

Project Title: Protecting a PDF Using Python

Objective:

Project is to create a Python tool that locks PDF files with a password. Users can select a PDF and set a password to protect it.

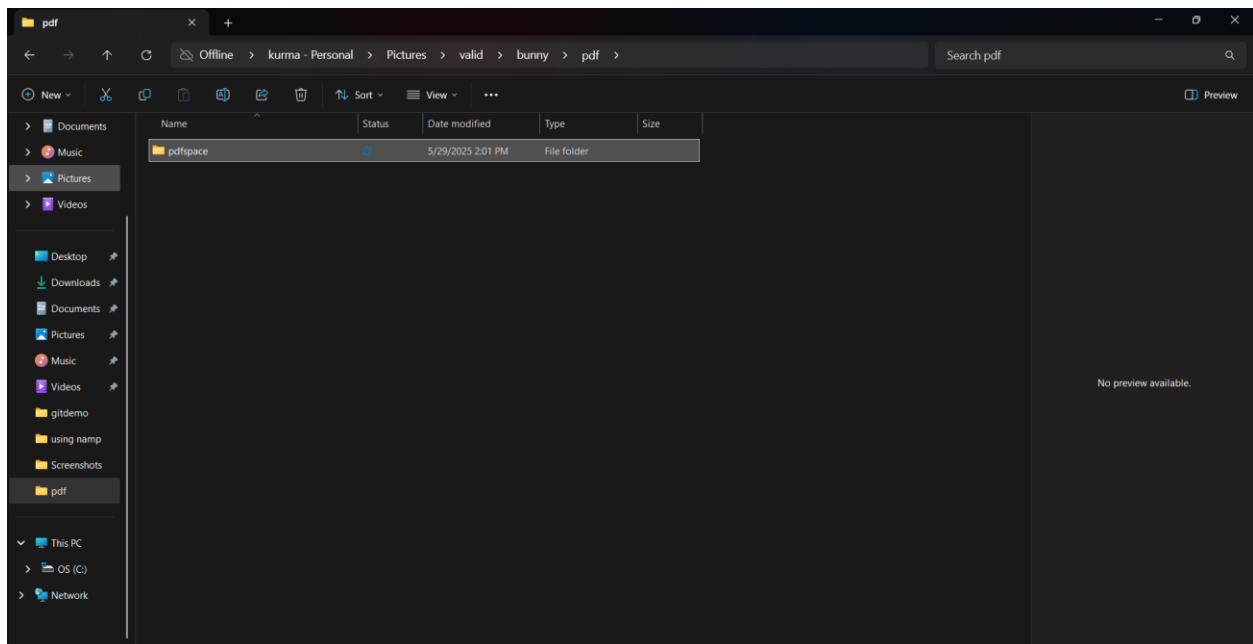
Project Overview:

This project is about protecting a PDF file using Python. Many times, we want to keep our PDF documents safe so that only people with the password can open them. In this project, we created a small Python program that adds a password to any PDF file.

Step-by-Step Procedure:

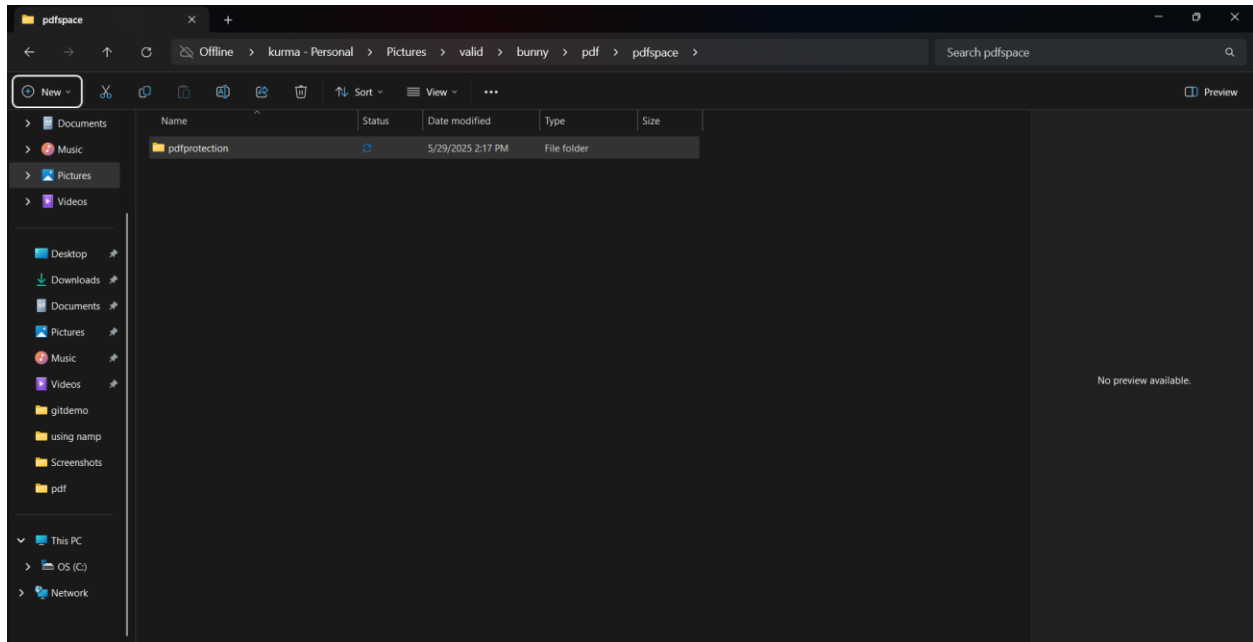
1. Create a New Folder:

