

EXPERIMENT NUMBER – 3.3

| |
|--------|
| NAME – |
| UID – |
| SEC- |
| BRANCH |
| SUB |
| SEM- |
| D.O.P |

AIM– On the basis of Tuples learning write and give the output of the followings programs

1. Write a Python program to generate 26 text files named A.txt, B.txt, and so on up to Z.txt
2. Write a Python program to create a file where all letters of English alphabet are listed by specified number of letters on each line
3. Write a Python program to read a random line from a file.
4. Write a Python program to count the frequency of words in a file
5. Write a Python program to copy the contents of a file to another file

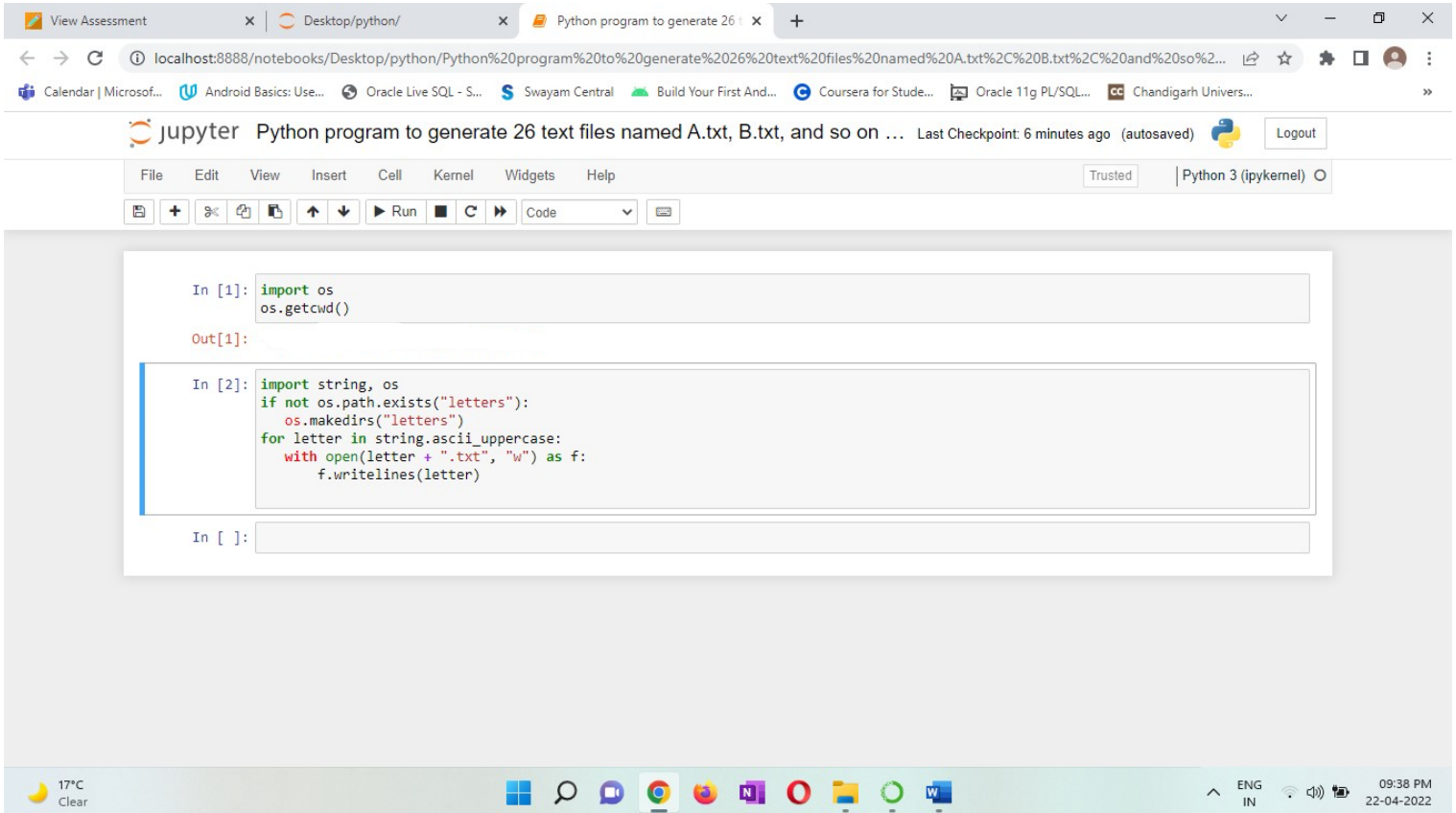
Solution–

1. Write a Python program to generate 26 text files named A.txt, B.txt, and so on up to Z.txt

Program Code –

```
import string, os if not
os.path.exists("letters"):
    os.makedirs("letters")
for
letter in string.ascii_uppercase:
with open(letter + ".txt", "w") as f:
    f.writelines(letter)
```

Screenshot of Code-



The screenshot displays a Jupyter Notebook interface in a web browser. The browser tabs include "View Assessment", "Desktop/python/", and "Python program to generate 26...". The address bar shows the URL: localhost:8888/notebooks/Desktop/python/Python%20program%20to%20generate%2026%20text%20files%20named%20A.txt%2C%20B.txt%2C%20and%20so%20... The Jupyter interface shows the title "Python program to generate 26 text files named A.txt, B.txt, and so on ..." and a "Logout" button. The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The toolbar contains icons for file operations and execution. The code area shows two input cells:

```
In [1]: import os
os.getcwd()

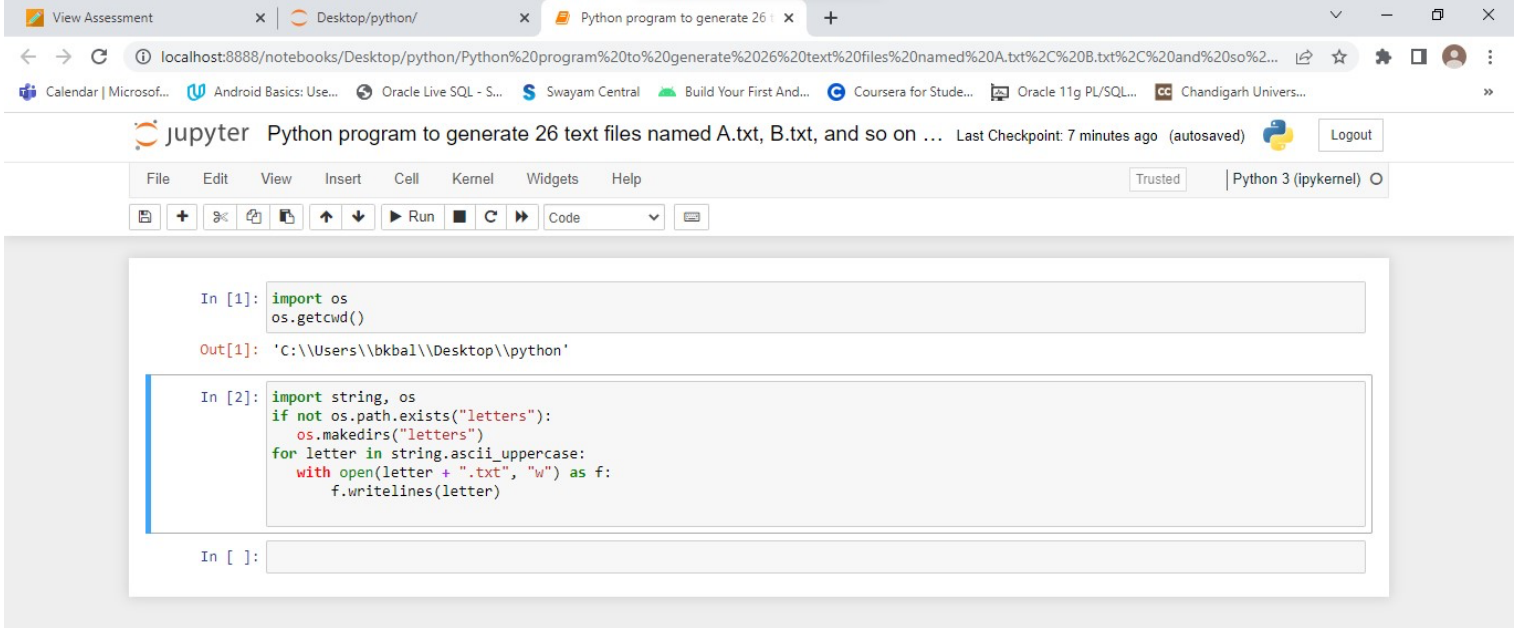
Out[1]:

In [2]: import string, os
if not os.path.exists("letters"):
    os.makedirs("letters")
for letter in string.ascii_uppercase:
    with open(letter + ".txt", "w") as f:
        f.writelines(letter)

In [ ]:
```

