

## 1) Aim: Program To Check Whether Given Number Is Palindrome Or Not

```
palindrome5.py - Z:\palindrome5.py (3.7.3)
File Edit Format Run Options Window Help
num=int(input("Enter a Number:"))
temp=num
rev=0
while(num>0):
    rev=(rev*10)+num%10
    num=num//10
if(temp==rev):
    print("Given number is Palindrome")
else:
    print("Given number is not palindrome")
```

### Output:

```
Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: Z:\palindrome5.py =====
Enter a Number:525
Given number is Palindrome
>>>|
```

## 2) Aim: Program To Check Weather Given String Is Palindrome Or Not

```
string palindrome6.py - Z:\string palindrome6.py (3.7.3)
File Edit Format Run Options Window Help
str1=input("Enter a string: ")
print("Given string is :",str1)
print("Reversed string is:",str1[::-1])
if str1==str1[-1::-1]:
    print("Given string is palindrome")
else:
    print("Given string is not palindrome")
```

### Output:

```
>>>
===== RESTART: Z:\string palindrome6.py =====MA
Enter a string: MADAM
Given string is : MADAM
Reversed string is: MADAM
Given string is palindrome
>>>|
```

## 3) Aim: Read A Text File Line By Line And Display Each Word Separated By #

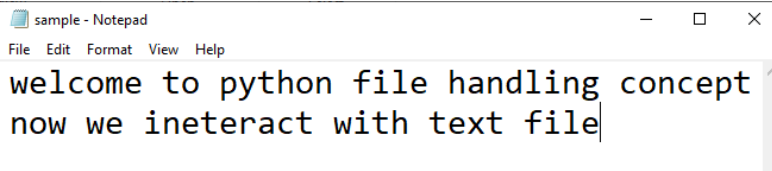
```
sample - Notepad
File Edit Format View Help
welcome to python file handling concept
now we ineteract with text file

filehandling1.py - Z:\filehandling1.py (3.7.3)
File Edit Format Run Options Window Help
myfile=open(r"Z:\sample.txt","r")
a=myfile.readlines()
for i in range(len(a)):
    b=a[i].split()
    print("#".join(b))
myfile.close()
```

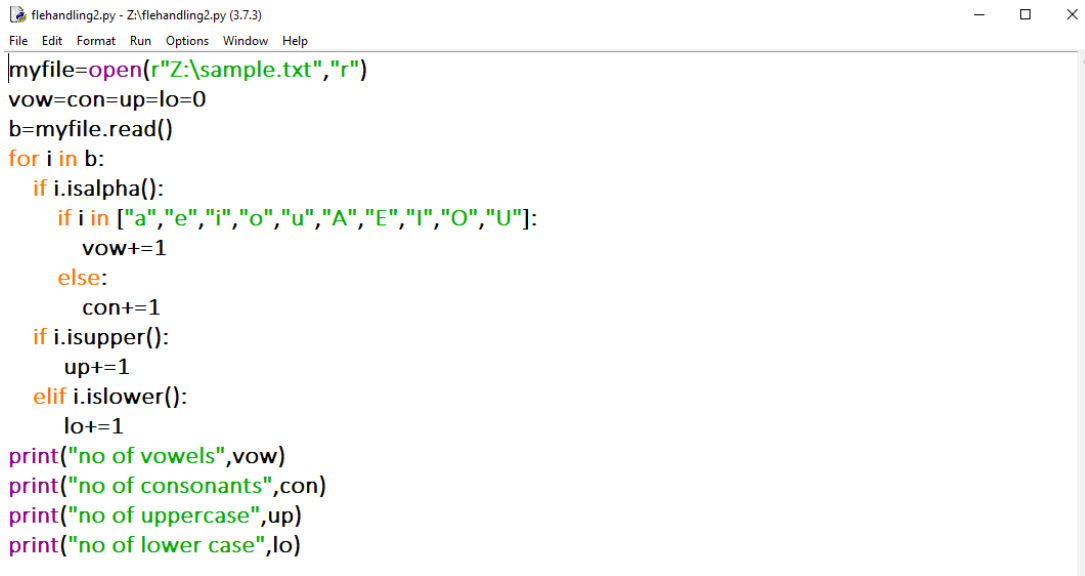
### Output:

```
>>>
===== RESTART: Z:\filehandling1.py =====
welcome#to#python#file#handling#concept
now#we#ineteract#with#text#file
>>>|
```

**4) Aim: Read a text file and display the number of vowels/consonants/uppercase/lowercase characters in the file.**



```
sample - Notepad
File Edit Format View Help
welcome to python file handling concept
now we ineteract with text file
```

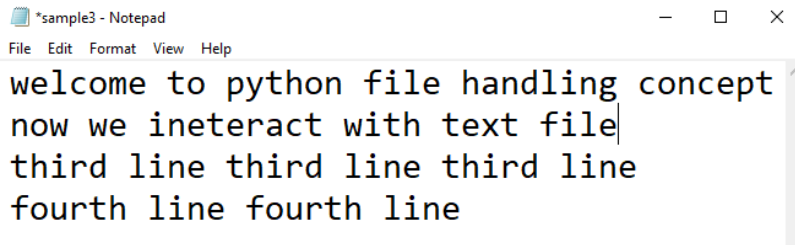


```
flehandling2.py - Z:\flehandling2.py (3.7.3)
File Edit Format Run Options Window Help
myfile=open(r"Z:\sample.txt","r")
vow=con=up=lo=0
b=myfile.read()
for i in b:
    if i.isalpha():
        if i in ["a","e","i","o","u","A","E","I","O","U"]:
            vow+=1
        else:
            con+=1
    if i.isupper():
        up+=1
    elif i.islower():
        lo+=1
print("no of vowels",vow)
print("no of consonants",con)
print("no of uppercase",up)
print("no of lower case",lo)
```

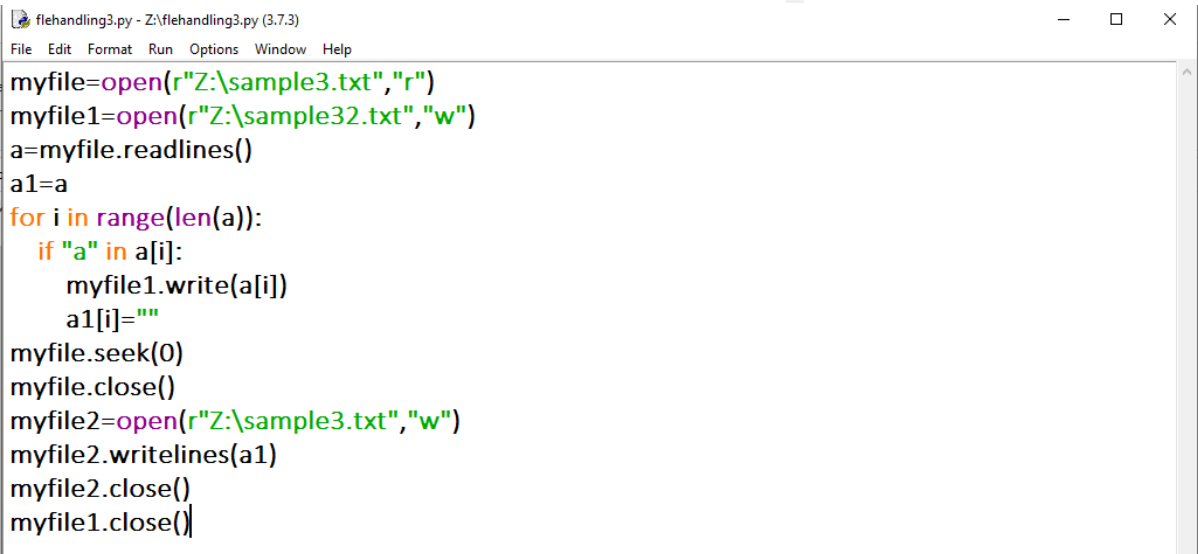
**Output:**

```
>>>
===== RESTART: Z:\flehandling2.py =====
no of vowels 21
no of consonants 39
no of uppercase 0
no of lower case 60
>>>|
```

**5) Aim : Remove all the lines that contain the character a' in a file and write it to another file**

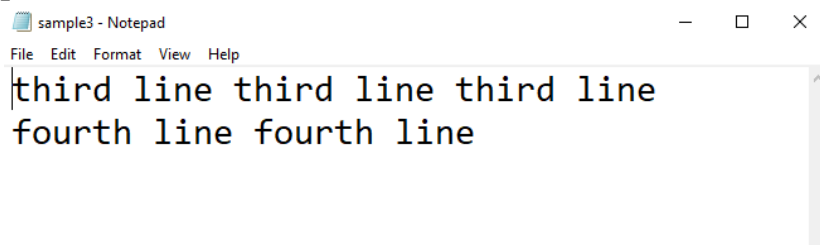


```
*sample3 - Notepad
File Edit Format View Help
welcome to python file handling concept
now we ineteract with text file
third line third line third line
fourth line fourth line
```



```
flehandling3.py - Z:\flehandling3.py (3.7.3)
File Edit Format Run Options Window Help
myfile=open(r"Z:\sample3.txt","r")
myfile1=open(r"Z:\sample32.txt","w")
a=myfile.readlines()
a1=a
for i in range(len(a)):
    if "a" in a[i]:
        myfile1.write(a[i])
        a1[i]=" "
myfile.seek(0)
myfile.close()
myfile2=open(r"Z:\sample3.txt","w")
myfile2.writelines(a1)
myfile2.close()
myfile1.close()
```

**Output:**



```
sample3 - Notepad
File Edit Format View Help
third line third line third line
fourth line fourth line
```

welcome to python file handling concept  
now we ineteract with text file

6) Aim : Create a binary file with name and roll number. Search for a given roll number and display the name, if not found display appropriate message.

```
*binaryfile4.py - Z:\binaryfile4.py (3.7.3)
File Edit Format Run Options Window Help

import pickle
myfile=open(r"Z:\sample4.dat","wb+")
a={}
found=False
for i in range(5):
    roll=int(input("enter the roll num"))
    name=input("enter the name")
    a["roll"]=roll
    a["name"]=name
    pickle.dump(a,myfile)

myfile.seek(0)
c=int(input("enter the roll no to search"))
try:
    while True:
        b=pickle.load(myfile)
        if b["roll"]==c:
            print (b)
            found=True
except EOFError:
    if found:
        myfile.close()
    else:
        print("student details not found")
        myfile.close()
```

### Output:

```
>>>
===== RESTART: Z:\binaryfile4.py =====
Enter the roll num : 101
Enter the name : ANU
Enter the roll num : 102
Enter the name : BINCY
Enter the roll num : 103
Enter the name : CHARLES
Enter the roll num : 104
Enter the name : HIRA
Enter the roll num : 105
Enter the name : SINI
Enter the roll no to search : 104
{'roll': 104, 'name': 'HIRA'}
>>>|
```

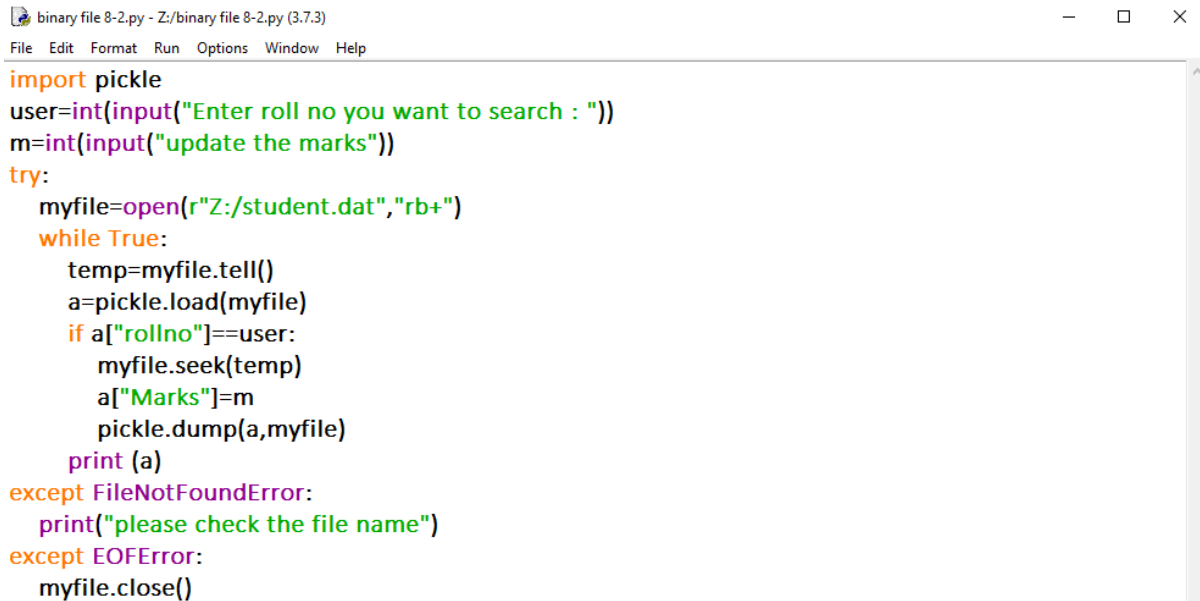
7) Aim : Create A Binary File With Roll Number, Name And Marks. Input A Roll Number And Update The Marks.

```
binary file 8.py - Z:/binary file 8.py (3.7.3)
File Edit Format Run Options Window Help

import pickle
emp1={"rollNo":101,"name":"student1","Marks":90}
emp2={"rollNo":102,"name":"student2","Marks":56}
emp3={"rollNo":103,"name":"student3","Marks":97}
emp4={"rollNo":104,"name":"student4","Marks":19}
emp5={"rollNo":105,"name":"student5","Marks":95}
emp6={"rollNo":106,"name":"student6","Marks":92}
emp7={"rollNo":107,"name":"student7","Marks":57}
emp8={"rollNo":108,"name":"student8","Marks":99}
myfile=open(r"Z:/student.dat","wb")
pickle.dump(emp1,myfile)
pickle.dump(emp2,myfile)
pickle.dump(emp3,myfile)
pickle.dump(emp4,myfile)
pickle.dump(emp5,myfile)
pickle.dump(emp6,myfile)
pickle.dump(emp7,myfile)
pickle.dump(emp8,myfile)
print("succesful")
myfile.close()
```

