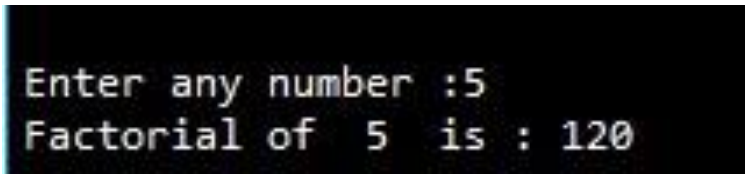
Program 1 : Input any number from user and calculate factorial of a number

```
num = int(input("Enter any number :"))
fact = 1
n = num
while num>1:
   fact = fact * num
   num-=1
print("Factorial of ", n , " is :",fact)
```

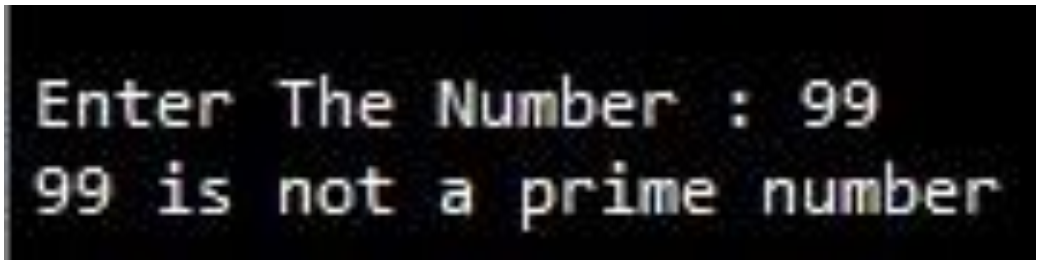
OUTPUT:

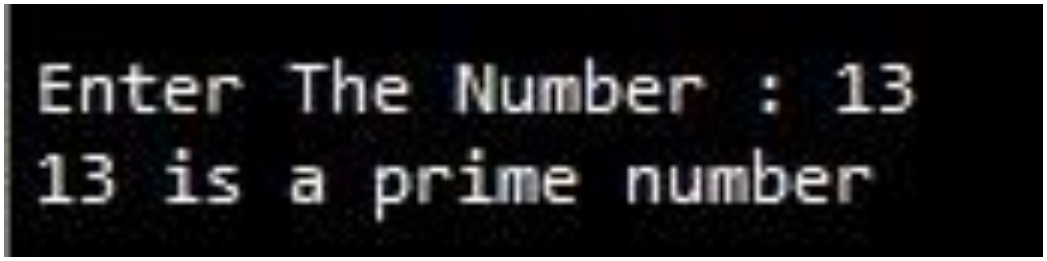


Program 2 : Input any number from user and check it is Prime no. or not.

```
def prime(x):
   if x > 1:
      for i in range(2,x):
        if (x % i) == 0:
           print(x,"is not a prime number")
           break
      else:
        print(x,"is a prime number")


   else:
      print(x,"is not a prime number")
n= int(input("Enter The Number : "))
prime(n)
```

OUTPUT:





Program 3 : Program to search any word in given string/sentence.

```
def countWrd(strg,word):
    s = strg.split()
    count=0
    for w in s:
        if w==word:
            count+=1
    return count

strg = str(input("Enter The String : "))
word = str(input("Enter The Word : "))

a = strg.find(word)
l=len(word)
f = strg[a:a+l]
count=countWrd(strg,word)
if f == word:
    print("",word," occurs ",count," times...")
else:
    print("Word Not Found")
```

OUTPUT:

```
Enter The String : I Love Python

Enter The Word : Python
 Python  occurs  1  times...
```

```
Enter The String : I Love Python

Enter The Word : python
Word Not Found
```

Program 4 : Program to read and display file content line by line with each word separated by "#".

```
f = open("poem.txt")
for line in f:
   print(line)
   words = line.split()
   for w in words:
      print(w+'#',end='')
   print()
f.close()


'''f = open("poem.txt")
fcont=f.read()
words = fcont.split()
for w in words:
  print(w+'#',end='')
f.close()'''
```

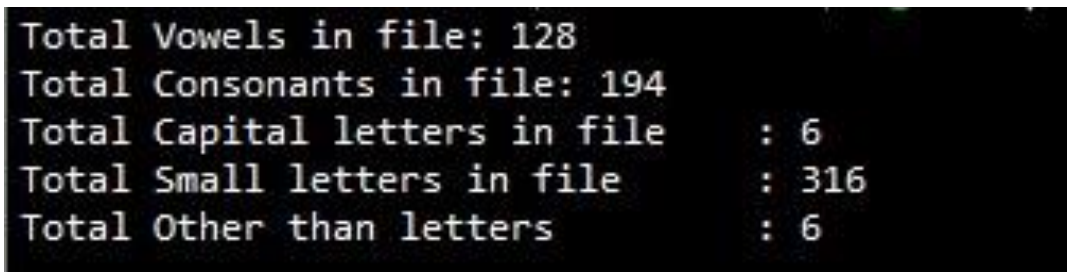
OUTPUT:

```
Difficult because we think that
happiness is found

Difficult#because#we#think#that#happin
ess#is#found#
Only in the places where wealth and
fame abound.

Only#in#the#places#where#wealth#and#fa
me#abound.#
And so we go on searching in places of
pleasure

And#so#we#go#on#searching#in#places#of
#pleasure#
seeking recognition and monetary
treasure,

seeking#recognition#and#monetary#treas
ure,#
Unaware that happiness is just a state
of mind

Unaware#that#happiness#is#just#a#state
#of#mind#
within the reach of everyone who takes
time to be kind,

within#the#reach#of#everyone#who#takes
#time#to#be#kind,#
For in making others happy we will be
happy, too.

For#in#making#others#happy#we#will#be#
happy,#too.#
For the happiness you give away
returns to shine on you.
For#the#happiness#you#give#away#return
s#to#shine#on#you.#
```

Program 5 : Program to read the content of file and display the total number of consonants, uppercase characters, vowels and lower case characters.

```
f = open("poem.txt")
v=c=u=l=o=0
data = f.read()
vowels=['a','e','i','o','u']
for ch in data:
    if ch.isalpha():
        if ch.lower() in vowels:
            v+=1
        else:
            c+=1
    if ch.isupper():
        u+=1
    elif ch.islower():
        l+=1
    elif ch!=' ' and ch!='\n':
        o+=1
print("Total Vowels in file :",v)
print("Total Consonants in file :",c)
print("Total Capital letters in file :",u)
print("Total Small letters in file :",l)
print("Total Other than letters :",o)
f.close()
```

OUTPUT:



Program 6 : Program to create binary file to store Rollno and Name, Search any Rollno and display name if Rollno is found otherwise display a message "Rollno not found".

```
import pickle
student=[]
f=open('student.dat','wb')
ans='y'
while ans.lower()=='y':
   roll = int(input("Enter Roll Number :"))
   name = input("Enter Name :")
   student.append([roll,name])
   ans=input("Add More ?(Y)")
pickle.dump(student,f)
f.close()

f=open('student.dat','rb')
student=[]
while True:
   try:
      student = pickle.load(f)
   except EOFError:
      break
ans='y'
while ans.lower()=='y':
   found=False
   r = int(input("Enter Roll number to search :"))
   for s in student:
      if s[0]==r:
         print("## Name is :",s[1], " ##")
         found=True
         break
   if not found:
      print("####Sorry! Roll number not found ####")
   ans=input("Search more ?(Y) :")
f.close()
```

OUTPUT:

Adding Record:

```
Enter Roll Number :1

Enter Name :Kirtan

Add More ?(Y)y

Enter Roll Number :2

Enter Name :KP

Add More ?(Y)
```

Searching Record:

```
Enter Roll number to search :2
## Name is : KP  ##

Search more ?(Y) :y

Enter Roll number to search :1
## Name is : Kirtan  ##

Search more ?(Y) :
```

Program 7 : Program to create binary file to store Roll no, Name and Marks and update marks of entered Roll no.

```
import pickle
student=[]
f=open('student.dat','wb')
ans='y'
while ans.lower()=='y':
   roll = int(input("Enter Roll Number :"))
   name = input("Enter Name :")
   marks = int(input("Enter Marks :"))
   student.append([roll,name,marks])
   ans=input("Add More ?(Y)")
pickle.dump(student,f)
f.close()

f=open('student.dat','rb')
student=[]
while True:
   try:
      student = pickle.load(f)
   except EOFError:
      break
f.close()

ans='y'
while ans.lower()=='y':
   f=open("student.dat","wb")
   rec=[]
   found=False
   r = int(input("Enter Roll number to update :"))
   for s in student:
      if s[0]==r:
         print("## Name is :",s[1], " ##")
         print("## Current Marks is :",s[2]," ##")
         m = int(input("Enter new marks :"))
```

```
            s[2]=m
            found=True
        rec.append(s)
    pickle.dump(rec,f)
    if found==True:
        print("Record updated")
    else:
        print("####Sorry! Roll number not found ####")
        f.close()
    ans=input("Update more ?(Y) :")

print("Displaying updated file")
f=open("student.dat","rb")
rec=[]
try:
    rec=pickle.load(f)
    for i in rec:
        print(i)
except:
    print("Error")
f.close()
```

OUTPUT:

Creating Record:

```
Enter Roll Number :1

Enter Name :Kirtan

Enter Marks :410

Add More ?(Y)y

Enter Roll Number :2

Enter Name :KP

Enter Marks :400
```

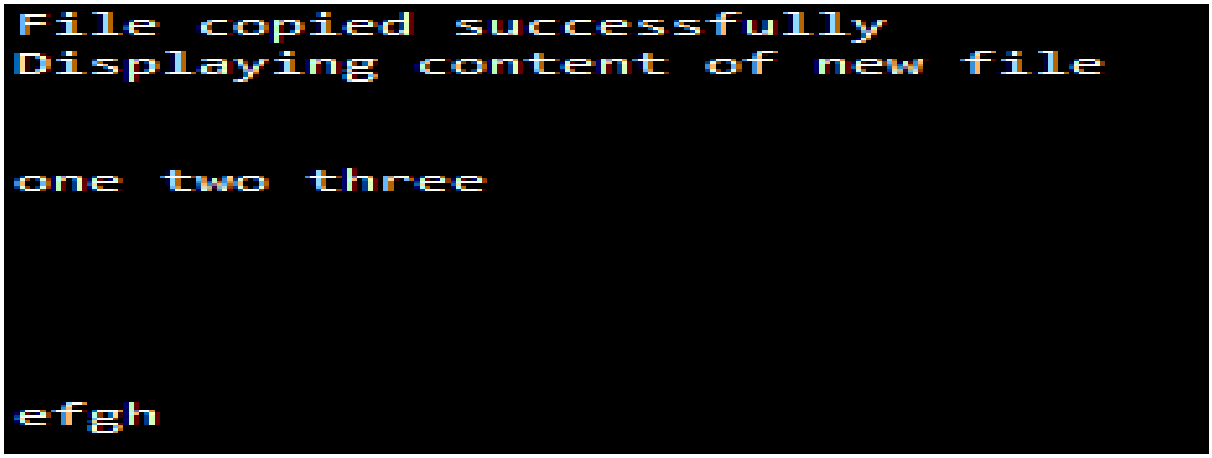
Updating Record:

```
Enter Roll number to update :1
## Name is : Kirtan  ##
## Current Marks is : 410  ##

Enter new marks :400
Record updated

Update more ?(Y) :
Displaying updated file
[1, 'Kirtan', 400]
[2, 'KP', 400]
```