The Windows taskbar at the bottom shows the system tray with a temperature of 17°C, language set to ENG IN, and the date and time as 09:38 PM on 22-04-2022.

Output-



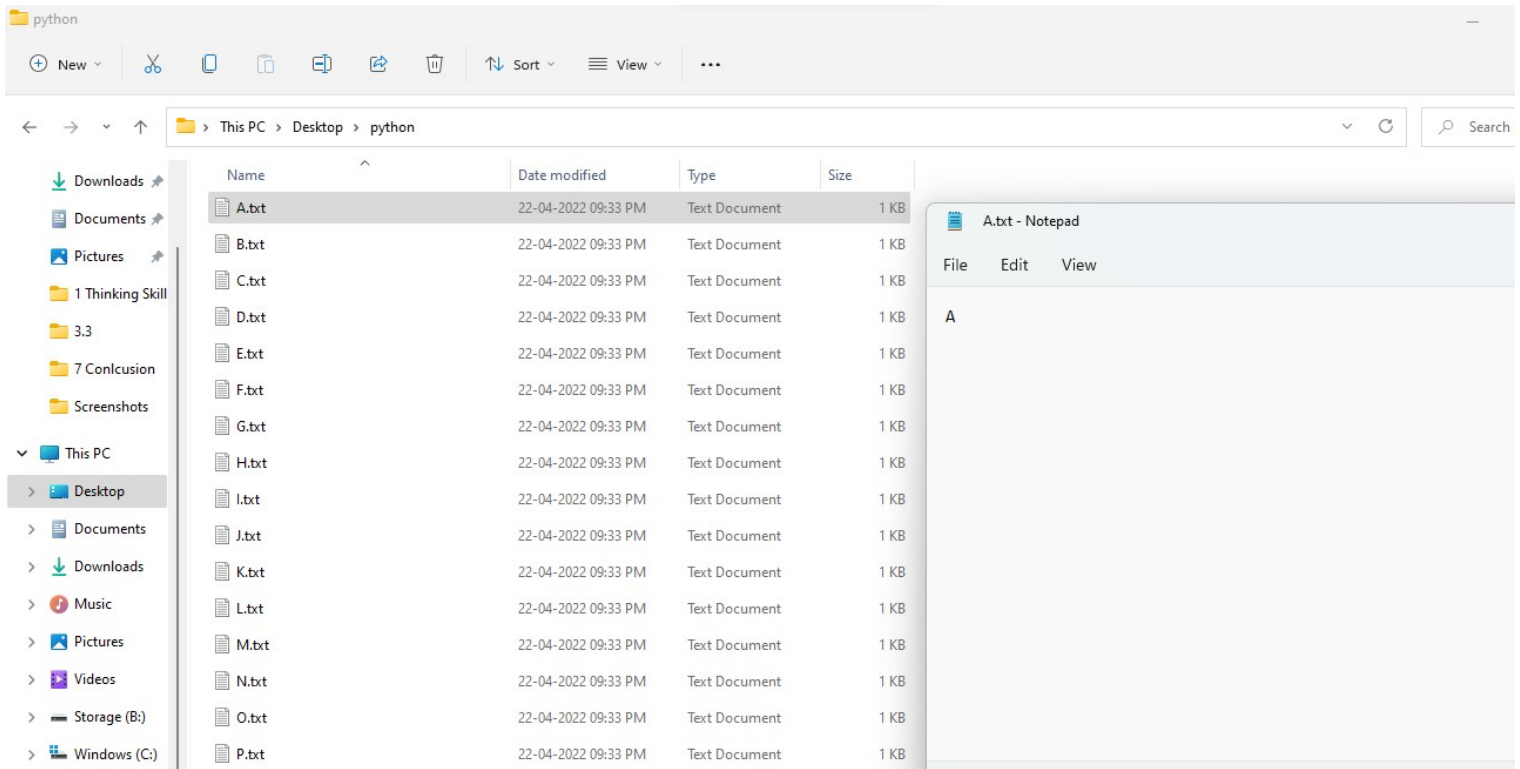
The screenshot shows a Jupyter Notebook with two code cells. The first cell contains the code to get the current working directory, and the second cell contains the code to generate 26 text files named A.txt through P.txt.

```
In [1]: import os
os.getcwd()

Out[1]: 'C:\\Users\\bkbai\\Desktop\\python'
```

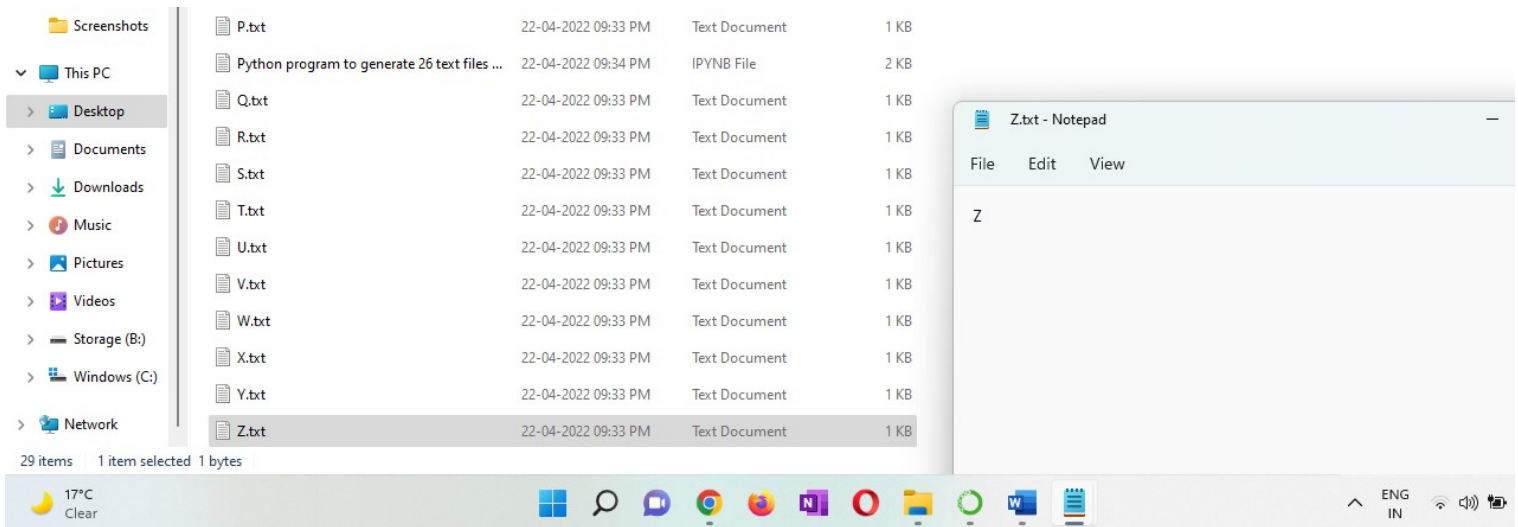
```
In [2]: import string, os
if not os.path.exists("letters"):
    os.makedirs("letters")
for letter in string.ascii_uppercase:
    with open(letter + ".txt", "w") as f:
        f.writelines(letter)
```

In []:



The screenshot shows a Windows File Explorer window displaying a directory named 'python' on the Desktop. The directory contains 26 text files, each named with a letter from A to P, all with a size of 1 KB and a date modified of 22-04-2022 09:33 PM. A Notepad window titled 'A.txt - Notepad' is open, showing the letter 'A'.

| Name | Date modified | Type | Size |
|-------|---------------------|---------------|------|
| A.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| B.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| C.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| D.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| E.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| F.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| G.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| H.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| I.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| J.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| K.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| L.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| M.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| N.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| O.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| P.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |



2. Write a Python program to create a file where all letters of English alphabet are listed by specified number of letters on each line

Program Code –

```
import string
def letters_file_line(n):
    with open("letter.txt", "w") as f:
        alphabet = string.ascii_uppercase
        letters = [alphabet[i:i
+ n] + "\n" for i in range(0, len(alphabet), n)]
        f.writelines(letters)
letters_file_line(3)
```

Screenshot of Code–



The screenshot shows a Jupyter Notebook interface in a web browser. The browser tabs include 'View Assessment', 'Desktop/python/', and 'Python program to create a file...'. The address bar shows the URL: localhost:8888/notebooks/Desktop/python/Python%20program%20to%20create%20a%20file%20where%20all%20letters%20of%20English%20alphabet%20are%20... The notebook title is 'Python program to create a file where all letters of English alphabet are...'. The code in the cell is:

```
In [2]: import string
def letters_file_line(n):
    with open("letter.txt", "w") as f:
        alphabet = string.ascii_uppercase
        letters = [alphabet[i:i + n] + "\n" for i in range(0, len(alphabet), n)]
        f.writelines(letters)
letters_file_line(3)
```

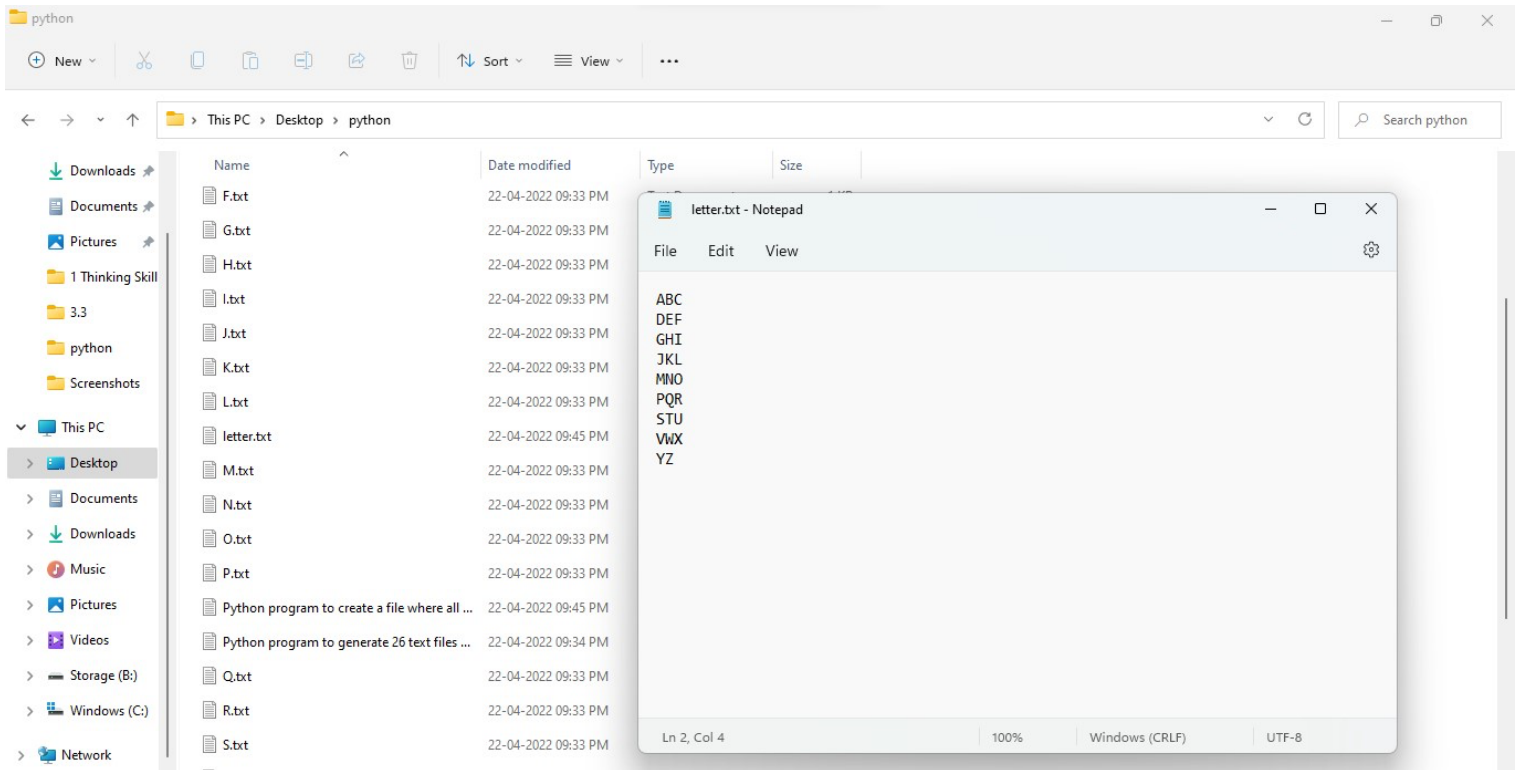
The output area below the code is empty. The system tray at the bottom shows a temperature of 17°C, a clear sky, and the time 09:47 PM on 22-04-2022.