Create a folder on your computer and name it **pdfspace**. This folder will store all the files related to this project



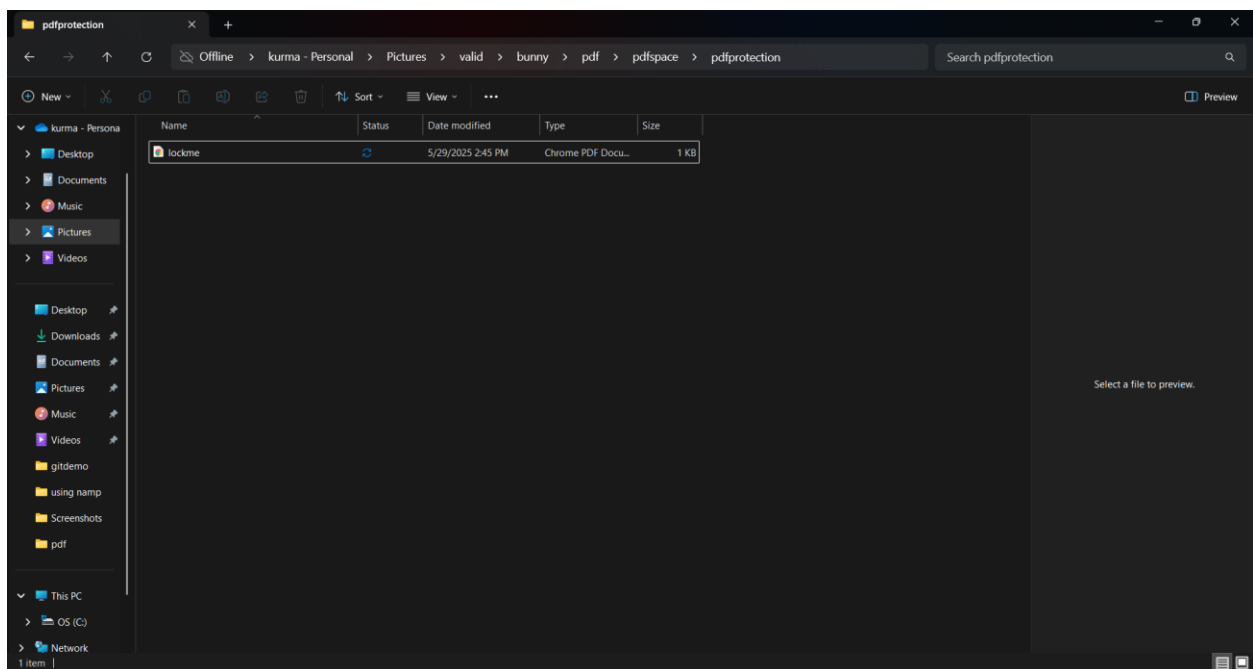
2. Create a New File:

Inside the **pdfspace** folder, create a new file and name it **pdfprotection**. This will be your working area.



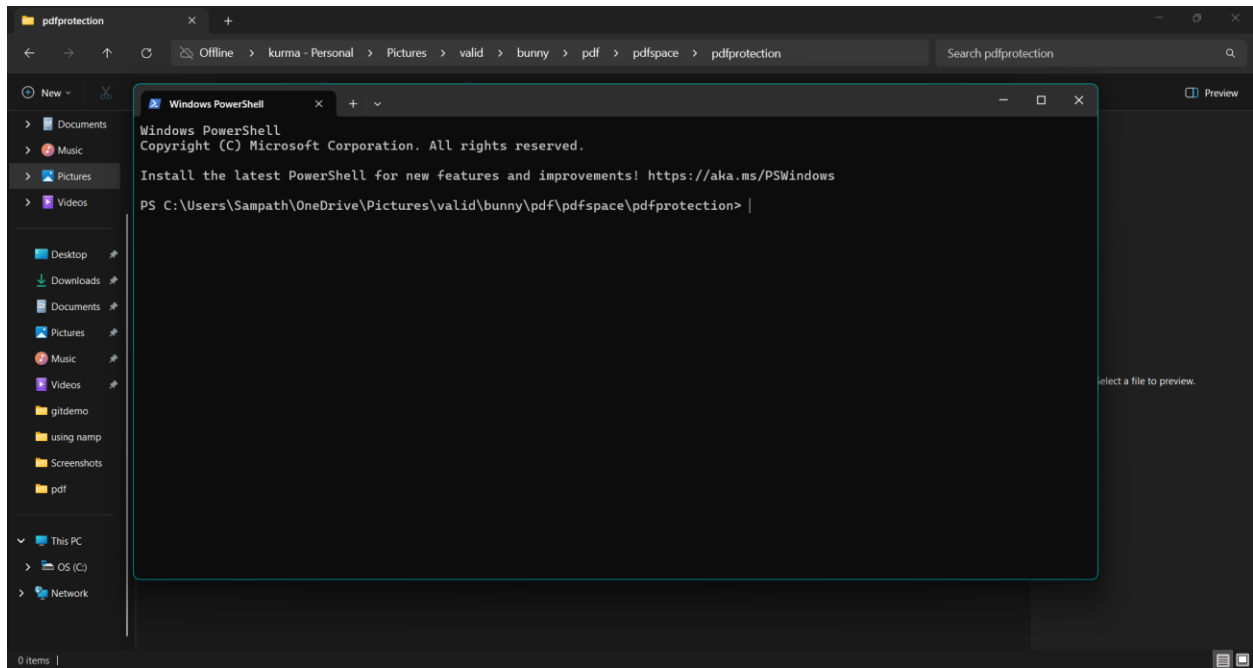
3. Open the File:

Now, open the **pdfprotection** file and add unlocked pdf



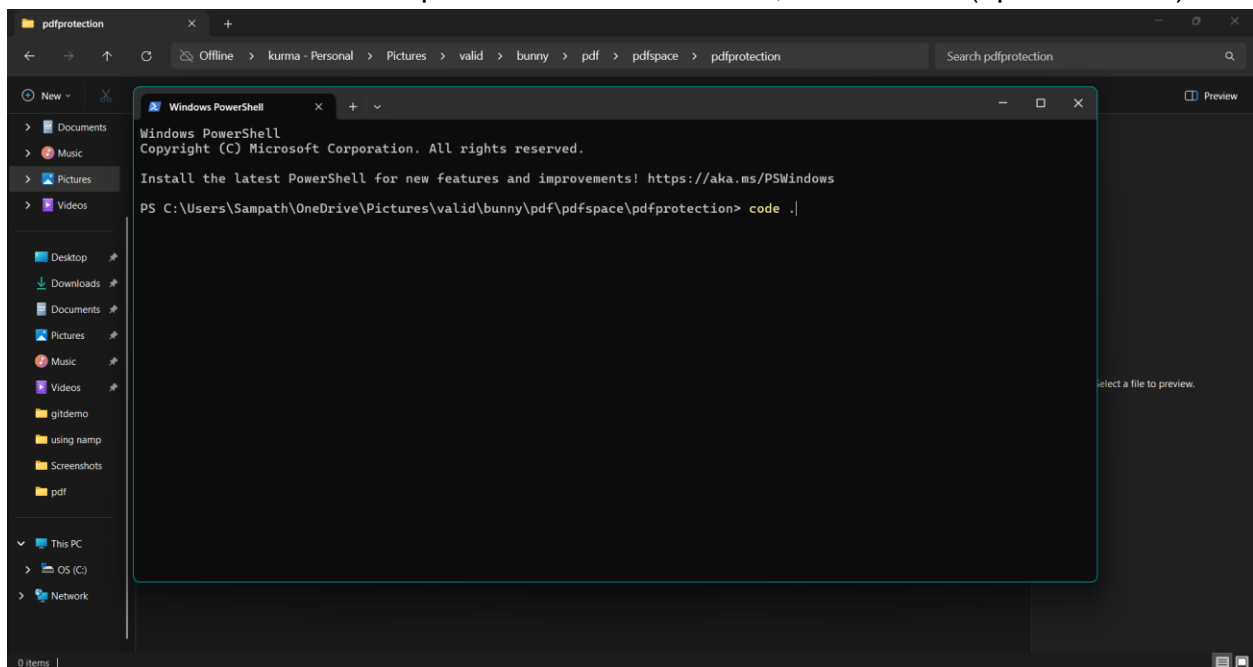
4. Open Terminal:

Right-click anywhere inside the file or folder area and select "Open in Terminal". This will launch a terminal window linked to your current folder. You will now see a terminal open at the bottom or side



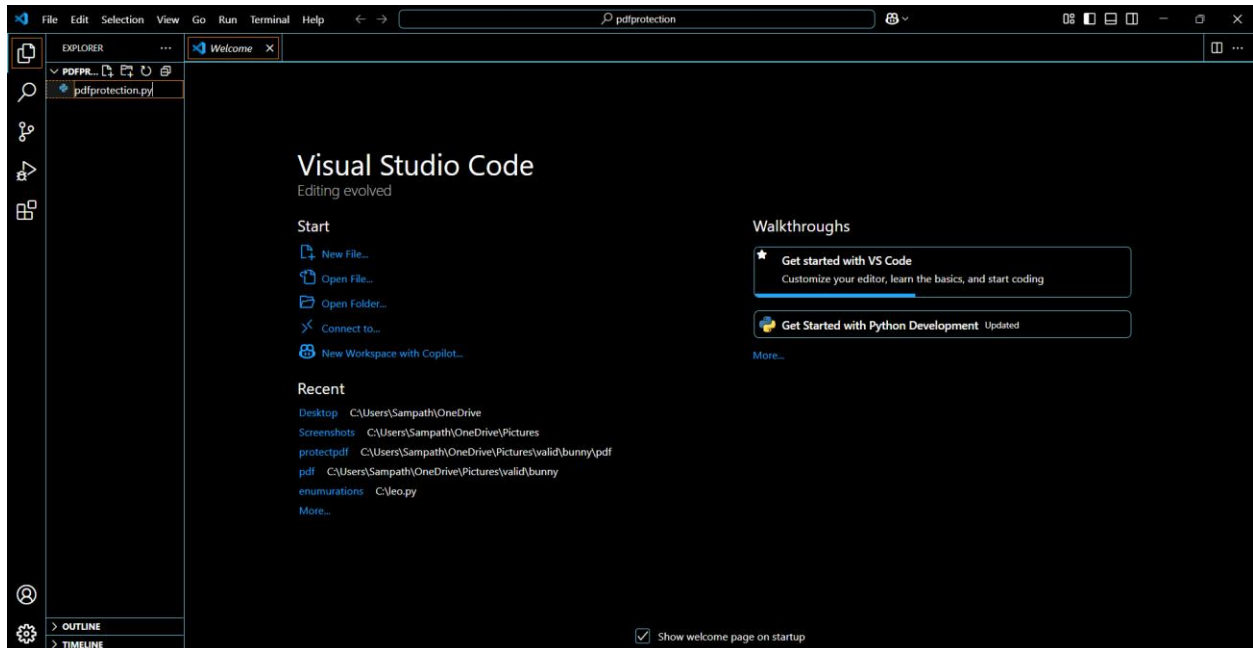
5. Terminal is Opened:

You will now see a terminal open at the bottom or side, enter **code .**(opens vscode)

