Darshan UNIVERSITY
योगः कर्मसु कौशलम्

(https://www.darshan.ac.in/)

Python Programming - 2101CS405

Lab - 8

File handling

A

01) WAP to read entire file named abc.txt

In [2]:
```
f = open('hello.txt','r')
print(f.read())
```

```
Hello Good Morning
```

02) WAP to print program it self on console.

In [2]:
```
with open(__file__, 'r') as file:
    content = file.read()

print(content)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[2], line 1
----> 1 with open(__file__, 'r') as file:
      2     content = file.read()
      4 print(content)

NameError: name '__file__' is not defined
```

03) WAP to read first 5 lines from the file named abc.txt

In [4]:
```
f = open('hello.txt','r')
for i in range(0,5):
    print(f.readline())
```

```
Hello Good Morning

Hi

How Are you?

Akshat

Preyarsh
```

04) WAP to find the longest word in a file named abc.txt

In [6]:
```
f = open('hello.txt','r')
ans = ""
data = f.readlines()
for line in data:
    word = line.split()
    for i in word:
        if len(ans)<len(i):
            ans=i
print(ans)
```

```
University
```

05) WAP to find the size of the file named abc.txt

In [7]:
```
f = open('hello.txt','r')
lines = f.read()
ans = 0
for i in lines:
    ans+=1
else:
    print(ans)
```

```
139
```

06) WAP to implement search function to search specific occurance of word in a given text file.

In [13]:
```
f = open("hello.txt", "r")
count = 0
word = input("Enter Word For Search : ")
for line in f.readlines():
    for words in line.lower().split(" "):
        if words == word or word+"\n" == words:
            count += 1
if count == 0:
    print("word not found")
else:
    print(count, "occurence")
f.close()
```

```
Enter Word For Search : preyarsh
2 occurence
```

B

01) WAP to write first 100 prime numbers to a file named primenumbers.txt

(Note: each number should be in new line)

In [28]:
```
primes = []
temp = 1
num = -1
while temp<=100:
    is_prime = True
    num+=1
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            is_prime = False
            break
    if is_prime:
        temp+=1
        primes.append(str(num) + "\n")

with open("prime.txt", "w") as fp:
    fp.writelines(primes)
```

02) WAP to merge two files and write it in a new file.

In [35]:
```
f1 = open('hello.txt','r')
data = f1.read()
data += "\n"
f2 = open('prime.txt','r')
data += f2.read()
f3 = open("ans.txt","w+")
f3.write(data)
f3.close()
f = open("ans.txt",'r')
print(f.read())
```

```
419
421
431
433
439
443
449
457
461
463
467
479
487
491
499
503
509
521
```

03) WAP to encrypt a text file.

In [9]:
```
from cryptography.fernet import Fernet

key = Fernet.generate_key()
fernet = Fernet(key)

f = open("ans.txt", "r")
data = f.read()

encMessage = fernet.encrypt(data.encode())

file = open("ans.txt", "wb")
file.write(encMessage)
```

Out[9]: 100

04) WAP to decrypt a previously encrypted file.

In [10]:
```
from cryptography.fernet import Fernet

key = Fernet.generate_key()
fernet = Fernet(key)

# Read the content of the file
file = open("ans.txt", "rb")
data = file.read()

decMessage = fernet.decrypt(data).decode()

print("Decrypted:", decMessage)
```

```
---------------------------------------------------------------------------
InvalidSignature                          Traceback (most recent call last)
File ~\anaconda3\Lib\site-packages\cryptography\fernet.py:134, in Fernet._ver
ify_signature(self, data)
    133 try:
--> 134     h.verify(data[-32:])
    135 except InvalidSignature:

InvalidSignature: Signature did not match digest.

During handling of the above exception, another exception occurred:

InvalidToken                              Traceback (most recent call last)
Cell In[10], line 10
      7 file = open("ans.txt", "rb")
      8 data = file.read()
---> 10 decMessage = fernet.decrypt(data).decode()
     12 print("Decrypted:", decMessage)

File ~\anaconda3\Lib\site-packages\cryptography\fernet.py:91, in Fernet.decry
pt(self, token, ttl)
     89 else:
     90     time_info = (ttl, int(time.time()))
---> 91 return self._decrypt_data(data, timestamp, time_info)

File ~\anaconda3\Lib\site-packages\cryptography\fernet.py:152, in Fernet._dec
rypt_data(self, data, timestamp, time_info)
    149     if current_time + _MAX_CLOCK_SKEW < timestamp:
    150         raise InvalidToken
--> 152 self._verify_signature(data)
    154 iv = data[9:25]
    155 ciphertext = data[25:-32]

File ~\anaconda3\Lib\site-packages\cryptography\fernet.py:136, in Fernet._ver
ify_signature(self, data)
    134     h.verify(data[-32:])
    135 except InvalidSignature:
--> 136     raise InvalidToken

InvalidToken:
```

05) WAP to remove a word from text file.

In [ ]: