everyone can understand the terminologies and cloud services stated.

Following services are required for Azure cloud:

- Azure Subscription
- Azure Logic Apps
- Function App
- Virtual Machines
- Recovery Service Vault
- Azure Container Registry

Services required other than cloud:

- Docker Desktop
- Anaconda Spyder or Microsoft Visual Studio (with Docker CLI and Azure CLI installed)
- Selenium Web driver

In this manual we will first look at the configuration that is required for backup automation and then we will see configuration of deduplication technique.

2 Setup

As a initial step we need to login azure portal by using the link portal.azure.com. Everyone will see below page as shown in figure 1 but with different services at the start. This depends on the user which services he/she has used recently.

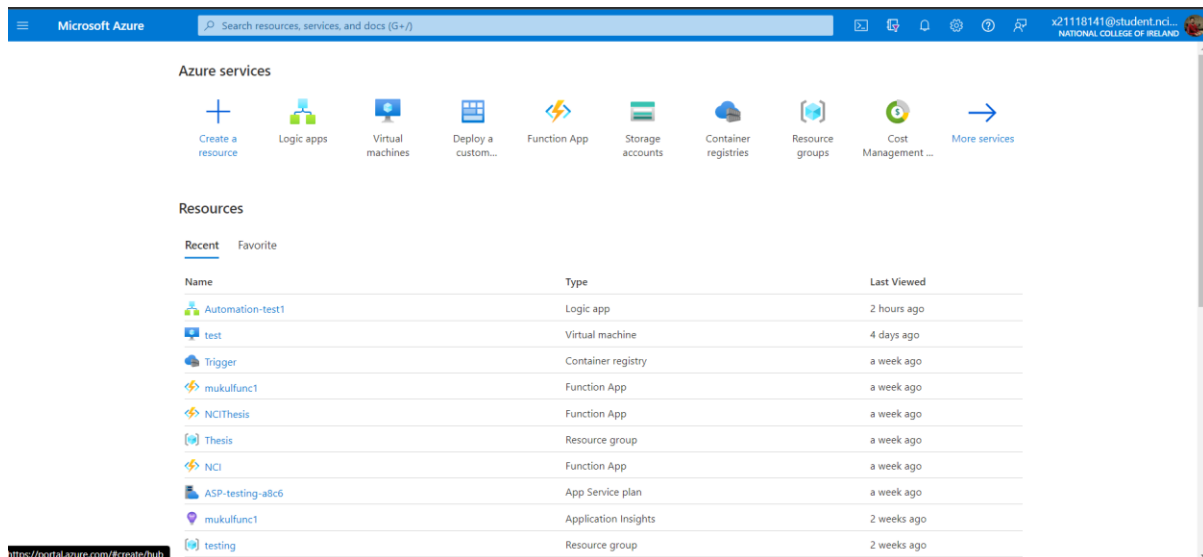


Figure 1

Now first we will create all the services required for the setup. But first we will create a resource group because in Azure every resource resides inside resource group. For creating a resource search for resource group in search bar. When the page is loaded then it will ask for the subscription, name for resource group and region for the resource. After entering all the details click on create. Resource group will be created as shown in figure 2.

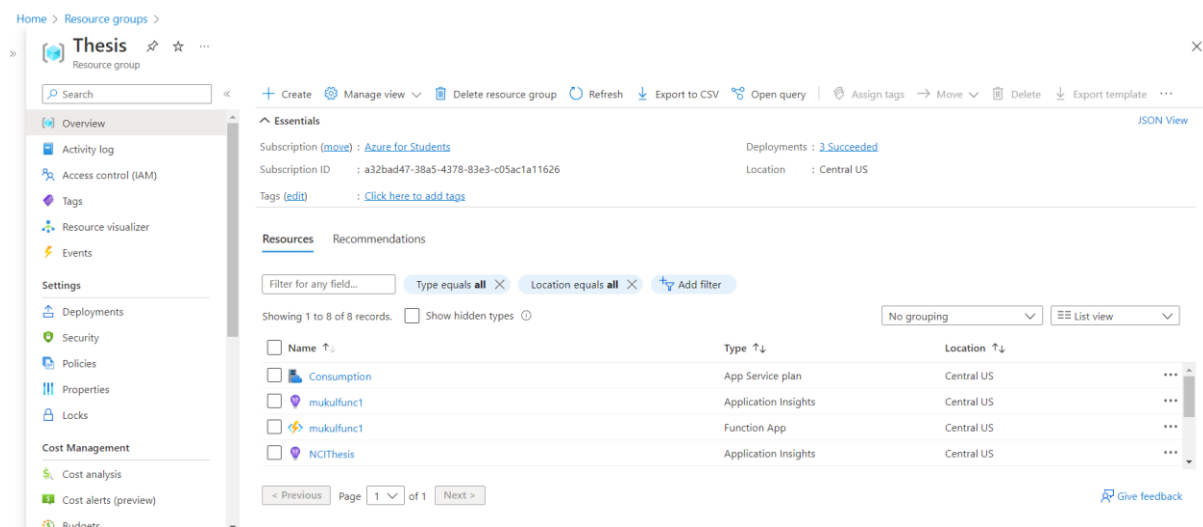


Figure 2

Now we will create an virtual machine. For this, search for Virtual Machines in search bar. Once the page is loaded it will ask for the resource group. Enter the resource group name which we created in the above step. Now it will ask for the operating system – Select Windows. Now enter your username and password. Make sure that RDP port is open. This will help us to login to the server with the help of credentials. Now click on create + validation. Finally hit create. It will create and virtual machine as shown in figure 3.

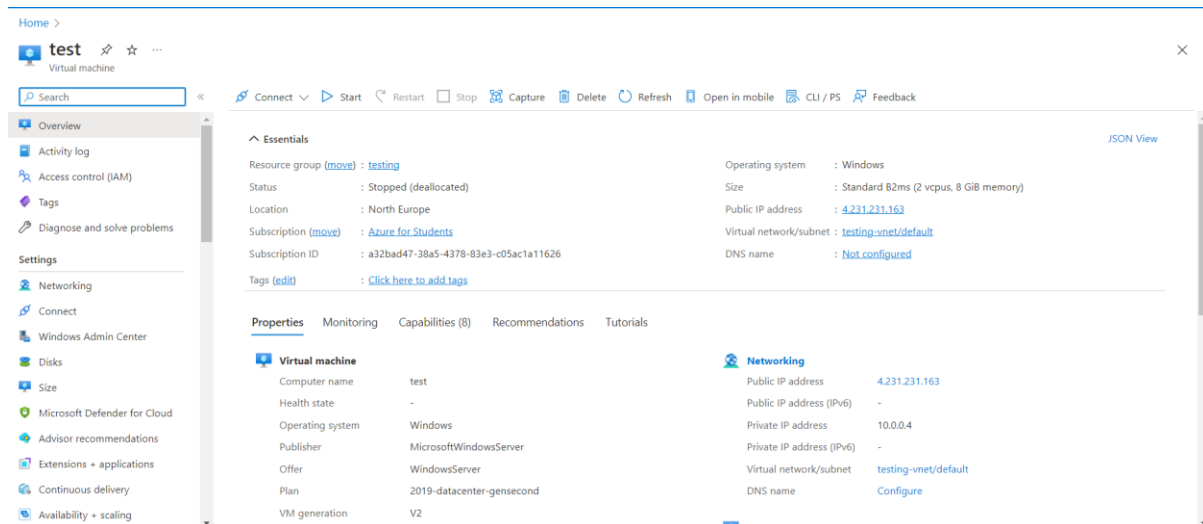


Figure 3

Now we will create our next service which is a main service that is Azure Logic Apps. Search for Logic Apps in the search bar and click on Add. It will ask for resource group name, select the one which we have created in the previous steps. Give a name to Logic Apps. For publish click on workflow and select the region same as your resource group. Click on create. Once the logic app is created then on the left side look for “Logic App Designer”. After opening this we can create our flow. Below figure 4 shows the workflow which we have created for the research.

