

## PYTHON LAB PROGRAMS

### Lab Program-1:

**Write a program to check the given number is belongs to Fibonacci series or not.**

```
print("Enter a number:")
n=int(input())
f1=0
f2=1
i=1
flag=0
if n==0:
    flag=1
else:
    while(i<=n):
        f3=f1+f2
        if n==f3:
            flag=1
            break
        else:
            f1=f2
            f2=f3
        i=f3
if flag==1:
    print("Given number is in Fibonacci Series")
else:
    print("Given number is not in Fibonacci Series")
```

### Lab Program-2:

**Write a program to print the multiplication table for a given number.**

```
print("Enter a number:")
n=int(input())
i=1
while i<=10:
    print(n,"*",i,"=",n*i)
    i=i+1
```

**Lab Program-3:****Write a program to find the sum of n natural numbers.**

```
print("Enter a number:")
n=int(input())
i=1
sum=0
while i<=n:
    sum=sum+i
    i=i+1
print("Sum of n natural numbers = ",sum)
```

**Lab Program-4:****Write a program to check the given number is prime number or not.**

```
print("Enter a number:")
n=int(input())
flag=0
i=2
while i<n:
    if n%i==0:
        flag=1
        break
    i=i+1
if flag==0:
    print("Given number is PRIME NUMBER")
else:
    print("Given number is NOT-PRIME NUMBER")
```

**Lab Program-5****Write a program to create a simple calculator.**

```
print("Enter two numbers:")
a=int(input())
b=int(input())
print("Enter Operator(+,-,*,/,%): ")
opr=input()
if opr=='+':
    print(a,"+",b,"=",a+b)
elif opr=='-':
    print(a,"-",b,"=",a-b)
elif opr=='*':
```

---

```
    print(a,"*",b,"=", (a*b))
elif opr=='/':
    print(a,"/",b,"=", (a/b))
elif opr=='%':
    print(a,"%",b,"=", (a%b))
else:
    print("Invalid Operator")
```

### Lab Program-6

**Write a program to solve Quadratic Equation.**

```
import math
print("Enter a,b,c values:")
a=int(input())
b=int(input())
c=int(input())
d=(b*b)-(4*a*c)
if d<0:
    print("Roots are Imaginary")
elif d==0:
    print("Roots are Equal")
    x=-b/(2*a)
    print("Solution=",x)
else:
    print("Roots are Real and Distinct")
    x1=(-b+math.sqrt(b*b-4*a*c))/(2*a)
    x2=(-b - math.sqrt(b * b - 4 * a * c)) / (2 * a)
    print("Solution 1=",x1)
    print("Solution 2=",x2)
```

### Lab Program-7

**Write a program to demonstrate Exceptions in Python.**

```
print("Enter two numbers for division:")
a=int(input())
b=int(input())
try:
    div=a/b
except:
    print("You are trying to divide a number by zero")
else:
    print("Division=", div)
```

---

finally:

```
print("This is Division Program")
```

### **Lab Program-8:**

**Write a program to perform string functions.**

```
print("Enter a string:")
str=input()
print("Given String = ",str)
print("Length of Given string=",len(str))
print("Lower case conversion=",str.lower())
print("Upper case conversion=",str.upper())
print("Capitalization of string=",str.capitalize())
print("Given string is Digit:",str.isdigit())
print("Given string is Alphabetic:",str.isalpha())
print("Given string is Alpha-Numeric:",str.isalnum())
```

### **Lab Program-9**

**Write a program to perform all the list functions using lists.**

```
list=[50,40,10,20,30]
print("Length of List=",len(list))
print("Sum of all the elements=",sum(list))
print("Minimum element=",min(list))
print("Maximum element=",max(list))
print("Sorted List=",sorted(list))
list.append(60)
print("After adding another element=",list)
list.remove(40)
print("After removing an element=",list)
```

### **Lab Program-10**

**Write a program to implement Stacks using Lists.**

```
stack=[]
def push():
    print("Enter element to push:")
    ele=input()
    stack.append(ele)
```

```
def pop():
    top=len(stack)
    if top==0:
        print("Stack is UNDERFLOW")
    else:
        print("Popped Element=",stack.pop())

def display():
    top=len(stack)
    if top==0:
        print("Stack is UNDERFLOW")
    else:
        print(stack,"<-top")

while(1):
    print("1.PUSH\n2.POP\n3.DISPLAY\n4.EXIT")
    print("Enter your choice:")
    ch=int(input())
    if ch==1:
        push()
    elif ch==2:
        pop()
    elif ch==3:
        display()
    elif ch==4:
        exit()
    else:
        print("Invalid Choice")
```