## Output :-

```
>>>
===== RESTART: Z:/binary file 8.py =====
successful
>>>|
```

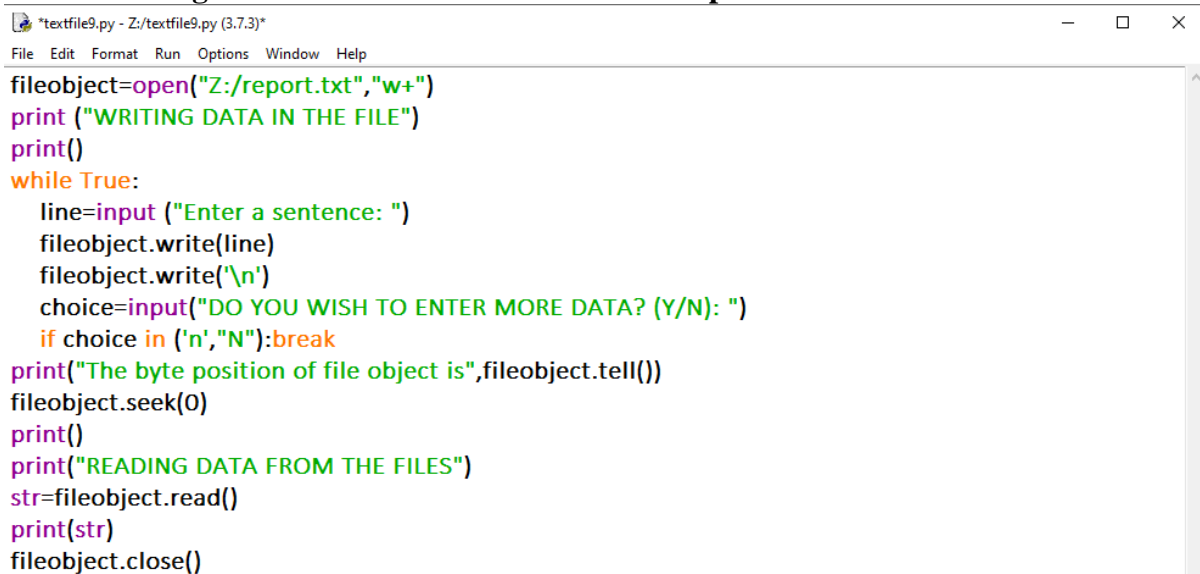


```
binary file 8-2.py - Z:/binary file 8-2.py (3.7.3)
File Edit Format Run Options Window Help
import pickle
user=int(input("Enter roll no you want to search : "))
m=int(input("update the marks"))
try:
    myfile=open(r"Z:/student.dat","rb+")
    while True:
        temp=myfile.tell()
        a=pickle.load(myfile)
        if a["rollno"]==user:
            myfile.seek(temp)
            a["Marks"]=m
            pickle.dump(a,myfile)
            print (a)
except FileNotFoundError:
    print("please check the file name")
except EOFError:
    myfile.close()
```

## Output :-

```
>>>
===== RESTART: Z:/binary file 8-2.py =====
Enter roll no you want to search : 103
update the marks100
{'rollno': 101, 'name': 'student1', 'Marks': 90}
{'rollno': 102, 'name': 'student2', 'Marks': 56}
{'rollno': 103, 'name': 'student3', 'Marks': 100}
{'rollno': 104, 'name': 'student4', 'Marks': 19}
{'rollno': 105, 'name': 'student5', 'Marks': 95}
{'rollno': 106, 'name': 'student6', 'Marks': 92}
{'rollno': 107, 'name': 'student7', 'Marks': 57}
{'rollno': 108, 'name': 'student8', 'Marks': 99}
>>>|
```

## 8) Aim : Write A Program To Perform Read And Write Operations On A Text File



```
*textfile9.py - Z:/textfile9.py (3.7.3)*
File Edit Format Run Options Window Help
fileobject=open("Z:/report.txt","w+")
print ("WRITING DATA IN THE FILE")
print()
while True:
    line=input ("Enter a sentence: ")
    fileobject.write(line)
    fileobject.write("\n")
    choice=input("DO YOU WISH TO ENTER MORE DATA? (Y/N): ")
    if choice in ('n',"N"):break
print("The byte position of file object is",fileobject.tell())
fileobject.seek(0)
print()
print("READING DATA FROM THE FILES")
str=fileobject.read()
print(str)
fileobject.close()
```

## Output:

```
>>>
===== RESTART: Z:/textfile9.py =====
WRITING DATA IN THE FILE

Enter a sentence: WELCOME TO PYTHON PROGRAMMING
DO YOU WISH TO ENTER MORE DATA? (Y/N): N
The byte position of file object is 31
|
READING DATA FROM THE FILES
WELCOME TO PYTHON PROGRAMMING
>>>
```

9) Aim: Program to enter two numbers and print the arithmetic operations like +,-,\*,/,// and %.

```
arithmetic operations7.py - Z:\arithmetic operations7.py (3.7.3)
File Edit Format Run Options Window Help
num1 = int(input("Enter First number:"))
num2 = int(input("Enter Second number:"))
add = num1 + num2
dif = num1 - num2
mul = num1 * num2
div = num1 / num2
floor_div = num1 // num2
power = num1 ** num2
modulus = num1 % num2
print("sum of",num1,"and",num2,"is:",add)
print("Difference of",num1,"and",num2,"is:",dif)
print("product of",num1,"and",num2,"is:",mul)
print("Division of",num1,"and",num2,"is:",div)
print("Floor Division of",num1,"and",num2,"is:",floor_div)
print("Exponent of",num1,"and",num2,"is:",power)
print("Modulus of",num1,"and",num2,"is:",modulus)
```

Output:

```
>>>
===== RESTART: Z:\arithmetic operations7.py =====
Enter First number:10
Enter Second number:5
sum of 10 and 5 is: 15
Difference of 10 and 5 is: 5
product of 10 and 5 is: 50
Division of 10 and 5 is: 2.0
Floor Division of 10 and 5 is: 2
Exponent of 10 and 5 is: 100000
Modulus of 10 and 5 is: 0
>>>|
```