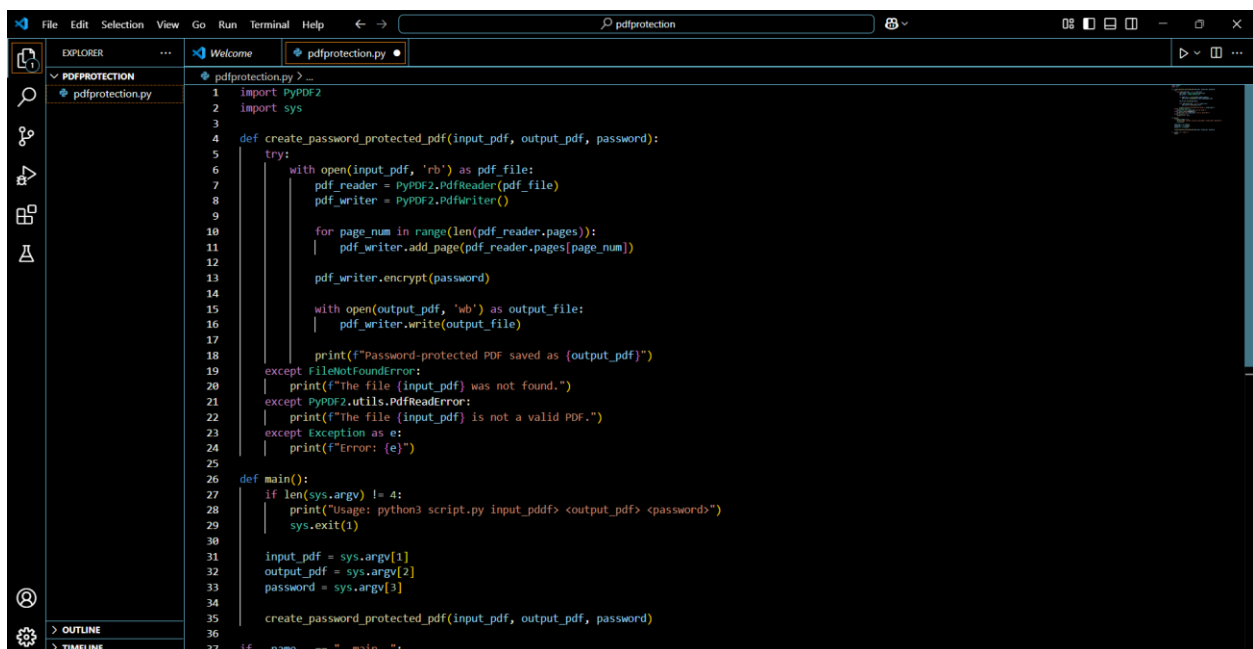


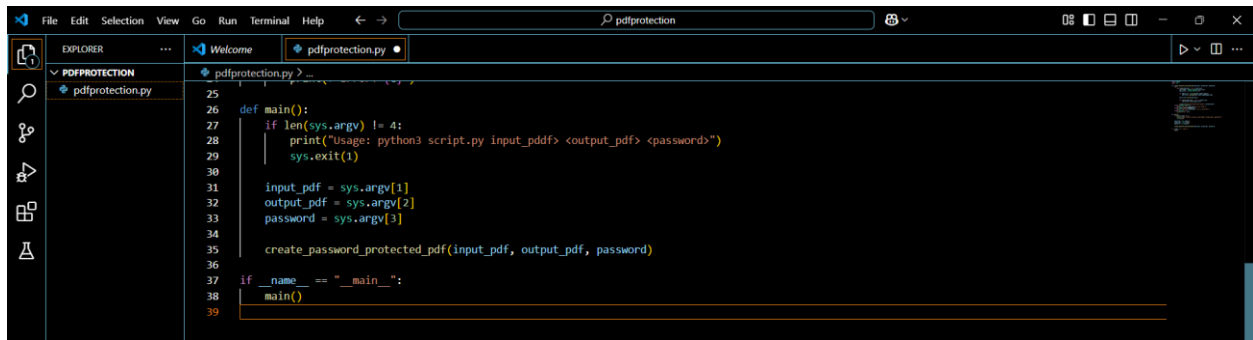
6. Create a Python File:

At the top-left of your code editor, create a new Python file and name it pdfprotection.py. This is where you'll write your Python code to protect the PDF.



7. Write the Code:





CODE:

```
import PyPDF2
import sys
```

```
def create_password_protected_pdf(input_pdf, output_pdf, password):
    try:
```

```
        with open(input_pdf, 'rb') as pdf_file:
            pdf_reader = PyPDF2.PdfReader(pdf_file)
            pdf_writer = PyPDF2.PdfWriter()
```

```
            for page_num in range(len(pdf_reader.pages)):
                pdf_writer.add_page(pdf_reader.pages[page_num])
```

```
            pdf_writer.encrypt(password)
```

```
            with open(output_pdf, 'wb') as output_file:
                pdf_writer.write(output_file)
```

```
            print(f"Password-protected PDF saved as {output_pdf}")
```

```
    except FileNotFoundError:
```

```
        print(f"The file {input_pdf} was not found.")
```

```
    except PyPDF2.utils.PdfReadError:
```

```
        print(f"The file {input_pdf} is not a valid PDF.")
```

```
    except Exception as e:
```

```
        print(f"Error: {e}")
```

```
def main():
```

```
    if len(sys.argv) != 4:
```

```
        print("Usage: python3 script.py input_pdf <output_pdf> <password>")
        sys.exit(1)
```

```
    input_pdf = sys.argv[1]
```

```
    output_pdf = sys.argv[2]
```

```
    password = sys.argv[3]
```

```
    create_password_protected_pdf(input_pdf, output_pdf, password)
```