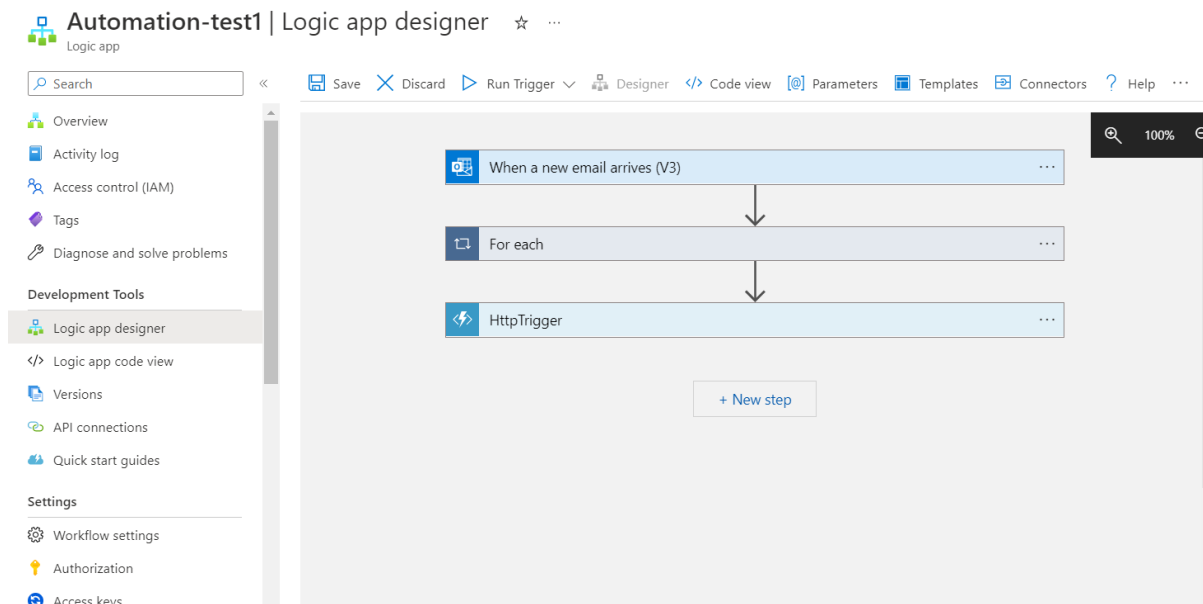


Figure 4

In the next step we will create Azure Container Registry which will store our Docker image containing all the selenium libraries. To create, search container registries in search box. Click on Create. It will ask for resource group – enter the name. Now enter the fields as per below information.

1. Registry name – Trigger

2. Location – same as resource group
3. SKU – Basic

Azure Container registry is created as shown in figure 5.