10) Aim: Write a program to find whether an inputted number is perfect or not.

```
*perfect number 10.py - Z:/perfect number 10.py (3.7.3)*
File Edit Format Run Options Window Help
n = int(input("Enter any number : "))
sum1 = 0
for i in range(1, n):
    if(n % i == 0):
        sum1 = sum1 + i
if (sum1 == n) :
    print("The number is a perfect number!")
else:
    print("The number is not a perfect number!")
```

Output:

```
>>>
===== RESTART: Z:/perfect number 10.py =====
Enter any number : 6
The number is a perfect number!
>>>
===== RESTART: Z:/perfect number 10.py =====
Enter any number : 5
The number is not a perfect number!
>>>|
```

11) Aim: Write a Program to check if the entered number is Armstrong or not.

```
*Armstrong 11.py - Z:\Armstrong 11.py (3.7.3)*
File Edit Format Run Options Window Help
no=int(input("Enter any number to check : "))
no1 = no
sum = 0
while (no>0):
    ans = no % 10;
    sum = sum + (ans * ans * ans)
    no = int (no / 10)
if sum == no1:
    print("Armstrong Number")
else:
    print("Not an Armstrong Number")
```

Output:

```
>>>
===== RESTART: Z:/Armstrong 11.py =====
Enter any number to check : 371
Armstrong Number
>>>
===== RESTART: Z:/Armstrong 11.py =====
Enter any number to check : 1234
Not an Armstrong Number
>>>|
```

12) Aim: Write a Program to find factorial of the entered number.

```
*factorial 12.py - Z:/factorial 12.py (3.7.3)*
File Edit Format Run Options Window Help
num = int(input("Enter the number for calculating its factorial : "))
fact = 1
i = 1
while i<=num:
    fact = fact*i
    i = i + 1
print("The factorial of ",num,"=",fact)
```

Output:

```
>>>
===== RESTART: Z:/factorial 12.py =====
Enter the number for calculating its factorial : 5
The factorial of 5 = 120
>>>
===== RESTART: Z:/factorial 12.py =====
Enter the number for calculating its factorial : 3
The factorial of 3 = 6
>>>|
```

### 13) Write a program to enter the number of terms and top print the fibonacci series

```
*fibnoci14.py - Z:\fibnoci14.py (3.7.3)*
File Edit Format Run Options Window Help
nterms = int(input("How many terms? "))
n1,n2 = 0, 1
count = 0
if nterms <= 0:
    print("Please enter a positive integer")
elif nterms == 1:
    print("Fibonacci sequence upto",nterms,":")
    print(n1)
else:
    print("Fibonacci sequence:")
    while count < nterms:
        print(n1)
        nth = n1 + n2
        n1 = n2
        n2 = nth
        count += 1
```

#### Output:

```
===== RESTART: Z:/fibnoci14.py =====
How many terms? 10
Fibonacci sequence:
0
1
1
2
3
5
8
13
21
34
>>> |
```

### 14) Write a random number generator that generates random numbers between 1 and 6(simulates a dice)

```
*random number 13.py - Z:/random number 13.py (3.7.3)*
File Edit Format Run Options Window Help
import random
while True:
    num=random.randint(1,6)
    if num==6:
        print("Hey.....you got",num,".....Congratulations!!!")
    elif num==1:
        print("Well tried.....But you got ",num)
    else:
        print("You got: ",num)
    print(num)
    ch=input("Roll again ? (Y/N)")
    if ch in 'Nn':
        break
    print("Thanks for playing !!!!!!!!!!!")
```

#### Output:

```
>>>
===== RESTART: Z:/random number 13.py =====
You got: 4
4
Roll again ? (Y/N)y
Thanks for playing !!!!!!!!!!!
Well tried.....But you got 1
1
Roll again ? (Y/N)|
```

## 15) Write a python program to implement a stack using a list data structure.

```
*stackusinglist15.py - Z:/stackusinglist15.py (3.7.3)*
File Edit Format Run Options Window Help
stack = ["one", "Two", "Three"]
stack.append("Four")
stack.append("Five")
print("Print the elements in the stack",stack)
print("pop 1",stack.pop())
print("After performing first pop",stack)
print("pop 2",stack.pop())
print("After performing second pop",stack)
```

### Output:

```
>>>
===== RESTART: Z:/stackusinglist15.py =====
Print the elements in the stack ['one', 'Two', 'Three', 'Four', 'Five']
pop 1 Five
After performing first pop ['one', 'Two', 'Three', 'Four']
pop 2 Four
After performing second pop ['one', 'Two', 'Three']
>>> |
```

## 16) Create a CSV file by entering user- id and password, read and search the password for given user-id

```
*csvfile.py - C:/Users/jnvm22/AppData/Local/Programs/Python/Python37-32/csvfile.py (3.7.3)*
File Edit Format Run Options Window Help
```

```
import csv
def write():
    f=open("details.csv", "w",newline="")
    wo=csv.writer(f)
    wo.writerow(["UserId","Password"])
    while True:
        u_id=input("Enter User - Id : ")
        pswd=input("Enter Password : ")
        data=[u_id,pswd]
        wo.writerow(data)
        ch=input("Do you want to enter more record (Y/N) : ")
        if ch in 'Nn':
            break
    f.close()
def read():
    f=open("details.csv","r")
    ro=csv.reader(f)
    for i in ro:
        print(i)
def search():
    f=open("details.csv","r")
    Found=0
    u=input("Enter user- id to Search : ")
    ro=csv.reader(f)
    next(ro)
    for i in ro:
        if i[0]==u:
            print(i[1])
            Found=1

    f.close()
    if Found==0:
        print("Sorry...No record found..")
write()
read()
search()
```

Activate Windows  
Go to Settings to activate Windows.