Program 8 : Program to read the content of file line by line and write it to another file except for the lines contains "a" letter in it.

```
f1=open("file1.txt")

f2=open("file2.txt","w")

for line in f1 :
   if 'a' not in line:
      f2.write(line)

print("File copied successfully")
f1.close()
f2.close()

print("Displaying content of new file")
f1=open("file2.txt")
for i in f1:
   print(i)

f1.close()
```

OUTPUT:

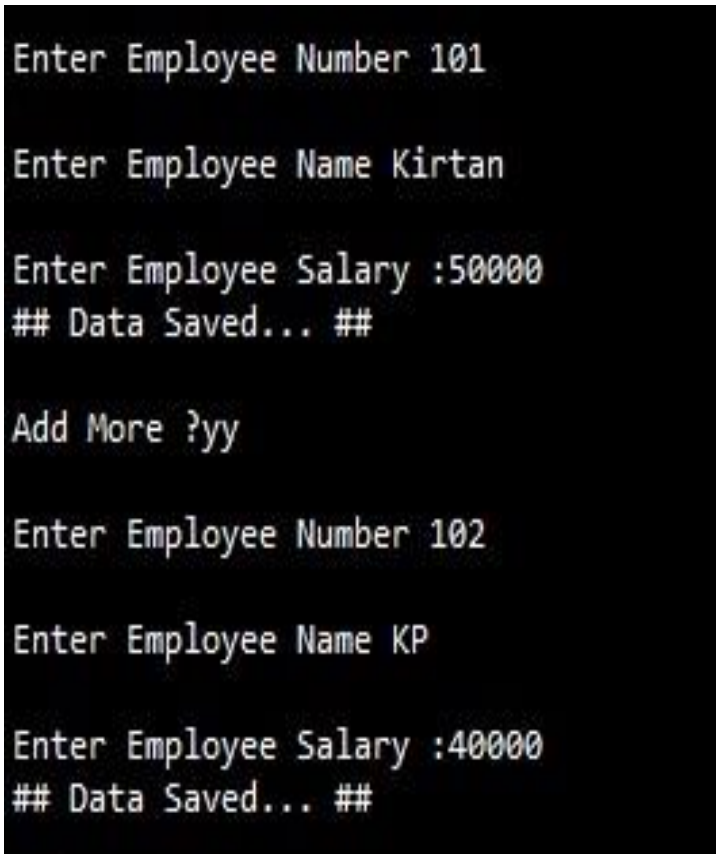


Program 9 : Program to create CSV file and store empno, name, salary and search any empno and display name, salary and if not found appropriate message.

```
import csv
with open('myfile.csv',mode='a') as csvfile:
   mywriter = csv.writer(csvfile,delimiter=',')
   ans='y'
   while ans.lower()=='y':
      eno=int(input("Enter Employee Number "))
      name=input("Enter Employee Name ")
      salary=int(input("Enter Employee Salary :"))
      mywriter.writerow([eno,name,salary])
      print("## Data Saved... ##")
      ans=input("Add More ?y")
csvfile.close()

ans='y'
with open('myfile.csv',mode='r') as csvfile:
   myreader = csv.reader(csvfile,delimiter=',')
   while ans=='y':
      found=False
      e = int(input("Enter Employee Number to search :"))
      for row in myreader:
         if len(row)!=0:
            if int(row[0])==e:
               print("============================")
               print("NAME  :",row[1])
               print("SALARY :",row[2])
               found=True
               break
      if not found:
         print("=========================")
         print("     EMPNO NOT FOUND")
         print("=========================")
      ans = input("Search More ? (Y)")
```

OUTPUT:

Adding Record:



Searching Record:

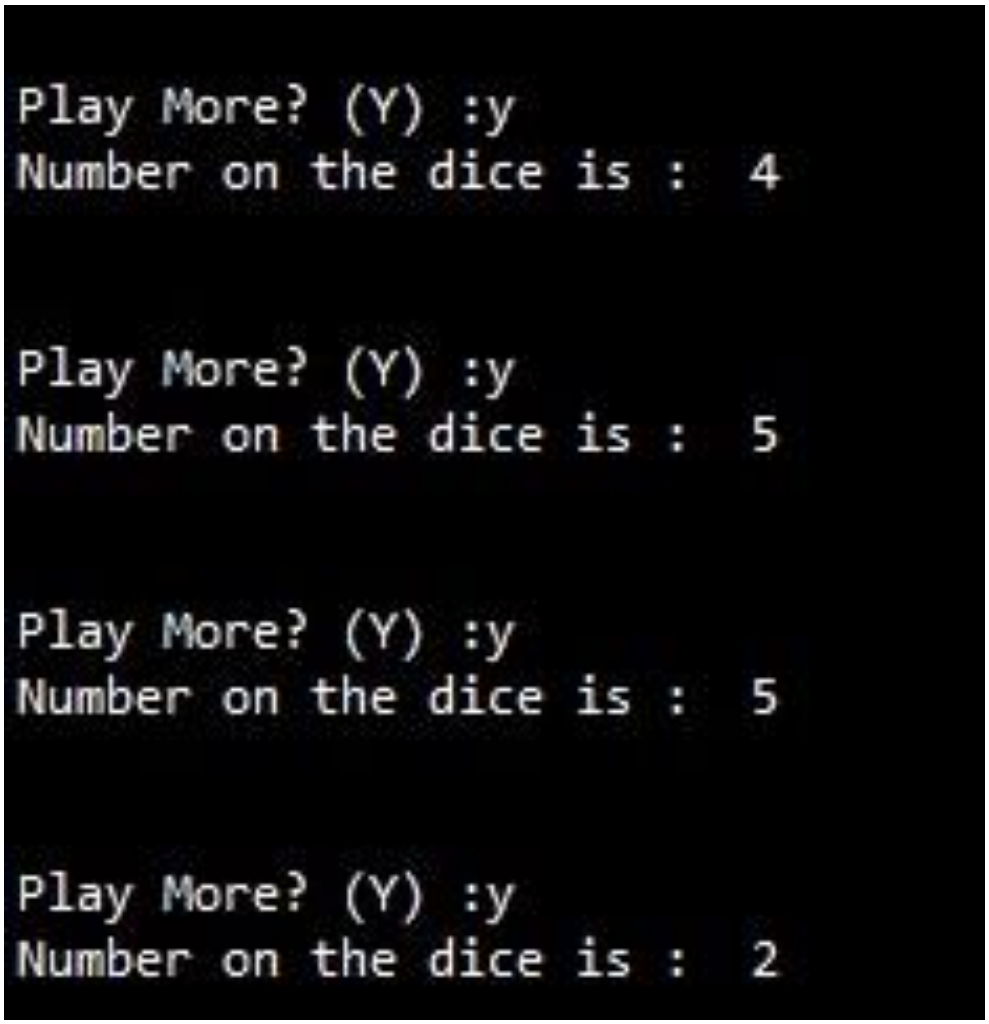
```
Enter Employee Number to search :101
============================
NAME  : Kirtan
SALARY : 50000

Search More ? (Y)y

Enter Employee Number to search :102
============================
NAME  : KP
SALARY : 40000

Search More ? (Y)
```

```
Enter Employee Number to search :103
============================
      EMPNO NOT FOUND
============================

Search More ? (Y)
```

Program 10 : Program to generate random number 1-6, simulating a dice.

```
import random
play = "y"
while play == "y":

    print("Number on the dice is : ",random.randint(1,6))
    print()
    p=input("Play More? (Y) :")
    if p.lower()!="y":
        play="n"
        break
```

OUTPUT:



Program 11 : Program to write a menu driven program to implement stack using list having option of PUSH, POP, PEEK, DISPLAY and EXIT.

```
def isEmpty( stk ):
    if stk==[]:
        return True
    else:
        return False

def Push(stk, item):
    stk.append(item)
    top = len(stk) - 1

def Pop(stk):
    if isEmpty(stk):
        return "Underflow"
    else:
        item=stk.pop()
        if len(stk) == 0:
            top = None

        else:
            top = len(stk) - 1
        return item

def Peek(stk):
    if isEmpty(stk):
        return "Underflow"

    else:
        top = len(stk) - 1
        return stk[top]

def Display(stk):
    if isEmpty(stk):
```

```
        print("Stack empty")

    else:
        top = len(stk) - 1
        print(stk[top],"<-- top")
        for a in range(top-1,-1,-1):
            print(stk[a])


Stack =[]
top = None

while True:
    print("STACK OPERATIONS")
    print("1. Push")
    print("2. Pop")
    print("3. Peek")
    print("4. Display stack")
    print("5. Exit")
    ch = int(input("Enter your choice (1-5) : "))
    if ch == 1:
        item = int(input("Enter item : "))
        Push(Stack, item)

    elif ch == 2:
        item = Pop(Stack)
        if item=="Underflow":
            print("Underflow! Stack is empty!")
        else:
            print("Popped item is", item)

    elif ch == 3:
        item=Peek(Stack)
        if item == "Underflow":
            print("Underflow! Stack is Empty!")
        else:
            print("Topmost item is", item)
```

```
    elif ch == 4:
        Display(Stack)


    elif ch == 5:
        break
    else:
        print("Invalid choice!")
```

OUTPUT:

Push:

```
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 1

Enter item : 6
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 1

Enter item : 8
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 1

Enter item : 2
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 1

Enter item : 4
```

Peek (before pop):

```
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 3
Topmost item is 4
```

Display (before pop):

```
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 4
4 <-- top
2
8
6
```

Pop and display:

```
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 2
Popped item is 4
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 4
2 <-- top
8
6
```

Exit:

```
STACK OPERATIONS
1. Push
2. Pop
3. Peek
4. Display stack
5. Exit

Enter your choice (1-5) : 5
```

Program 12 : Program to write a menu driven program to implement queue using list having option of ENQUEUE, DEQUEUE, PEEK, DISPLAY and EXIT.

