

1a

Hello world

1B

Enter your name Rahul
Enter your age 18

your name is Rahul
your age is 18

The year you will turn 100 year old is: "2105

Practical :- 1

Aim :- Basic Python Programming

1A

Aim

→ write a program to Display Hello world on the Screen
print ('Hello world')

1B

Aim:

write a program to accept user name And Age print a message Address To Them The year that They will turn 100 years old

→
import datetime
a = input ("enter your name ")
age = input ("enter your age ")
print ("your name is ", a)
print ("your age is ", age)
b = 100 - age
c = datetime.datetime.now().year
d = c + b
print ("the year you will turn to 100 year old is : ", d)

1C

Enter the Number : 4
The Number is Even

Enter the Number : 5
The Number is Odd

1D

Enter 1st Number : 4

Enter 2nd Number : 2

Addition is : 6

Subtraction is : 2

Multiply is : 8

Division is : 2

1C

Aim:-

write a program to Accept a Number from user And check whether the Number is even or odd

\rightarrow value = int(input("Enter the Number"))
if (value % 2 == 0):

 print("The Number is even")

else:
 print("The Number is odd")

1D

Aim:-

write a program to accept two Numbers from the user and perform Basic Arithmetic operations

\rightarrow a = int(input("Enter 1st Number"))
b = int(input("Enter 2nd Number"))
print("Addition is:", a+b)
print("Subtraction is:", a-b)
print("Multiply is:", a*b)
print("Division is:", a/b)

1E
a
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

1F

Enter the Number: 1426

The Reverse Number: 6241

realme Shot by R@hul

2023/10/27 20:21

1E

Aim:-

Write A program To generate Financial Series

→

a=0

b=1

c=1

print(a)

print(b)

print(c)

while (a<50):

a=b

b=c

c=a+b

print(c)

1F

Write a program to reverse the given number.

→ A = int(input("Enter the Number"))

for

n = int(input("enter the No. number"))

rev - n = 0

while (n>0):

remainder = n % 10

rev = rev - n * 10 + remainder

n = n // 10

print("The reverse number is: rev-n")



FOR EDUCATIONAL USE

16.

Enter a Number 153

153 is an Armstrong Number

Enter a Number 127

127 is not a Armstrong Number

1H

Enter Any Number 16461

Number is palindrom

Enter any Number 17642

Number is Not palindrom

16.

Aim:-

Write a program to accept Number from the user And check whether it is an Armstrong Value or Not

$\rightarrow n = \text{int}(\text{input}("enter a number"))$

$s = 0$

$t = n$

while $t > 0$:

$d = t \% 10$

$s = s + d^3 \quad \# s = s + d * d * d$

$t // = 10$

if $n == s$

print(n, "is an Armstrong number")

else:

print(n, "is not an Armstrong number")

1H

Aim:-

Write a program to Accept the Value from the User and check for palindrome

$\rightarrow a = \text{int}(\text{input}("enter any number"))$

$temp = a$

$rem = 1$

$rev = 0$

while ($a > 0$):

$rem = a \% 10$

$rev = rev * 10 + rem$

```
a=a//10  
print ("Reverse : ", rev)  
print ("Original : ", temp)  
if (rev == temp):  
    print ("The number is palindrome")  
else:  
    print ("Number is Not palindrome")
```

18

Aim:

Write a program to accept the int from user and calculate factorial of given value:

```
→ a = int(input("Enter any Number: "))  
fact = 1  
for i in range(1, a+1):  
    fact = fact * i  
print ("The Factorial is", fact)
```

FOR EDUCATIONAL USE

AJ

Enter a character

BJ 7

5

Practical-2

Aim: User Defined Functions

AJ WAP that defines the function which takes a character as argument, and returns true if it's a vowel otherwise false

-> Def def vowel(a):
if (a == 'a' or a == 'e' or a == 'i' or a == 'u' or
a == 'o'):
 return True
else:
 return False

```
a = input("Enter a character\n")  
print(vowel(a))
```

B) Define a function that computes length of a given sequence

->>> def length(a):
flag = 0
for i in a:
 flag += 1
print("the length of the given sequence is:", flag)

```
length('SYBSCIT')  
length([1, 2, 4, 6, 7])
```

FOR EDUCATIONAL USE

c) Define a Function Histogram () that takes a list of Integers and print a Histogram on the Screen.

```
->>> def histogram (*a):
    for i in a
        print ('*' * i)
histogram (1, 2, 3)
```

a)

Enter a String & check if it is a palindrome

M O M

It is a palindrome

b) 2

Practical - 3

Aim : Working with Strings

A) define a function to check whether the given string is a palindrome or Not.