Output-

This screenshot is identical to the one above, showing the same Jupyter Notebook interface with the same Python code. The code is:

```
In [2]: import string
def letters_file_line(n):
    with open("letter.txt", "w") as f:
        alphabet = string.ascii_uppercase
        letters = [alphabet[i:i + n] + "\n" for i in range(0, len(alphabet), n)]
        f.writelines(letters)
letters_file_line(3)
```

The output area is empty. The system tray at the bottom shows the time 09:47 PM on 22-04-2022.



3. Write a Python program to read a random line from a file.

Program Code –

```
import random  
def random_line(fname):  
    lines = open(fname).read().splitlines()  
    return random.choice(lines)  
print(random_line('apple.txt'))
```

Screenshot of Code–



The screenshot shows a web browser window with a Jupyter Notebook interface. The browser tabs include 'View Assessment', 'Desktop/python/', and 'Python program to read a random line from a file'. The address bar shows the URL: localhost:8888/notebooks/Desktop/python/%20Python%20program%20to%20read%20a%20random%20line%20from%20a%20file.ipynb. The Jupyter Notebook title is 'Python program to read a random line from a file' with a 'Last Checkpoint: 4 minutes ago (autosaved)' and a 'Logout' button. The notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations and execution. The code cell contains the following Python code:

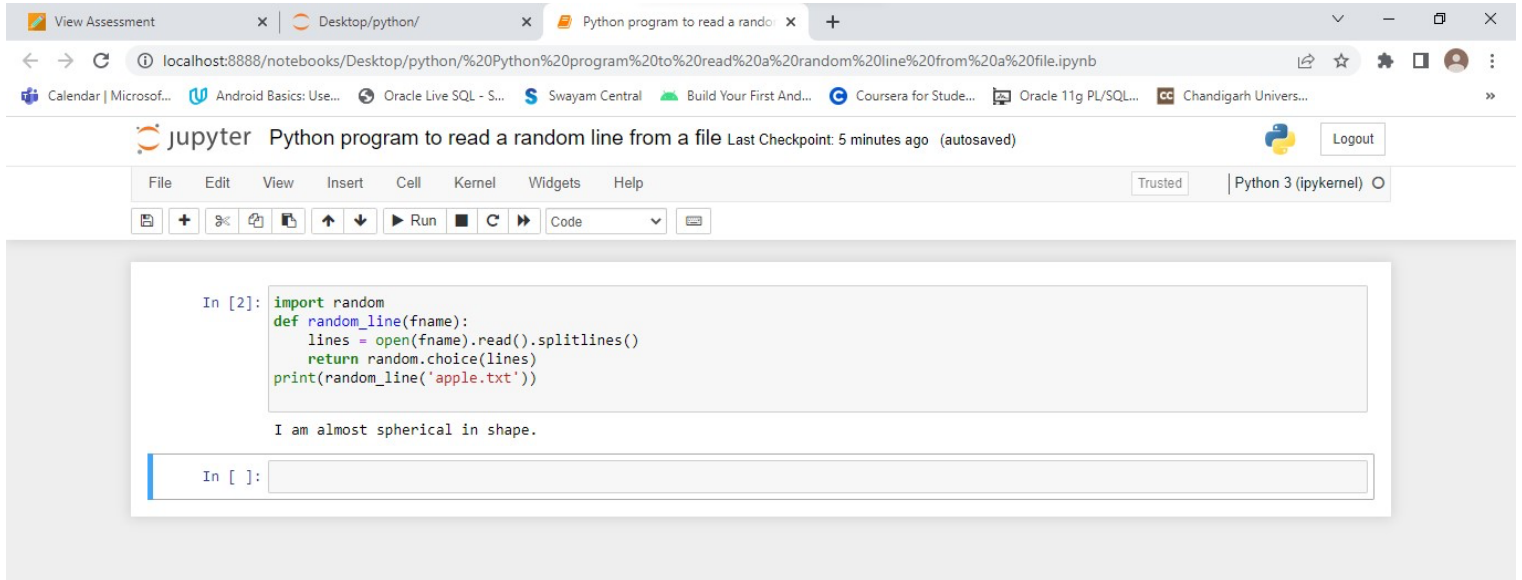
```
In [2]: import random
def random_line(fname):
    lines = open(fname).read().splitlines()
    return random.choice(lines)
print(random_line('apple.txt'))
```

The output of the code is displayed below the cell:

```
I am almost spherical in shape.
```

Below the output, there is an input prompt 'In []:' followed by an empty text box. At the bottom of the browser window, the Windows taskbar is visible, showing the system tray with the date and time: 09:55 PM, 22-04-2022.

Output-



View Assessment | Desktop/python/ | Python program to read a random line from a file

localhost:8888/notebooks/Desktop/python/%20Python%20program%20to%20read%20a%20random%20line%20from%20a%20file.ipynb

