Project Report on
Python Project
WEEK - 5

TEAM MEMBER (SELF)
Name : SONU KUMAR RAY
Phone No : +91 9967901030
Email-id : sonukumargarha8472@gmail.com

PASSWORD MANAGER

Description: The password manager is a Python project that securely stores and manages user passwords. It allows users to store their passwords for various accounts, generate strong passwords, and retrieve passwords when needed.

Scope: The scope of this project involves implementing encryption algorithms to secure password storage, designing a user interface to input and retrieve passwords, and developing functions to generate strong passwords and store/retrieve them from a database.

A password manager is a software application that helps users generate, store, and manage their passwords securely. It can be a useful tool for managing project content and ensuring that sensitive information is protected. Here are some key points about password managers for project content:

Following is the things should be in that project

1.Password Generation

2.Secure Storage

3.Auto-Fill and Auto-Login

4.Cross-Platform Accessibility:

5.Secure Sharing

6.Password Auditing:

7.Two-Factor Authentication (2FA)

8.Backup and Sync

Overall, using a password manager for project content can help you maintain strong, unique passwords, enhance security, and streamline access to your project accounts and resources. It is important to choose a reputable password manager with robust security measures and regularly update your master password to maintain the integrity of your project content.

Program Code of the Project is Below :

```
from cryptography.fernet import Fernet

class passwordManager:
        def __init__(self):
                self.key=None
                self.password_file=None
                self.password_dict = {}
```

```
            def create_key(self , path):
                        self.key = Fernet.generate_key()
                        with open(path , 'wb') as f:
                                    f.write(self.key)

            def load_key(self , path):
                        with open(path, 'rb') as f:
                                    self.key=f.read()

            def create_password_file(self , path , initial_values=None):
                        self.password_file = path

        if initial_values is not None:
                                    for key, value in initial_values.items():
                                                pass
    def load_password_file(self , path):
        self.password_file = path
        with open(path ,'r') as f:
                for line in f:
                            site,encrypted = line.split(":")

self.password_dict[site]=Fernet(self.key).decrypt(encrypted.encode()).decode()
 """   def load_password_file(self ,path):
        self.password_file = path

        with open(path, 'r') as f:
            for line in f:
                site , encrypted = line.split(":")"""


    def add_password(self ,site ,password):
            self.password_dict[site]=password_dict

            if self.password_file is not None:
                        with open(self.password_file , 'a+') as f:
                                    encrypted = Fernet(self.key).encrypt(password.encode())
                                    f.write(site + ":" + encrypted.decode()+"\n")

    def get_password(self ,site):
            return self.password_dict[site]


    def main():
```

```
  password ={
   "email":"sonu@gmail.com",
   "facebook":"facebook",
   "instagram":"instagram"
  }

  pm=passwordManager()


  print("""What do you want to select ?
(1) Create a new Key
(2) Load an Existing key
(3) Create new password file
(4) Load existing password file
(5) Add a new password
(6) get a password
(7) Quit
""")

  done = False

  while not done:

          choice=input("Enter Your Choice: ")
          if(choice)=="1":
                  path = input("Enter path: ")
                  pm.create_key(path)
          elif choice=="2":
                  path =input("Enter path:")
                  pm.load_key(path)
          elif choice =="3":
                  path = input("Enter path: ")
                  pm.create_password_file(path,password)
          elif choice =="4":
                  path = input("Enter path: ")
                  pm.load_password_file(path)
          elif choice =="5":
                  site = input("Enter the site: ")
                  password= input("Enter the password: ")
                  pm.add_password(site , password)

          elif choice=="6":
                  site =input("what site do you want: ")
                  print(f"Password for {site} is {pm.get_password(site)}")
```

```
        elif choice=="q":
            done =True
            print("Bye")
        else:
            print("Invalid choice!")


if __name__=="__main__":
    main()
```

SONU KUMAR RAY

Candidate Signature