Activate Windows



## Output:

>>>

```
RESTART: C:/Users/jnvm22/AppData/Local/Programs/Python/Python37-32/csvfile.py
```

```
Enter User - Id : JNV HASSAN
```

```
Enter Password : MANAVI
```

```
Do you want to enter more record (Y/N) : N
```

```
['UserId', 'Password']
```

```
['JNV HASSAN', 'MANAVI']
```

```
Enter user- id to Search : JNV HASSAN
```

```
MANAVI
```

>>>

```
RESTART: C:/Users/jnvm22/AppData/Local/Programs/Python/Python37-32/csvfile.py
```

```
['UserId', 'Password']
```

```
['JNV HASSAN', 'MANAVI']
```

```
Enter user- id to Search : ABC
```

```
Sorry...No record found..
```

>>>|

## 17) Create a student table and insert data . Implement the following SQL commands on the student table:

- ALTER table to add new attributes / modify data type / drop attribute
- UPDATE table to modify data
- ORDER By to display data in ascending / descending order
- DELETE to remove tuple(s)
- GROUP BY and find the min, max, sum, count and average

### Creating Database

```
mysql> create database class12cs;  
Query OK, 1 row affected (0.10 sec)
```

### Use Database

```
mysql> use class12cs;  
Database changed  
mysql>
```

### Create Table

```
mysql> Create table student (Rno int primary key, Name varchar(10) NOT NULL, Gender varchar(8), Marks int, Scode varchar(5));  
Query OK, 0 rows affected (1.48 sec)
```

### Structure of Table

```
mysql> Desc student;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| Rno   | int           | NO   | PRI | NULL    |       |  
| Name  | varchar(10)   | NO   |     | NULL    |       |  
| Gender| varchar(8)    | YES  |     | NULL    |       |  
| Marks | int           | YES  |     | NULL    |       |  
| Scode | varchar(5)    | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
5 rows in set (0.20 sec)
```

### Insert Values

```
mysql> Insert into student values(1, 'Neha', 'Female', 100, 'S101');  
Query OK, 1 row affected (0.09 sec)  
  
mysql> Insert into student values(2, 'Ammu', 'Female', 90, 'S101');  
Query OK, 1 row affected (0.08 sec)  
  
mysql> Insert into student values(3, 'Kavitha', 'Female', 80, 'S102');  
Query OK, 1 row affected (0.06 sec)  
  
mysql> Insert into student values(4, 'Manu', 'male', 70, 'S103');  
Query OK, 1 row affected (0.09 sec)  
  
mysql> Insert into student values(5, 'Ram', 'male', 60, 'S104');  
Query OK, 1 row affected (0.07 sec)  
  
mysql> Insert into student values(6, 'Raju', 'male', 50, 'S105');  
Query OK, 1 row affected (0.07 sec)
```

## Select All row

```
mysql> Select * from student ;
+-----+
| Rno | Name   | Gender | Marks | Scode |
+-----+
| 1   | Neha   | Female | 100   | S101  |
| 2   | Ammu   | Female | 90    | S101  |
| 3   | Kavitha | Female | 80    | S102  |
| 4   | Manu   | male   | 70    | S103  |
| 5   | Ram    | male   | 60    | S104  |
| 6   | Raju   | male   | 50    | S105  |
+-----+
6 rows in set (0.00 sec)
```

## Select particular row

```
mysql> Select * from student where Scode='s101';
+-----+
| Rno | Name | Gender | Marks | Scode |
+-----+
| 1   | Neha | Female | 100   | S101  |
| 2   | Ammu | Female | 90    | S101  |
+-----+
2 rows in set (0.00 sec)
```

## Select using between operator

```
mysql> Select Name from student where Marks between 50 and 80;
+-----+
| Name |
+-----+
| Kavitha |
| Manu |
| Ram |
| Raju |
+-----+
4 rows in set (0.00 sec)
```

## Alter table (add new columns)

```
mysql> Alter table student add column(contactNo varchar(15),Address char(10));
Query OK, 0 rows affected (0.23 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

## Alter table (Remove column)

```
mysql> Alter table student drop contactNo;
Query OK, 0 rows affected (0.12 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

## Alter table (modify Data type)

```
mysql> Alter table student modify marks decimal;
Query OK, 6 rows affected (0.27 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

```
mysql> desc student;
+-----+
| Field | Type           | Null | Key | Default | Extra |
+-----+
| Rno   | int            | NO   | PRI | NULL    |       |
| Name  | varchar(10)    | NO   |     | NULL    |       |
| Gender | varchar(8)     | YES  |     | NULL    |       |
| marks | decimal(10,0) | YES  |     | NULL    |       |
| Scode | varchar(5)     | YES  |     | NULL    |       |
| Address | char(10)       | YES  |     | NULL    |       |
+-----+
6 rows in set (0.06 sec)
```

## Update Command