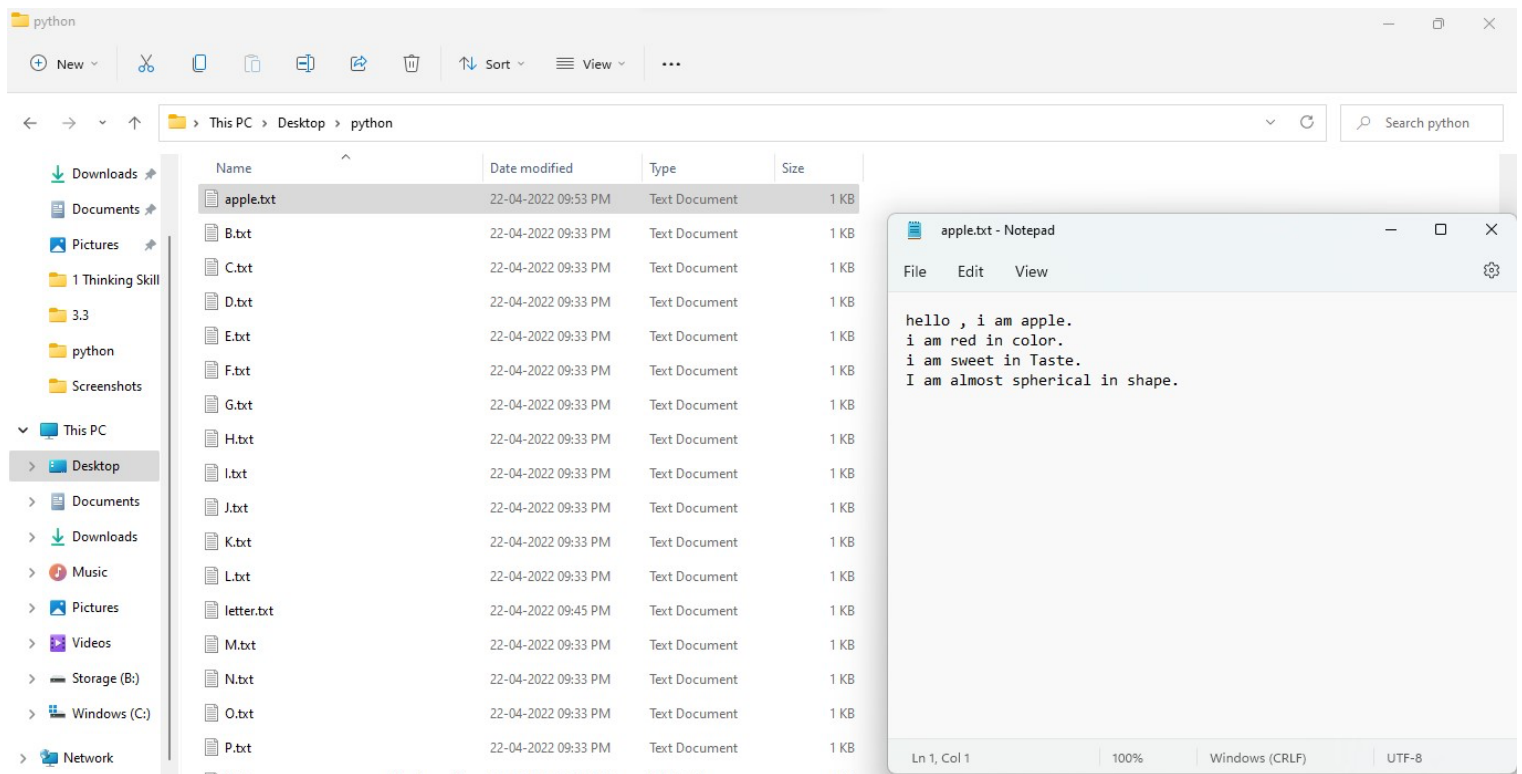
jupyter Python program to read a random line from a file Last Checkpoint: 5 minutes ago (autosaved)

```
In [2]: import random
def random_line(fname):
    lines = open(fname).read().splitlines()
    return random.choice(lines)
print(random_line('apple.txt'))
```

I am almost spherical in shape.

In []:

apple.txt file-



python

This PC > Desktop > python

| Name | Date modified | Type | Size |
|------------|---------------------|---------------|------|
| apple.txt | 22-04-2022 09:53 PM | Text Document | 1 KB |
| B.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| C.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| D.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| E.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| F.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| G.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| H.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| I.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| J.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| K.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| L.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| letter.txt | 22-04-2022 09:45 PM | Text Document | 1 KB |
| M.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| N.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| O.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |
| P.txt | 22-04-2022 09:33 PM | Text Document | 1 KB |

apple.txt - Notepad

```
File Edit View
```

hello , i am apple.
i am red in color.
i am sweet in Taste.
I am almost spherical in shape.

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8

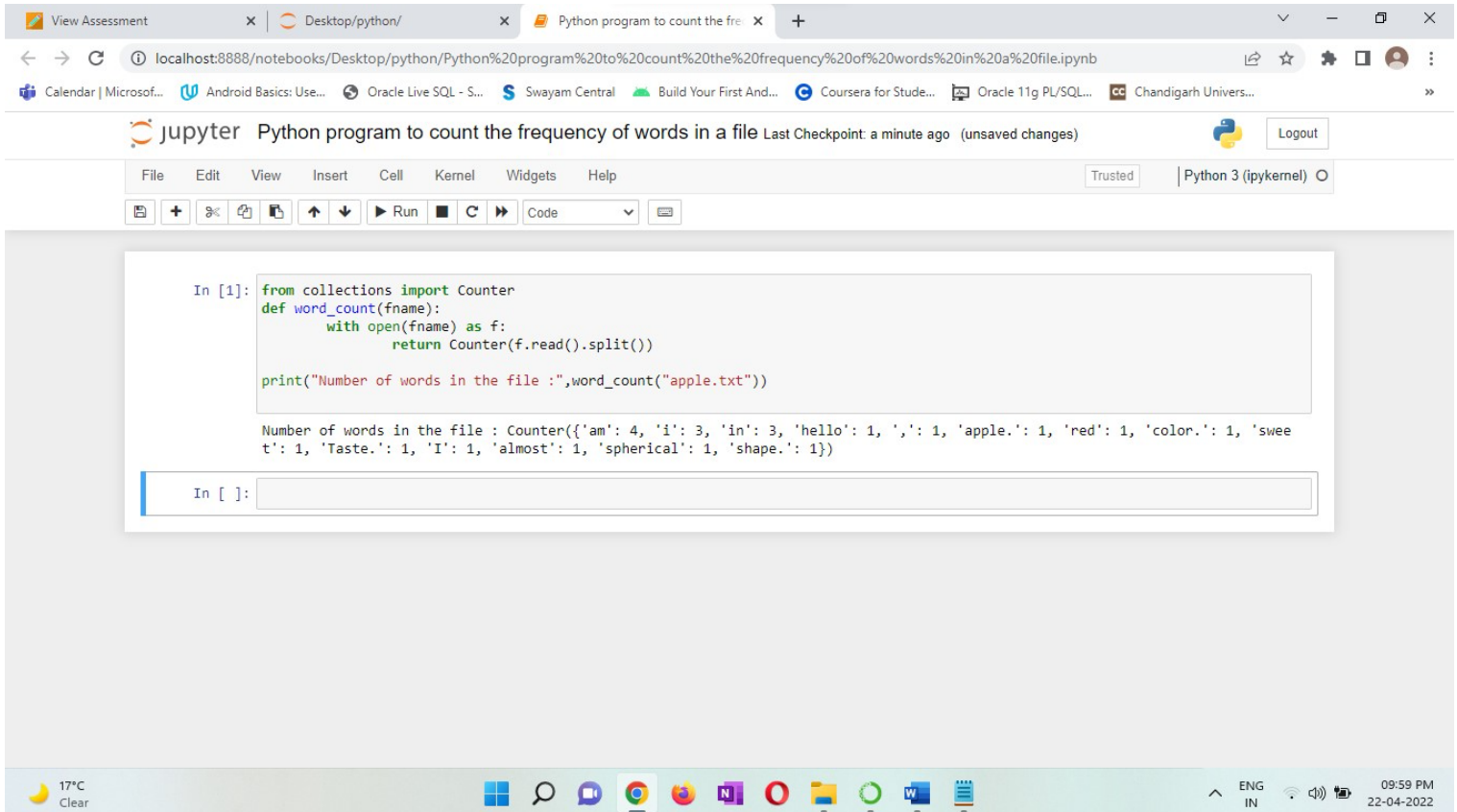
4. Write a Python program to count the frequency of words in a file