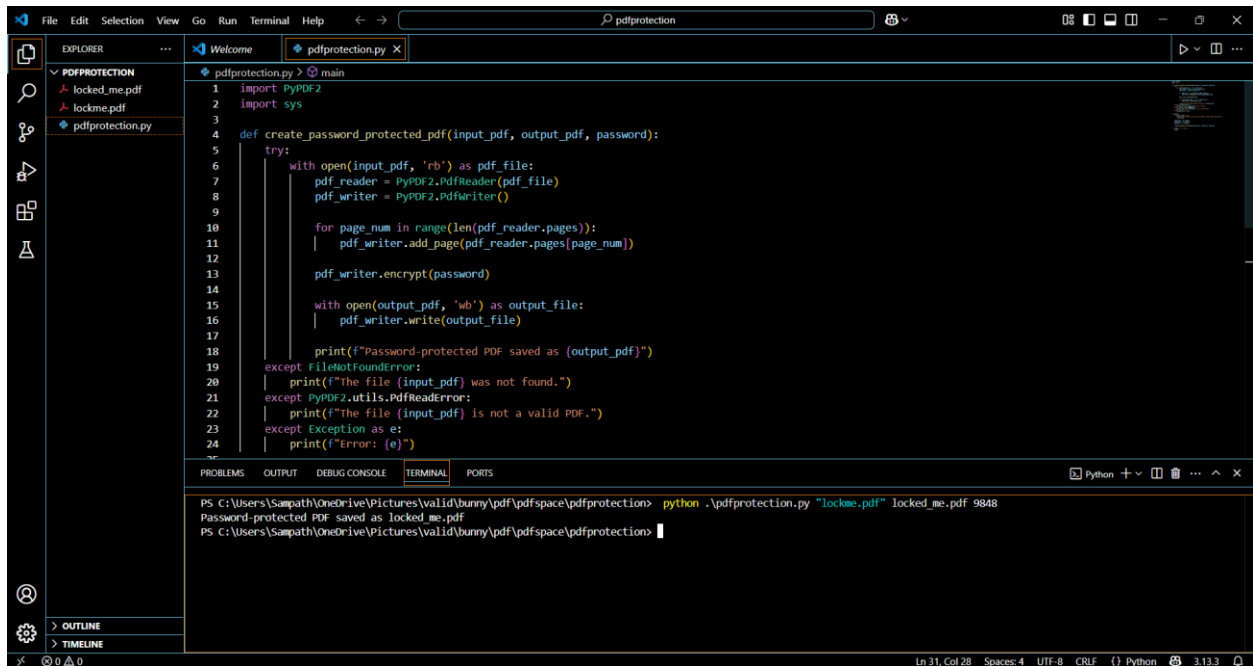
```
if __name__ == "__main__":
```

```
    main()
```

8. Run the Code in Terminal:

After writing the code, go to the terminal at the bottom and type the following command:

`python .\pdfprotection.py "lockme.pdf" locked_me.pdf 9848` and click enter.



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows a project named 'PDFPROTECTION' containing files 'locked_me.pdf', 'lockme.pdf', and 'pdfprotection.py'. The main editor displays the code for 'pdfprotection.py', which defines a function 'create_password_protected_pdf' that takes an input PDF, an output PDF, and a password, and returns a password-protected PDF. The terminal at the bottom shows the command 'python .\pdfprotection.py "lockme.pdf" locked_me.pdf 9848' being executed, followed by the output 'Password-protected PDF saved as locked_me.pdf'.

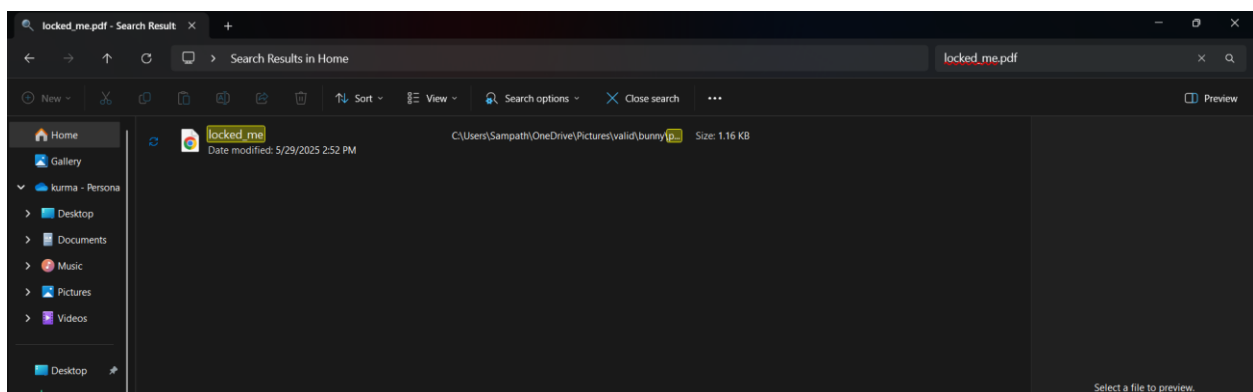
```
1 import PyPDF2
2 import sys
3
4 def create_password_protected_pdf(input_pdf, output_pdf, password):
5     try:
6         with open(input_pdf, 'rb') as pdf_file:
7             pdf_reader = PyPDF2.PdfReader(pdf_file)
8             pdf_writer = PyPDF2.PdfWriter()
9
10            for page_num in range(len(pdf_reader.pages)):
11                pdf_writer.add_page(pdf_reader.pages[page_num])
12
13            pdf_writer.encrypt(password)
14
15            with open(output_pdf, 'wb') as output_file:
16                pdf_writer.write(output_file)
17
18            print(f"Password-protected PDF saved as {output_pdf}")
19    except FileNotFoundError:
20        print(f"The file {input_pdf} was not found.")
21    except PyPDF2.utils.PdfReadError:
22        print(f"The file {input_pdf} is not a valid PDF.")
23    except Exception as e:
24        print(f"Error: {e}")
```

PS C:\Users\Sampath\OneDrive\Pictures\valid\bunny\pdf\pdfspace\pdfprotection> python .\pdfprotection.py "lockme.pdf" locked_me.pdf 9848
Password-protected PDF saved as locked_me.pdf
PS C:\Users\Sampath\OneDrive\Pictures\valid\bunny\pdf\pdfspace\pdfprotection>

In the terminal, the password-protected PDF was saved as 'locked_me.pdf'.

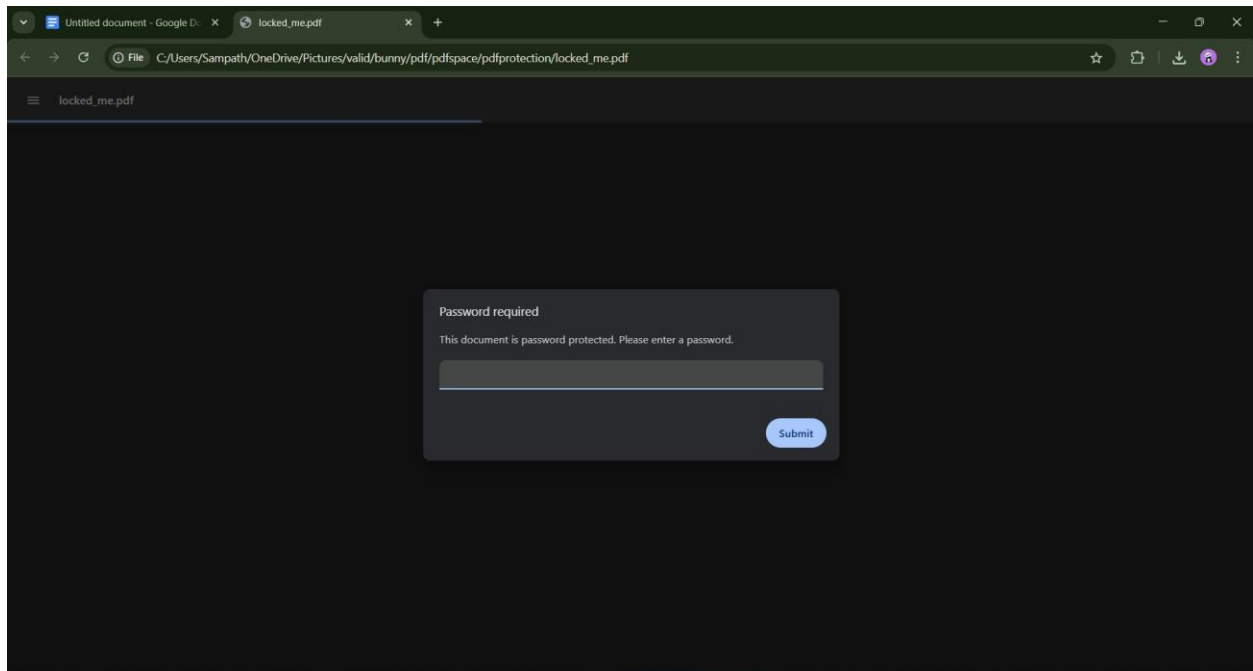
9. Check the PDF (locked_me.pdf).