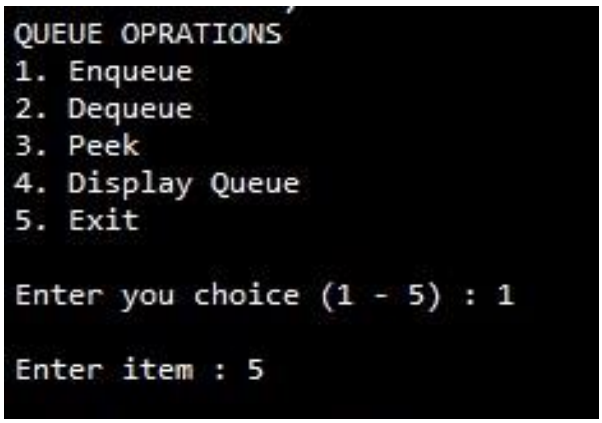
```
def isEmpty( Qu ):
   if Qu == []:
      return True
   else :
      return False

def Enqueue(Qu, item) :
   Qu . append ( item)
   if len(Qu) == 1 :
      front = rear = 0
   else :
      rear = len(Qu) - 1


def Dequeue (Qu):
   if isEmpty(Qu):
      return "Underflow"
   else:
      item=Qu.pop(0)

   if len(Qu) == 0:
      front = rear  = None
   return item

def Peek(Qu):
   if isEmpty(Qu):
      return "Underflow"
   else:
      front = 0
   return Qu[front]

def Display(Qu):
```

```
    if isEmpty(Qu):
        print("Queue Empty!")
    elif len(Qu) == 1:
        print(Qu[0],"<==front,rear")
    else:
        front = 0
        rear = len(Qu) - 1
        print(Qu[front], "<--front")
        for a in range(1,rear):
            print(Qu[a])
        print(Qu[rear],"<--rear")



queue = []
front = None
while True:
    print("QUEUE OPRATIONS")
    print("1. Enqueue")
    print("2. Dequeue")
    print("3. Peek")
    print("4. Display Queue")
    print("5. Exit")

    ch=int(input("Enter you choice (1 - 5) : "))
    if ch == 1:
        item=int(input("Enter item : "))
        Enqueue(queue,item)
        input("Press Enter to continue...")

    elif ch == 2:
        item = Dequeue(queue)
        if item == "Underflow":
            print("Underflow! Queue is empty!")
        else:
            print("Dequeue-ed item is",item)
        input("Press Enter to continue...")
```

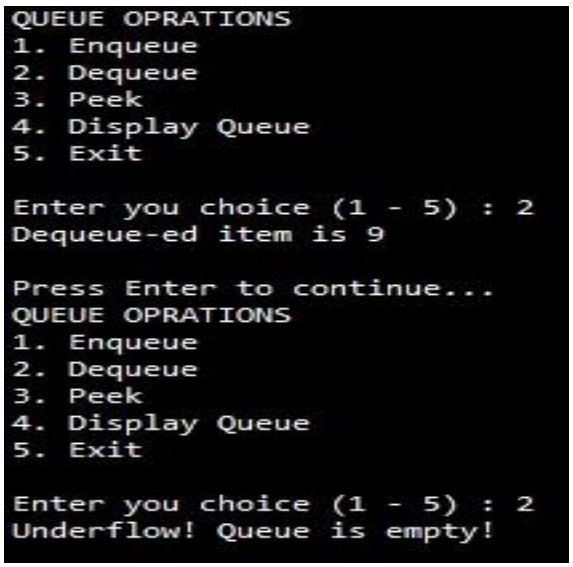
```
    elif ch ==3:
        item = Peek(queue)
        if item == "Underflow":
            print("Queue is empty!")
        else:
            print("Frontmost item is",item)
        input("Press Enter to continue...")

    elif ch ==4:
        Display(queue)
        input("Press Enter to continue...")

    elif ch ==5:
        break

    else:
        print("Invalid Choice!")
        input("Press Enter to continue...")
```

OUTPUT:

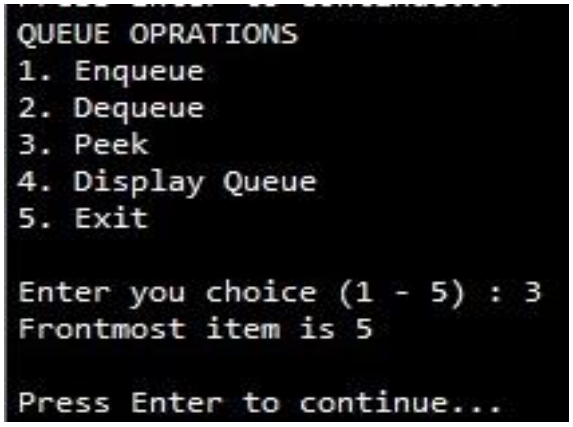
Enque:



Deque:



Peek :



Display:

```
QUEUE OPRATIONS
1. Enqueue
2. Dequeue
3. Peek
4. Display Queue
5. Exit

Enter you choice (1 - 5) : 4
5 <==front,rear

Press Enter to continue...
```

```
QUEUE OPRATIONS
1. Enqueue
2. Dequeue
3. Peek
4. Display Queue
5. Exit

Enter you choice (1 - 5) : 4
5 <--front
7
9 <--rear

Press Enter to continue...
```

Exit:

```
Press Enter to continue...
QUEUE OPRATIONS
1. Enqueue
2. Dequeue
3. Peek
4. Display Queue
5. Exit

Enter you choice (1 - 5) : 5
```

Program 13 : Program to take 10 sample phishing email, and find the most common word occurring in it. Email id will be provided like xyz@gmail.com.

```
phishingemail=[
"jackpotwin@lottery.com",\
"claimtheprize@mymoney.com","youarethewinner@lottery.com",\
"luckywinner@mymoney.com","spinthewheel@flipkart.com",\
"dealwinner@snapdeal.com","luckywinner@snapdeal.com",\
"luckyjackpot@americanlottery.com","claimtheprize@lootolottery.com",
\
"youarelucky@mymoney.com"
]

myd={}

for e in phishingemail:
   x=e.split('@')
   print(x)
   z=input()

   for w in x:
      if w not in myd:
         myd[w]=1
      else:
         myd[w]+=1
      print(myd)
      v=input()
      key_max = max(myd,key=myd.get)
print("Most Common Occuring word is :",key_max)
```