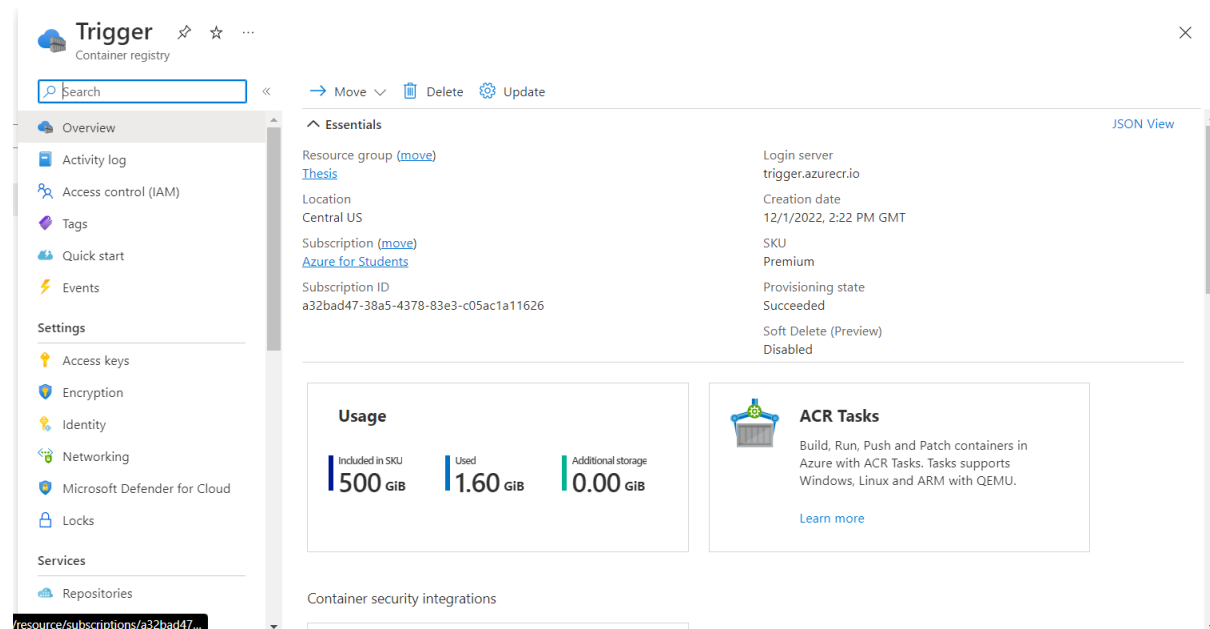


Figure 5

Now in further step we will create an docker image which will contain all selenium libraries and then create azure function with the help of this image.

For creating a docker image run the below commands one by one in visual studio terminal or powershell. For <<username>> and <<password>> we can look in azure container registry (acr) and in access tabs.

- `$acr_id = "<<your acr>>.azurecr.io"` – *setting the variable*
- `docker login $acr_id -u "Enter your username" -p "Enter your password" --` *creating docker image*
- `docker build --tag $acr_id/"name of image" . --` *buildig docker image*
- `docker push $acr_id/selenium:latest --` *pushing the image to ACR*

After running these commands we can the image named 'latest' created in ACR → Repositories.

Now our next step is to create an azure function and deploy the docker image from ACR. For this, we need to run the following commands but before we need to define some variables as follows.

Defining the variables

`$rgroup = "Enter resource group name"`

`$location = "Location name"`

`$planname = "Plan name: Consumption or serverless"`

`$storage = "Storage account name"`

`$function = "Function name"`

```
$azure_cr_id = "<<your acr>>.azurecr.io"
```

Once the variables are defined we can run the following commands which will create resource group (in our case already created), a storage account and app service plan.

- `az storage account create -n $storage -g $rgroup --sku Standard_LRS` – *creating storage account*
- `az appservice plan create --name $planname --resource-group $rgroup --sku consumption --is-linux` -- *creating app service plan*

Finally we will create an azure function using the docker image.

Now we can call this function in our workflow in Logic Apps as shown in figure 6.

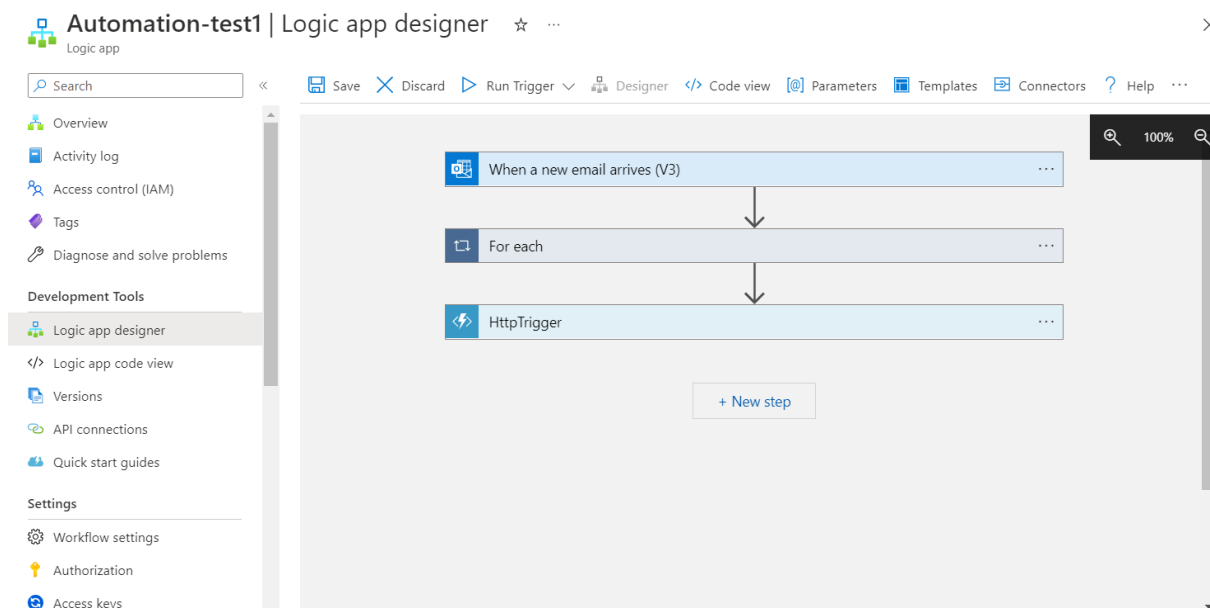


Figure 6

When a mail would come in Outlook about the backup request then it would trigger the workflow and start the trigger as shown in figure 7.

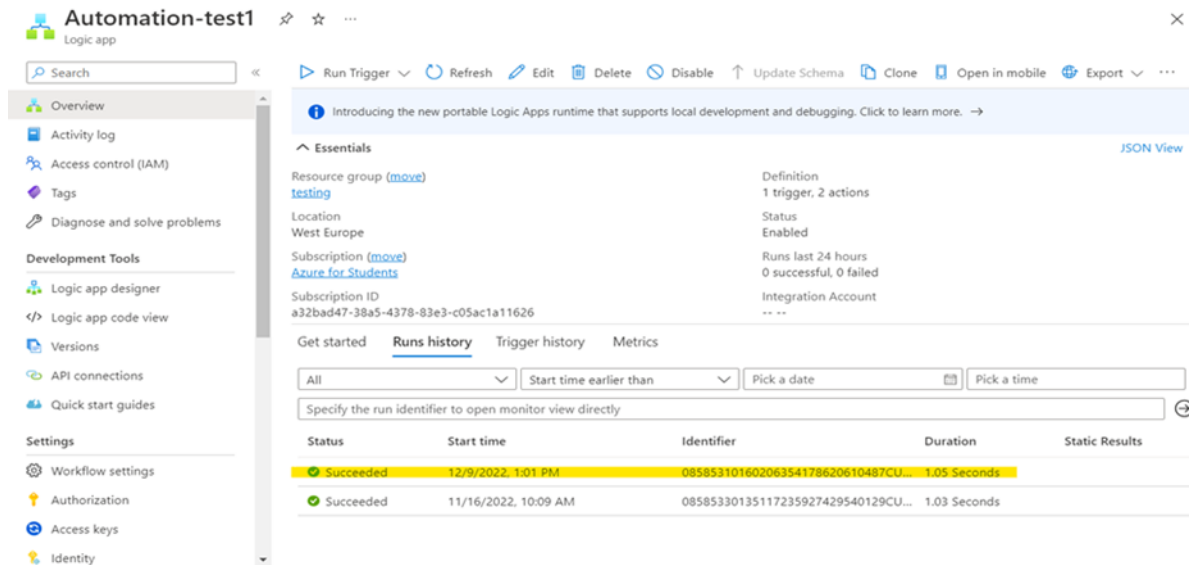


Figure 7

For Deduplication technique, the setup is not very complex.

Initially we need to have visual studio and python installed.

Once it is installed then open visual studio and create a new file <<name>>.py. Once the file is created we need to import some libraries like tkinter, hashlib, pathlib because they are used in our code. Then take the code which is there uploaded in Source files.

Now to test the code, we have to make some folder-file structure. First download multiple audio files from the internet. We need to make sure that it is in .wav format. Now keep the folder-file structure as shown in figure 8.

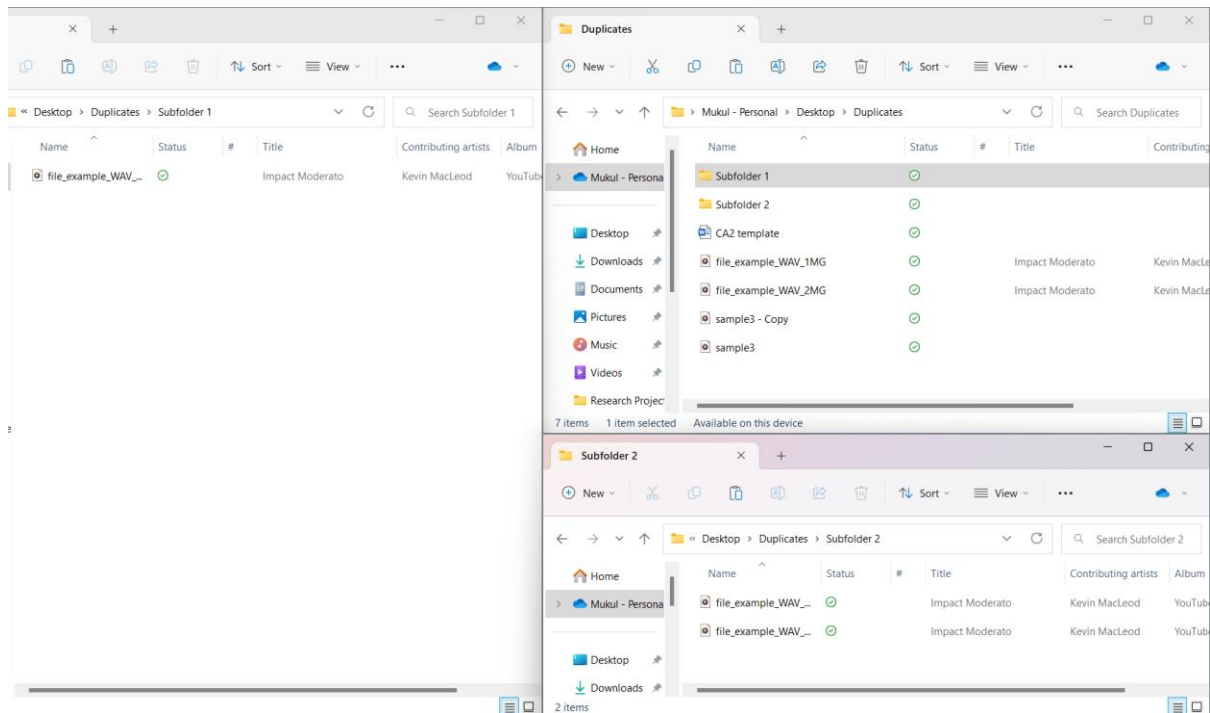


Figure 8

Now when the script is run then it can detect all the duplicate values and suggests which one to delete. Please see figure 9.

```

C:\Users\mukul\OneDrive\Desktop\Automation> & C:/Users/mukul/Anaconda3/Anaconda/python.exe c:/Users/mukul/OneDrive/Desktop/Automation/duplicate.py
3d3f546b9a5eab0279df4f19729776ed
32
f2101cf866dea872dbaeb8ac1e6cc9ae
32
773a3cc4edc23dd7a6cec71883d63b5d
32
a605903733e89769ef47f3f6a6b0bdbf
32
a605903733e89769ef47f3f6a6b0bdbf
32
C:\Users\mukul\OneDrive\Desktop\Duplicates\sample3.wav need to be deleted
773a3cc4edc23dd7a6cec71883d63b5d
32
C:\Users\mukul\OneDrive\Desktop\Duplicates\Subfolder 1\file_example_WAV_2MG.wav need to be deleted
f2101cf866dea872dbaeb8ac1e6cc9ae
32
C:\Users\mukul\OneDrive\Desktop\Duplicates\Subfolder 2\file_example_WAV_1MG.wav need to be deleted
773a3cc4edc23dd7a6cec71883d63b5d
32
C:\Users\mukul\OneDrive\Desktop\Duplicates\Subfolder 2\file_example_WAV_2MG.wav need to be deleted
(base) PS C:\Users\mukul\OneDrive\Desktop\Automation>

```

Figure 9

We can schedule this script as and when required. In our research we have scheduled this script with the help of Windows Task Scheduler. This is a free service given by Microsoft to all windows users which allows to schedule any script to automate the repetitive tasks.