Open the files and search for 'locked_me.pdf'



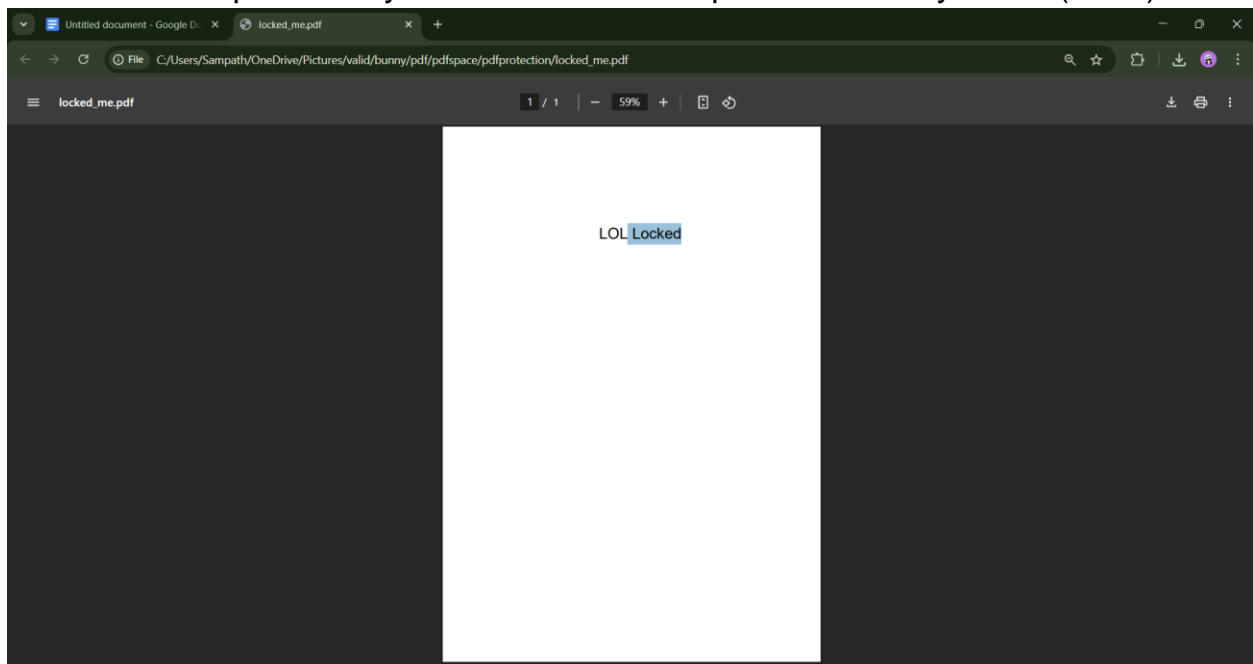
10. Open PDF

When you click to open the PDF, it will prompt you to enter a password.



11. Unlocking the PDF:

The PDF will open once you enter the correct password that you set (9848).



Conclusion:

In this project, I created a small Python tool to lock PDF files with a password. I used the PyPDF2 library to read the PDF, add the password, and save the new protected file. This helps keep the document safe so that only people who know the password can open it. Through this task, I learned how to work with files in Python and how to add basic security to them. It was a useful and interesting experience that improved my coding and problem-solving skills.

THANK YOU

Name: Kurma sampath

Intern ID: ITID0672