### **Lab Program-11**

**Write a program to perform operations on Dictionaries.**

```
dict={ 6:60,7:70,1:10,2:20,3:30,4:40,5:50}
print("Dictionary Elements are:",dict)
print("Length of Dictionary=",len(dict))
dict.update({ 1:5})
print("After Update Dictionary Elements are=",dict)
dict2=dict.copy()
```

```
print("After copy of Dictionary, dict2=",dict2)
dict.pop(3)
print("After Removing an element=",dict)
print("After Sorted Dictionary Elements=",sorted(dict))
dict.clear()
print("After clear Dictionary=",dict)
```

## Lab Program-12

**Write a program to read and write data into a file.**

```
print("Enter data to insert into file:")
str=input()
fileptr=open("d:/abc.txt","w")
fileptr.write(str)
fileptr.close()
```

```
fileptr = open("d:/abc.txt","r")
str = fileptr.read()
print("Entered data=",str)
fileptr.close()
```

## Lab Program-13

**Write a program to create GUI using TKinter Methods.**

```
from tkinter import *
top = Tk()
top.geometry("400x250")
name = Label(top, text = "Name").place(x = 30,y = 50)
email = Label(top, text = "Email").place(x = 30, y = 90)
password = Label(top, text = "Password").place(x = 30, y = 130)
e1 = Entry(top).place(x = 80, y = 50)
e2 = Entry(top).place(x = 80, y = 90)
e3 = Entry(top).place(x = 95, y = 130)
submit = Button(top, text = "Submit").place(x=110, y=160)
top.mainloop()
```

---

**Lab Program-14:**

**Write a program to create SQLite Database and Perform Operations on Tables.**

```
import sqlite3
conn = sqlite3.connect('vvfgc.db')
conn.execute("CREATE TABLE EMPLOYEE
              (ID INT PRIMARY KEY NOT NULL,
               NAME TEXT NOT NULL,
               AGE INT NOT NULL);")
print("Table Created")

conn.execute("INSERT INTO EMPLOYEE (ID,NAME,AGE) \
              VALUES (1, 'Raju', 32)");
conn.execute("INSERT INTO EMPLOYEE (ID,NAME,AGE) \
              VALUES (2, 'Ajay', 30)");
print("Records Inserted")

cursor = conn.execute("SELECT id, name,age from EMPLOYEE")
for row in cursor:
    print("ID = ", row[0])
    print("NAME = ", row[1])
    print("AGE = ", row[2])

conn.commit()
conn.close()
```

**Lab Program-15:**

**Write a program to implement sequential search.**

```
def lsearch(list,n,key):
    for i in range(0, n):
        if (list[i] == key):
            return i+1
    return -1

print("Enter no. of elements:")
n=int(input())
list=[]
```

---

```
i=0
print("Enter elements:")
while(i<n):
    ele=input()
    list.append(ele)
    i=i+1

print("Enter key element to search:")
key=input()
flag = lsearch(list, n, key)
if (flag == -1):
    print("Key Element not found")
else:
    print("Element found at position: ", flag)
```

**Lab Program-16:**

**Write a program to create dataframe from excel sheet using Pandas and perform operations on DataFrames.**

```
import pandas as pd
df = pd.DataFrame({'Name': ['John', 'Smith', 'Alex', 'James', 'Peter'],
                  'Age': [25, 30, 27, 22, 32]})

with pd.ExcelWriter('d:/sample.xlsx') as writer:
    df.to_excel(writer, sheet_name='Sheet1', index=False)

df = pd.read_excel('d:/sample.xlsx', sheet_name='Sheet1')
print(df)
```

**Lab Program-17:**

**Write a program to create array using NumPy and perform operations on Arrays.**

```
import numpy as np
arr=np.array([[1,2,3],[4,5,6]])
print("Array Elements are:\n",arr)
print("Array Dimension:",arr.ndim)
x=np.arange(12)
y=np.reshape(x, (4,3))
print("One Dimension:",x)
```

---



---

```
print("Converted to Two Dimension:\n",y)
s = slice(2,7,2)
print("After Slicing:",x[s])
```

### Lab Program-18

**Write a program to draw a line chart and bar chart using Matplotlib.**

#### # Line Chart

```
import numpy as np
import matplotlib.pyplot as plt
x = np.array([2,4,6,8,10])
y = np.array([2,4,6,8,10])
plt.plot(x, y)
plt.show()
```

#### #Bar Chart

```
import matplotlib.pyplot as plt
langs = ['C', 'C++', 'Java', 'Python', 'PHP']
students = [23,17,35,29,12]
plt.bar(langs,students)
plt.show()
```

### Regular Expression

**Definition:** A regular expression is a pattern search or matching in a string.

A regular expression is a special sequence of characters that helps you to match or find other strings or sets of strings, using a specialized syntax held in a pattern.

findall(): This method is used to search the particular string in a main string.

search(): This method is used to search the two or more strings in a main string.

### Lab Program-19

**Write a program to demonstrate basic regular expression.**

```
import re
s = "Today is the wonderful day"
```

```
print(s)
print("Enter Text to match:")
str=input()
x = re.findall(str,s)
if (x):
    print("Given text is MATCHED")
else:
    print("Given text is NOT-MATCHED")
```