```
mysql> Update student set Address = 'sector 18';
Query OK, 6 rows affected (0.13 sec)
Rows matched: 6 Changed: 6 Warnings: 0

mysql> SELECT * FROM STUDENT;
+-----+
| Rno | Name   | Gender | marks | Scode | Address |
+-----+
| 1   | Neha   | Female | 100   | S101  | sector 18 |
| 2   | Ammu   | Female | 90    | S101  | sector 18 |
| 3   | Kavitha | Female | 80    | S102  | sector 18 |
| 4   | Manu   | male   | 70    | S103  | sector 18 |
| 5   | Ram    | male   | 60    | S104  | sector 18 |
| 6   | Raju   | male   | 50    | S105  | sector 18 |
+-----+
6 rows in set (0.00 sec)
```

## Update particular row

```
mysql> Update student set address= 'sector 10' where scode='s101';
Query OK, 2 rows affected (0.08 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> SELECT * FROM STUDENT;
+----+-----+-----+-----+-----+-----+
| Rno | Name   | Gender | marks | Scode | Address |
+----+-----+-----+-----+-----+-----+
| 1   | Neha   | Female | 100   | S101  | sector 10 |
| 2   | Ammu   | Female | 90    | S101  | sector 10 |
| 3   | Kavitha | Female | 80    | S102  | sector 18 |
| 4   | Manu   | male   | 70    | S103  | sector 18 |
| 5   | Ram    | male   | 60    | S104  | sector 18 |
| 6   | Raju   | male   | 50    | S105  | sector 18 |
+----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

## Order by name Ascending order

```
mysql>
mysql> Select * from student order by name ASC;
+----+-----+-----+-----+-----+-----+
| Rno | Name   | Gender | marks | Scode | Address |
+----+-----+-----+-----+-----+-----+
| 2   | Ammu   | Female | 90    | S101  | sector 10 |
| 3   | Kavitha | Female | 80    | S102  | sector 18 |
| 4   | Manu   | male   | 70    | S103  | sector 18 |
| 1   | Neha   | Female | 100   | S101  | sector 10 |
| 6   | Raju   | male   | 50    | S105  | sector 18 |
| 5   | Ram    | male   | 60    | S104  | sector 18 |
+----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

## Order by marks Descending order

```
mysql> Select name,marks from student order by marks desc;
+-----+-----+
| name  | marks |
+-----+-----+
| Neha  | 100   |
| Ammu  | 90    |
| Kavitha | 80   |
| Manu  | 70    |
| Ram   | 60    |
| Raju  | 50    |
+-----+-----+
6 rows in set (0.00 sec)
```

## Delete particular row

```
mysql> Delete from student where Rno=6;
Query OK, 1 row affected (0.04 sec)
```

## GROUP BY

```
mysql> Select gender,count(*) from student GROUP BY Gender;
+-----+-----+
| gender | count(*) |
+-----+-----+
| Female | 3        |
| male   | 2        |
+-----+-----+
2 rows in set (0.02 sec)
```

## Select maximum mark

```
mysql> Select max(marks) from student;
+-----+
| max(marks) |
+-----+
| 100        |
+-----+
1 row in set (0.00 sec)
```

## Select minimum mark

```
mysql> Select min(marks) from student;
+-----+
| min(marks) |
+-----+
| 60         |
+-----+
1 row in set (0.00 sec)
```

## Average function

```
mysql> Select avg(marks) from student;
+-----+
| avg(marks) |
+-----+
| 80.0000    |
+-----+
1 row in set (0.00 sec)
```

## Count function

```
mysql> Select score,count(*) from student GROUP BY Score;
+-----+-----+
| score | count(*) |
+-----+-----+
| S101  |         2 |
| S102  |         1 |
| S103  |         1 |
| S104  |         1 |
+-----+-----+
4 rows in set (0.00 sec)
```

## HAVING Clause

```
mysql> Insert into student values(6,'Ramu','male',65,'S101','sector 10');
Query OK, 1 row affected (0.06 sec)

mysql> Select score,count(*) from student GROUP BY Score Having Count(*)>2;
+-----+-----+
| score | count(*) |
+-----+-----+
| S101  |         3 |
+-----+-----+
1 row in set (0.00 sec)
```

## Create new table

```
mysql> Create table stream(Scode varchar(5) primary key,Sname varchar(15));
Query OK, 0 rows affected (0.88 sec)
```

## Insert values

```
mysql> Insert into stream values('s101','Non Medical');
Query OK, 1 row affected (0.06 sec)

mysql> Insert into stream values('s102','Medical');
Query OK, 1 row affected (0.10 sec)

mysql> Insert into stream values('s103','Commerce');
Query OK, 1 row affected (0.05 sec)

mysql> Insert into stream values('s104','Art');
Query OK, 1 row affected (0.05 sec)
```

## Table Join

```
mysql> Select name,sname from student,stream where student.Scode=Stream.Scode;
+-----+-----+
| name  | sname      |
+-----+-----+
| Neha  | Non Medical |
| Ammu  | Non Medical |
| Kavitha | Medical    |
| Manu  | Commerce   |
| Ram   | Art        |
| Ramu  | Non Medical |
+-----+-----+
6 rows in set (0.00 sec)
```

18) Write a python program to connect with database , search the records from the table based on the rollno and display the details of the student.

**DATABASE : FILE**

**TABLE NAME : STUDENT**