OUTPUT:

```
['jackpotwin', 'lottery.com']


{'jackpotwin': 1}


{'jackpotwin': 1, 'lottery.com': 1}


['claimtheprize', 'mymoney.com']


{'jackpotwin': 1, 'lottery.com': 1, 'claimtheprize': 1}


{'jackpotwin': 1, 'lottery.com': 1, 'claimtheprize': 1,
'mymoney.com': 1}


['youarethewinner', 'lottery.com']


{'jackpotwin': 1, 'lottery.com': 1, 'claimtheprize': 1,
'mymoney.com': 1, 'youarethewinner': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 1, 'youarethewinner': 1}


['luckywinner', 'mymoney.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 1, 'youarethewinner': 1, 'luckywinner': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 1}


['spinthewheel', 'flipkart.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 1,
'spinthewheel': 1}
```

```
{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 1,
'spinthewheel': 1, 'flipkart.com': 1}


['dealwinner', 'snapdeal.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 1,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 1,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 1}


['luckywinner', 'snapdeal.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2}


['luckyjackpot', 'americanlottery.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2, 'luckyjackpot': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 1,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2, 'luckyjackpot': 1,
'americanlottery.com': 1}
```

```
['claimtheprize', 'lootolottery.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 2,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2, 'luckyjackpot': 1,
'americanlottery.com': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 2,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2, 'luckyjackpot': 1,
'americanlottery.com': 1, 'lootolottery.com': 1}


['youarelucky', 'mymoney.com']


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 2,
'mymoney.com': 2, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2, 'luckyjackpot': 1,
'americanlottery.com': 1, 'lootolottery.com': 1,
'youarelucky': 1}


{'jackpotwin': 1, 'lottery.com': 2, 'claimtheprize': 2,
'mymoney.com': 3, 'youarethewinner': 1, 'luckywinner': 2,
'spinthewheel': 1, 'flipkart.com': 1, 'dealwinner': 1,
'snapdeal.com': 2, 'luckyjackpot': 1,
'americanlottery.com': 1, 'lootolottery.com': 1,
'youarelucky': 1}


Most Common Occuring word is : mymoney.com
```

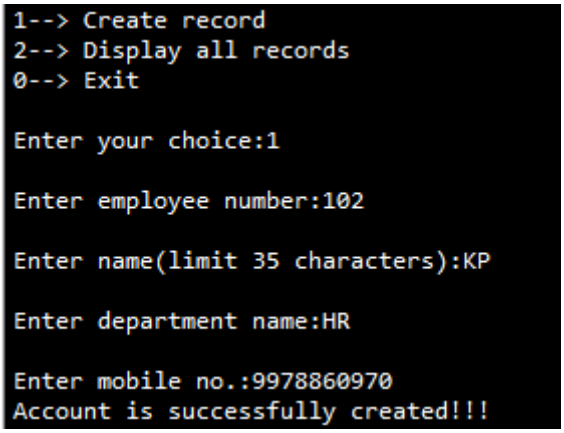
Program 14 : Program to connect with database and store record of employee and display records.

```
import mysql.connector
from tabulate import tabulate
mydb=mysql.connector.connect(host="localhost",user="root",passwd="r
oot")
mycursor=mydb.cursor()
mycursor.execute("create database if not exists employee")
mycursor.execute("use employee")
mycursor.execute("create table if not exists empl_details(emp_no char(4)
primary key,name varchar(30),dept char(20),mobileno char(10))")
mydb.commit()
while(True):

   print("1--> Create record")
   print("2--> Display all records")
   print("0--> Exit")
   ch=int(input("Enter your choice:"))

   if ch==1:
      try:
         emp_no=str(input("Enter employee number:"))
         name=input("Enter name(limit 35 characters):")
         dept=str(input("Enter department name:"))
         mn=str(input("Enter mobile no.:"))
         mycursor.execute("insert into empl_details
values('"+emp_no+"','"+name+"','"+dept+"','"+mn+"')")
         mydb.commit()
         print("Account is successfully created!!!")

      except:
         print("""
            Error in creating record...
            """)
```

```
    elif ch==2:
        try:
            mycursor.execute("select * from empl_details")
            rec=mycursor.fetchall()

print(tabulate(rec,headers=["emp_no","name","dept","mobileno"],tablef
mt="fancy_grid"))
        except:
            print("Error in Displaying record")


    elif ch==0:
        break

    else:
        print("""
              Enter Valid Choice...
              """)
```

OUTPUT:

Adding Record:



Display Record:

```
Account is successfully created!!!
1--> Create record
2--> Display all records
0--> Exit

Enter your choice:2
```

emp_no	name	dept	mobileno
101	Kirtan	Admin	9825718494
102	KP	HR	9978860970

Exit:

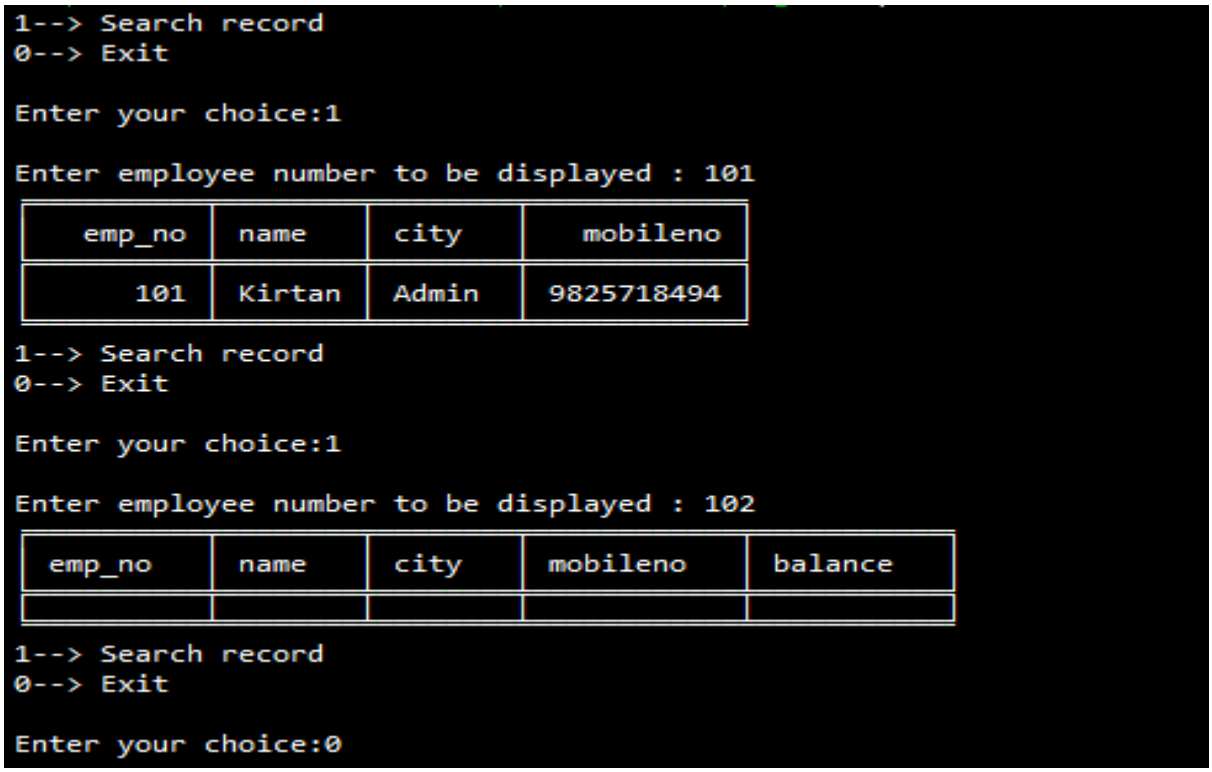
```
1--> Create record
2--> Display all records
0--> Exit

Enter your choice:0
```

Program 15 : Program to connect with database and search employee number in table employee and display record, if empno not found display appropriate message.

```
import mysql.connector
from tabulate import tabulate
mydb=mysql.connector.connect(host="localhost",user="root",passwd="r
oot")
mycursor=mydb.cursor()
mycursor.execute("create database if not exists employee")
mycursor.execute("use employee")
mycursor.execute("create table if not exists empl_details(emp_no char(4)
primary key,name varchar(30),dept char(20),mobileno char(10))")
mydb.commit()
while(True):

   print("1--> Search record")
   print("0--> Exit")
   ch=int(input("Enter your choice:"))

   if ch == 1:
      try:
         emp_no=str(input("Enter employee number to be displayed : "))
         mycursor.execute("select * from empl_details where emp_no =
'"+emp_no+"'")
         rec=mycursor.fetchall()

print(tabulate(rec,headers=["emp_no","name","city","mobileno","balanc
e"],tablefmt="fancy_grid"))
      except:
         print("""
            Error in Displaying record...
            """)

   elif ch==0:
      break
```

```
    else:
        print("""
            Enter Valid Choice...
            """)
```

OUTPUT:



```
1--> Search record
0--> Exit

Enter your choice:1

Enter employee number to be displayed : 101
```

emp_no	name	city	mobileno
101	Kirtan	Admin	9825718494

```
1--> Search record
0--> Exit

Enter your choice:1

Enter employee number to be displayed : 102
```

emp_no	name	city	mobileno	balance

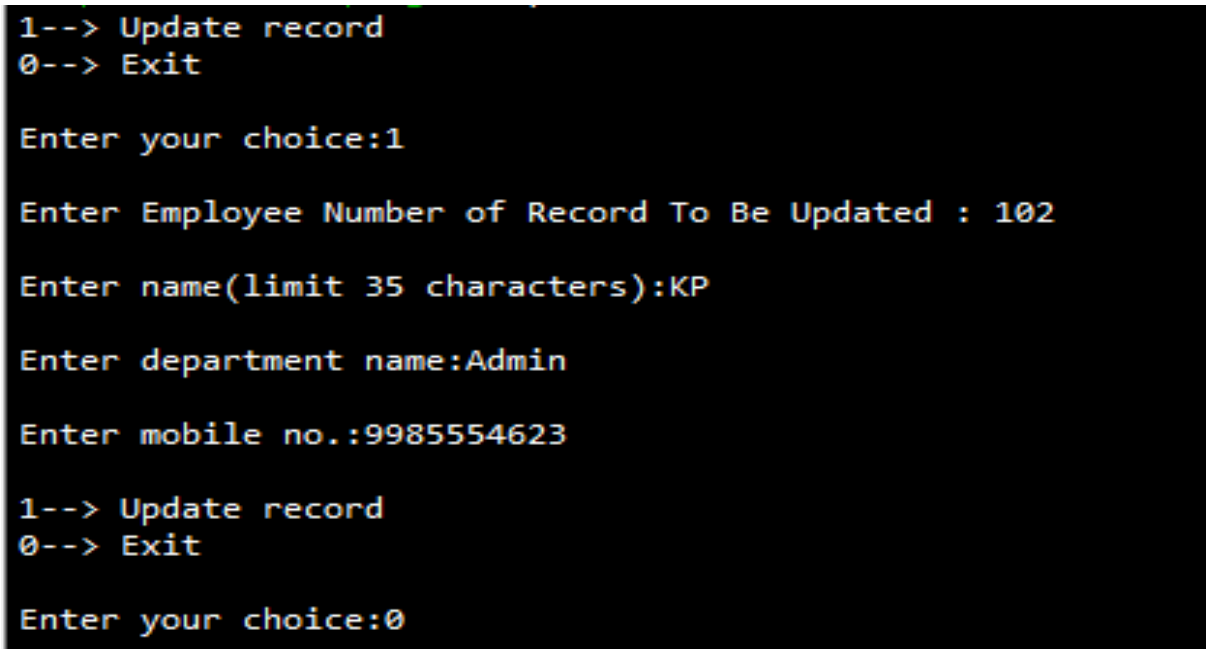
```
1--> Search record
0--> Exit

Enter your choice:0
```