Program Code –

```
from collections import
Counter
def word_count(fname):
    with open(fname) as f:
        return Counter(f.read().split())

print("Number of words in the file :",word_count("apple.txt"))
```

Screenshot of Code–



The screenshot shows a Jupyter Notebook interface with the following code and output:

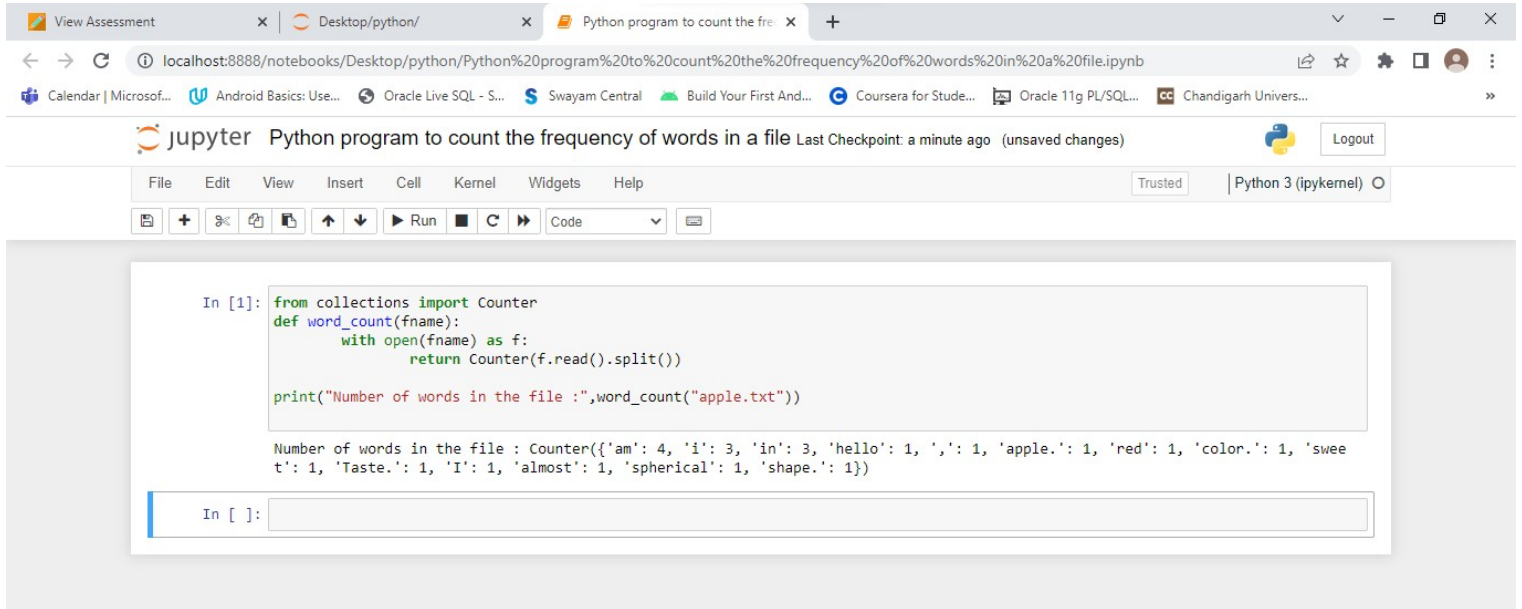
```
In [1]: from collections import Counter
def word_count(fname):
    with open(fname) as f:
        return Counter(f.read().split())

print("Number of words in the file :",word_count("apple.txt"))
```

Number of words in the file : Counter({'am': 4, 'i': 3, 'in': 3, 'hello': 1, ',': 1, 'apple.': 1, 'red': 1, 'color.': 1, 'sweet': 1, 'Taste.': 1, 'I': 1, 'almost': 1, 'spherical': 1, 'shape.': 1})

The screenshot also shows the browser tabs, the Jupyter Notebook title bar, and the Windows taskbar at the bottom.

Output-



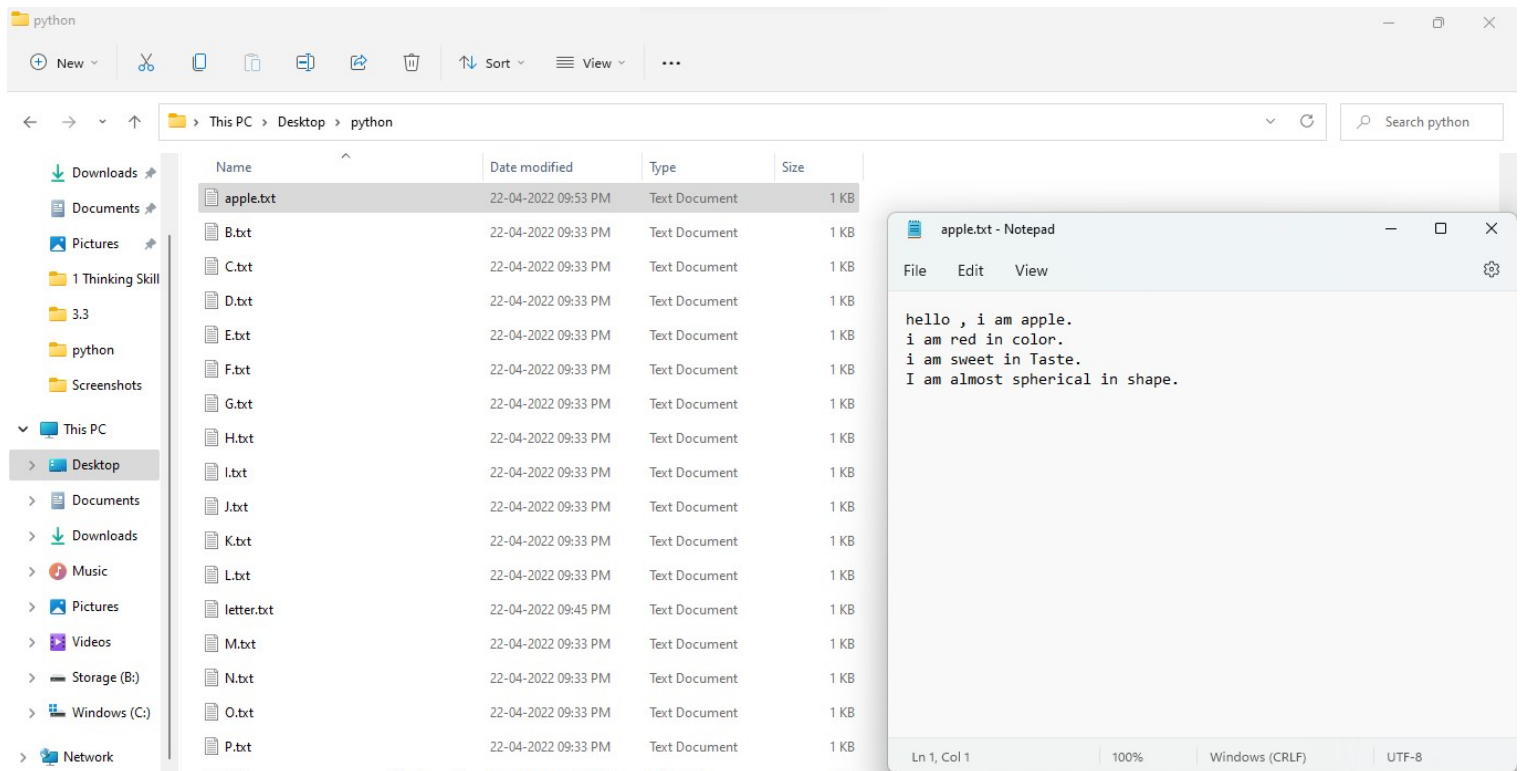
The screenshot shows a Jupyter Notebook interface with the following code and output:

```
In [1]: from collections import Counter
def word_count(fname):
    with open(fname) as f:
        return Counter(f.read().split())

print("Number of words in the file :",word_count("apple.txt"))
```

Number of words in the file : Counter({'am': 4, 'i': 3, 'in': 3, 'hello': 1, ',': 1, 'apple.': 1, 'red': 1, 'color.': 1, 'sweet': 1, 'Taste.': 1, 'I': 1, 'almost': 1, 'spherical': 1, 'shape.': 1})

apple.txt file-



The screenshot shows a file explorer window displaying the 'python' folder on the Desktop. The folder contains several text files, including 'apple.txt'. A Notepad window is open, showing the contents of 'apple.txt':

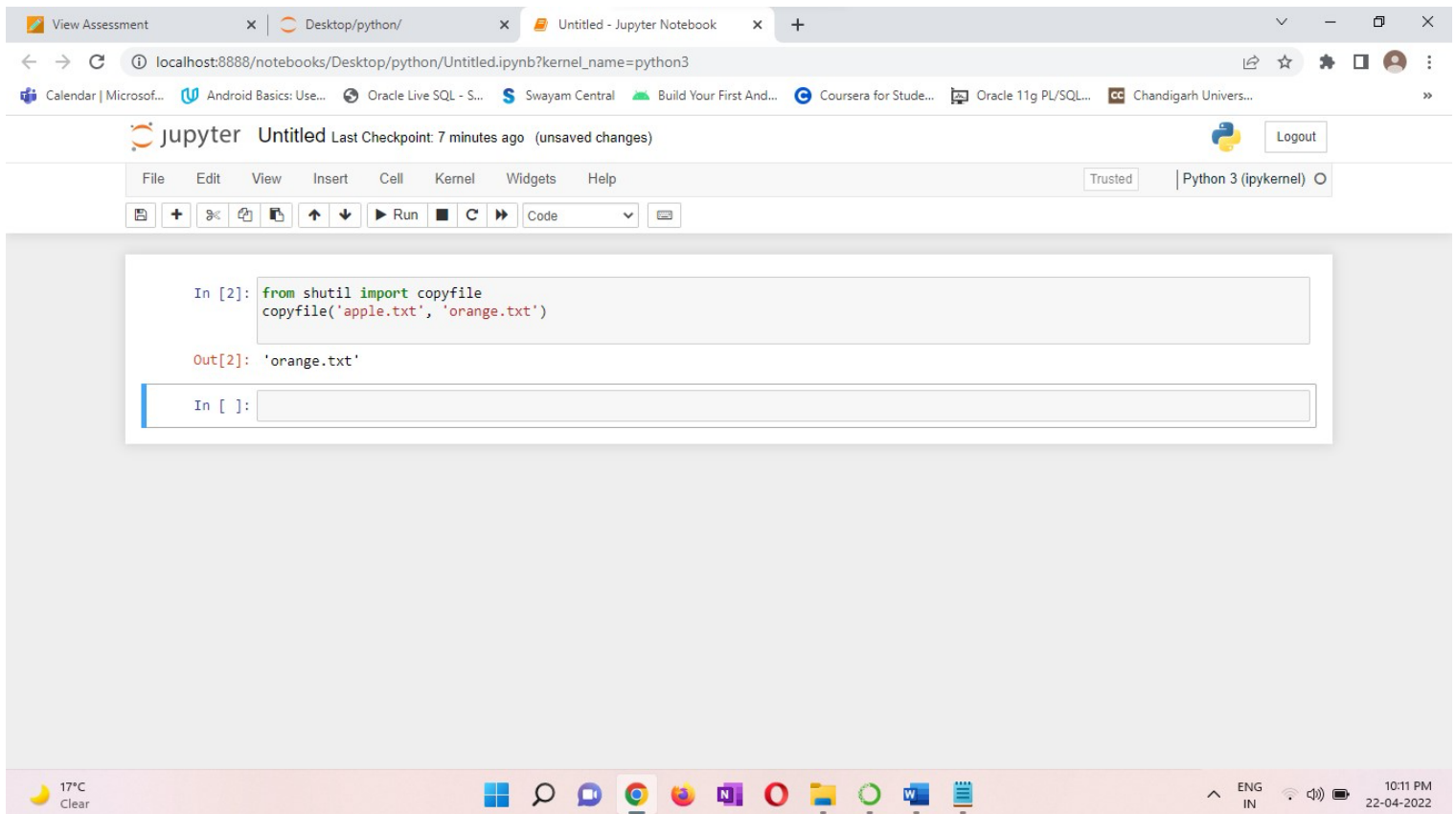
```
hello , i am apple.
i am red in color.
i am sweet in Taste.
I am almost spherical in shape.
```

5. Write a Python program to copy the contents of a file to another file

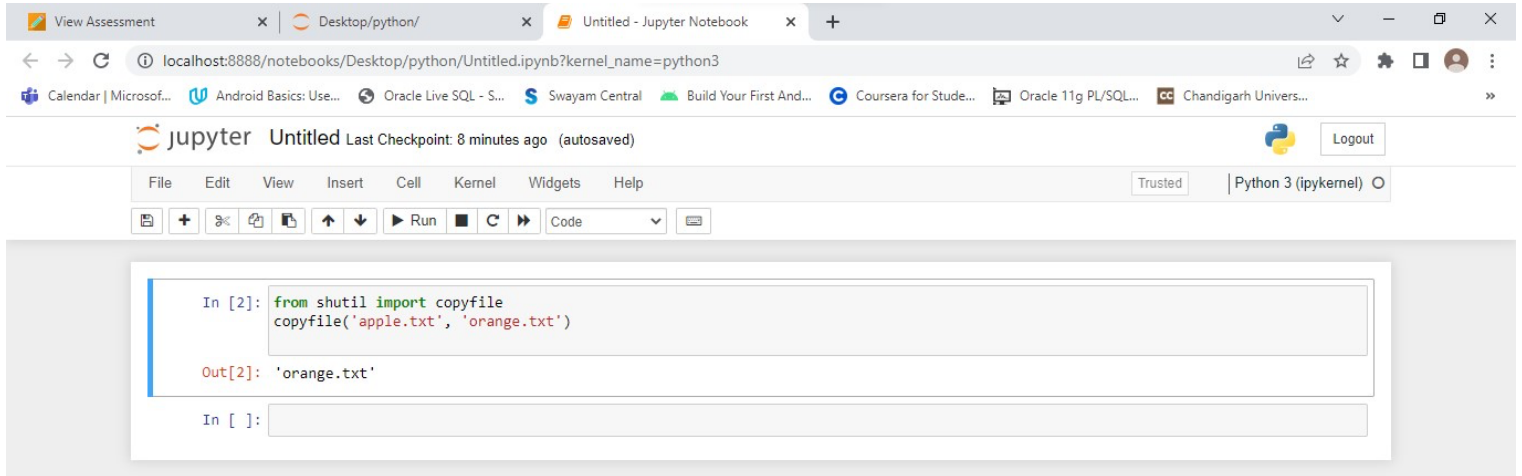
Program Code –

```
from shutil import copyfile  
  
copyfile('apple.txt', 'orange.txt')
```

Screenshot of Code–



Output-



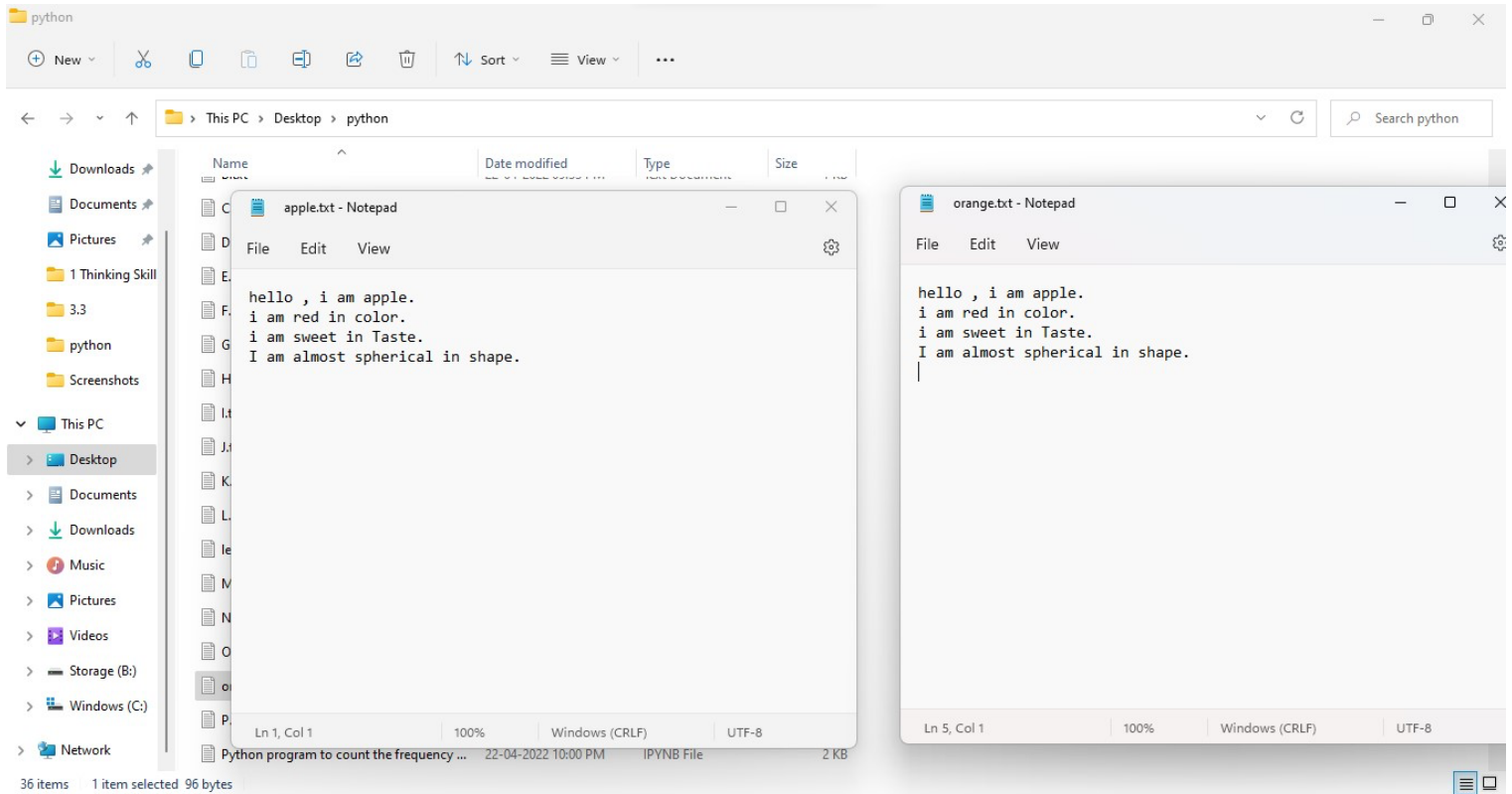
The screenshot shows a Jupyter Notebook interface in a web browser. The browser tabs include 'View Assessment', 'Desktop/python/', and 'Untitled - Jupyter Notebook'. The address bar shows 'localhost:8888/notebooks/Desktop/python/Untitled.ipynb?kernel_name=python3'. The Jupyter interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a code editor. The code in the cell is:

```
In [2]: from shutil import copyfile
        copyfile('apple.txt', 'orange.txt')
```

The output of the cell is:

```
Out[2]: 'orange.txt'
```

Below the output, there is an empty input prompt: 'In []:'.



Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
|---------|------------|----------------|---------------|
| 1. | | | |
| 2. | | | |
| 3. | | | |
| | | | |



DEPARTMENT OF ACADEMIC AFFAIRS

Discover. Learn. Empower.

NAAC
GRADE **A+**
ACCREDITED UNIVERSITY