```
sql 18.py - Z:\sql 18.py (3.7.3)
File Edit Format Run Options Window Help
import mysql.connector
con=mysql.connector.connect(host="localhost", user="root", password="1234", database='file', charset='utf8')
if con.is_connected():
    print("Connection succesful.....")
else:
    print("Ooops....could not connect")
cur=con.cursor()
found=0
cur.execute("select * from student where Rno=9")
data=cur.fetchall()
for i in data:
    print(i)
    found=1
if found==0:
    print("No Record Found")
```

## Output:

```
===== RESTART: Z:\sql 18.py =====
Connection succesful.....
(2, 'ANU')
>>>
===== RESTART: Z:\sql 18.py =====
Connection succesful.....
No Record Found
>>> |
```

19) Write a python Program to connect with database, insert records entered by the user in a table and display all the records.

```
*sql 18.py - Z:/sql 18.py (3.7.3)*
File Edit Format Run Options Window Help

import mysql.connector
con=mysql.connector.connect(host="localhost", user="root", password="1234", database='file', charset='utf8')
if con.is_connected():
    print("Connection succesful.....")
else:
    print("Ooops....could not connect")

cur=con.cursor()
r=int(input("Enter Roll No : "))
n=input("Enter Name : ")
query="insert into student values\
(%s, '%s' )"%(r,n)
print(query)
cur.execute(query)
con.commit()
print("Records inserted successfully.....")

cur.execute("select * from student")
data=cur.fetchall()
print(data)
```

### Output

```
>>>
===== RESTART: Z:/sql 18.py =====
Connection succesful.....
Enter Roll No : 4
Enter Name : kavi
insert into student values(4, 'kavi' )
Records inserted successfully.....
[[1, 'MANU'), (2, 'ANU'), (3, 'PRIYA'), (4, 'kavi')]
>>> |
```

20) Write a python program to update the student MARK on table based on the Rollno given by the user . If record not found display the appropriate message.

**DATABASE : FILE**

**TABLE NAME : STUDENT**

```
File Edit Format Run Options Window Help

import mysql.connector as my
a=my.connect(host='localhost',user='root',password='1234',database='file')
ans='y'
while ans.lower()=='y':
    b=a.cursor()
    r=int(input("ENTER THE ROLLNO YOU WANT TO UPDATE:"))
    q1="select * from student where Rno={}".format(r)
    b.execute(q1)
    e=b.fetchall()
    d=b.rowcount
    if d!=0:
        nm=int(input('ENTER THE NEW MARK:'))
        q2='UPDATE STUDENT SET MARK={} WHERE Rno={}'.format(nm,r)
        b.execute(q2)
        a.commit()
        print('RECORD UPDATED SUCCESSFULLY WITH NEW MARK')
    else:
        print('RECORD NOT FOUND')
    ans=input('DO YOU WANT TO UPDATE ANOTHER RECORD (Y/N):')
a.close()
```

## Output:

```
>>>
```

```
===== RESTART: Z:\sql20.py =====  
ENTER THE ROLLNO YOU WANT TO UPDATE:3  
ENTER THE NEW MARK:100  
RECORD UPDATED SUCCESSFULLY WITH NEW MARK  
DO YOU WANT TO UPDATE ANOTHER RECORD (Y/N):Y  
ENTER THE ROLLNO YOU WANT TO UPDATE:5  
RECORD NOT FOUND  
DO YOU WANT TO UPDATE ANOTHER RECORD (Y/N):  
>>>|
```

21) Write a python program to delete the particular record from the table based on the Rollno given by the user .  
If record not found display the appropriate message.

**DATABASE : FILE**

**TABLE NAME : STUDENT**

\*sql21.py - Z:/sql21.py (3.7.3)\*

File Edit Format Run Options Window Help

```
import mysql.connector as my  
a=my.connect(host='localhost',user='root',password='1234',database='file')  
ans='y'  
while ans.lower()=='y':  
    b=a.cursor()  
    r=int(input("ENTER THE ROLLNO YOU WANT TO DELETE:"))  
    q1="select * from student where Rno={}".format(r)  
    b.execute(q1)  
    e=b.fetchall()  
    d=b.rowcount  
    if d!=0:  
        q2='DELETE FROM STUDENT WHERE Rno={}'.format(r)  
        b.execute(q2)  
        a.commit()  
        print('RECORD DELETED SUCCESSFULLY')  
    else:  
        print('RECORD NOT FOUND')  
    ans=input('DO YOU WANT TO DELETE ANOTHER RECORD (Y/N):')  
a.close()
```

## Output:

```
>>>
```

```
===== RESTART: Z:/sql21.py =====  
ENTER THE ROLLNO YOU WANT TO DELETE:1  
RECORD DELETED SUCCESSFULLY  
DO YOU WANT TO DELETE ANOTHER RECORD (Y/N):Y  
ENTER THE ROLLNO YOU WANT TO DELETE:5  
RECORD NOT FOUND  
DO YOU WANT TO DELETE ANOTHER RECORD (Y/N):N  
>>>|
```

\*\*\*\*\*