Program 16 : Program to connect with database and update the employee record of entered empno.

```
import mysql.connector
from tabulate import tabulate
mydb=mysql.connector.connect(host="localhost",user="root",passwd="r
oot")
mycursor=mydb.cursor()
mycursor.execute("create database if not exists employee")
mycursor.execute("use employee")
mycursor.execute("create table if not exists empl_details(emp_no char(4)
primary key,name varchar(30),dept char(20),mobileno char(10))")
mydb.commit()
while(True):

   print("1--> Update record")
   print("0--> Exit")
   ch=int(input("Enter your choice:"))
   if ch==1:
      try:
         emp_no=str(input("Enter Employee Number of Record To Be
Updated : "))
         name=input("Enter name(limit 35 characters):")
         dept=str(input("Enter department name:"))
         mn=str(input("Enter mobile no.:"))


         mycursor.execute("update employee set name = {},dept={},mn={}
where emp_no={}".format(name,dept,mn))
         mydb.commit()

print(tabulate(headers=["emp_no","name","city","mobileno","balance"],t
ablefmt="fancy_grid"))
      except:
```

```
        print("Record Not Updated" )

    elif ch==0:
        break

    else:
        print(" Enter a  Valid Choice ")
```

OUTPUT:



Program 17 : Program to connect with database and delete the record of entered employee number.

```
import mysql.connector as mycon
con = mycon.connect(host='localhost',user='root',password="root",\
database="employee")
cur = con.cursor()

ans='y'
while ans.lower()=='y':
   eno = int(input("ENTER EMPNO TO DELETE :"))
   query="select * from empl_details where emp_no={}".format(eno)
   cur.execute(query)
   result = cur.fetchall()
   if cur.rowcount==0:
      print("Sorry! Empno not found ")
   else:
      print("%10s"%"EMPNO","%20s"%"NAME",
"%15s"%"DEPARTMENT", "%10s"%"SALARY")
   for row in result:

print("%10s"%row[0],"%20s"%row[1],"%15s"%row[2],"%10s"%row[3
])
      choice=input("\n## ARE YOUR SURE TO DELETE ? (Y) :")
      if choice.lower()=='y':
         query="delete from empl_details where
emp_no={}".format(eno)
         cur.execute(query)
         con.commit()
         print("=== RECORD DELETED SUCCESSFULLY! ===")
   ans=input("DELETE MORE ? (Y) :")
```

OUTPUT:

```
ENTER EMPNO TO DELETE :102
     EMPNO                  NAME       DEPARTMENT      SALARY
       102                    KP                HR 9978860970


## ARE YOUR SURE TO DELETE ? (Y) :y
=== RECORD DELETED SUCCESSFULLY! ===

DELETE MORE ? (Y) :
```

Program 18 : Write a program to count the number of vowels present in a text file.

```
file=open("poem.txt","r")
data=file.read()
print(data)
vowels=0
for ch in data:
   if ch in "AaEeIiOoUu":
      vowels+=1
print("No. of vowels :",vowels)
```

OUTPUT:

```
Difficult because we think that happiness is found
Only in the places where wealth and fame abound.
And so we go on searching in places of pleasure
seeking recognition and monetary treasure,
Unaware that happiness is just a state of mind
within the reach of everyone who takes time to be kind,
For in making others happy we will be happy, too.
For the happiness you give away returns to shine on you.
No. of vowels : 128
```

Program 19 : Write a program to write those lines which have the character 'p' from one text file to another text file.

```
def Display():
   f=open("poem.txt","r")
   f1=open("myfilep.txt","w")
   while True:
      line=f.readline()
      if line=='':
         break
      if 'p' in line:
         f1.write(line)
         print(line)
   f.close()
   f1.close()
Display()
```

OUTPUT:

```
Difficult because we think that happiness is found

Only in the places where wealth and fame abound.

And so we go on searching in places of pleasure

Unaware that happiness is just a state of mind

For in making others happy we will be happy, too.

For the happiness you give away returns to shine on you.
```

Program 20 : Write a program to count number of words in a file.

```
file=open("poem.txt","r")
count=0
for line in file:
   wrds=line.split(" ")
   count += len(wrds)
   print(line)
file.close()
print("No. of words in a file : ",count)
```

OUTPUT:

```
Difficult because we think that happiness is found

Only in the places where wealth and fame abound.

And so we go on searching in places of pleasure

seeking recognition and monetary treasure,

Unaware that happiness is just a state of mind

within the reach of everyone who takes time to be kind,

For in making others happy we will be happy, too.

For the happiness you give away returns to shine on you.
No. of words in a file :  74
```