-->>>

```
def palindrome(a):
    temp = a
    rev = a[::-1]
    if (rev == temp):
        print('It is a palindrome')
    else:
        print('It is Not a palindrome')
```

~~a = input('Enter a String')~~
~~temp = a~~
a = input('Enter a string')
temp = a
palindrome(a)

B)

WAP to accept one string and the character from user and Display whether the character is available string or not if available Display the no. of occurrence of the same character in the string

-->>>

```
Counter = 0
S = "programming"
char = "m"
for i in S:
    Counter += 1
    print(Counter)
```

iQU

FOR EDUCATIONAL USE

C]

- ① Enter any String Rahul
② Enter any String Chaurasiya

③ Enter first String Rahul
Enter Second String Chaurasiya

Rahul Rahul Rahul

④ Enter first String Rahul
Enter Second String Chaurasiya

Rah

⑤ Enter first String Rahul
Enter Second String

Ahu1

⑥ Enter the first String Bigw poketi bao

Enter the Second String Bossi

3

4

a = input("Enter the first string")
b = input("Enter the second string")
print(a + b)

C]

WAP to accept two string from the user and perform
following operation:

>>> a = input("Enter any String")
b = input("Enter any String")
print(a+b)

⑦ Repeat the first string for 3 times

a = input("Enter First String")
b = input("Enter Second String")
print(a * 3)

⑧ Display first three character from

a = input("Enter First String")
b = input("Enter Second String")
print(b[0:3])

⑨ Reverse the first String

>>> a = input("Enter first String")
b = input("Enter Second String")
print(a[::-1])

⑩ Display the length of Both Strings

>>> a = input("Enter the first String")
b = input("Enter the Second String")
print(len(a))
print(len(b))

Enter String : Rahul Chaurasiya

If it is not a pangram.

(OR)

Enter String : The quick brown fox jumps over
the lazy dogs

If it is a pangram.

Q) WAP to check whether the given string is pangram or not.

>>> flag = 0

S = input("Enter String :")

alphabet = "abcdefghijklmnopqrstuvwxyz"

for char in alphabet:

if char not in S.lower():

flag = 0

else :

flag = 1

if (flag == 1):

print("It is a pangram")

else :

print("It is not a pangram")

1] True

for i in range(0, len(list1)): if list1[i] == 2:
 for j in range(0, len(list2)): if list2[j] == 2:
 return True
return False

Practical :- 4

Aim:-

4A

Aim:-

Write function that takes two list and returns True if they atleast have one Common Number

Def Common-num(list1, list2):

For i in list1:
 For j in list2:
 if (i == j):
 return True

list1 = [1, 34, 56, 78, 56]

list2 = [1, 56, 45, 90, 35, 79]

print(list1)

print(list2)

print(Common-num(list1, list2))

4B

Aim

Define a list and perform following operation

1. Find length of list

2. Print all element which are less than 5

3. Display 4th and 7th Element from a list

4. Change 5th Element from the given list

5. Display all element in reverse order

b. 1
2
3
40
70

[10, 20, 30, 40, 90, 60, 70, 80, 1, 2, 3]
[3, 2, 1, 80, 70, 60, 90, 40, 30, 20, 10]

c.

[10, 20, 30, 40, 50]
[10, 20, 30, 40, 50]

→ a=[10, 20, 30, 40, 50, 60, 70, 80, 1, 2, 3]
print(len(a))
for i in a:
if (i<5)
print(i)

print(a[3])
print(a[6])
a[4]=90
print(a)
print(a[::-1])

4C

Aim

Write python program to Clone or Copy list

a=[10, 20, 30, 40, 50]
print(a)
b=a
print(b)

FOR EDUCATIONAL USE

5A

Enter a name
Rahul

Enter a Rollno
201

Enter a class
SY

```
{'Name': 'Rahul', 'Rollno': 201, 'Class': 'SY'}
```

Name : Rahul
Roll no : 201
Class : SY

```
(a) tuple  
o=d  
(d) tuple
```

Practical 5

Aim :-

Working with dictionary and tuple

5A

Aim :-

Write a program to accept the Student information from user and store it in a Dictionary Roll no, Name, program and display the Student information from Dictionary in proper format

$a = \{ \}$

```
name = input("enter a name \n")  
rollno = input("int(input("enter rollno \n"))  
Std = input("enter your class \n")  
a['name'] = name  
a['Rollno'] = rollno  
a['class'] = Std
```

print(a)

```
for i in a:  
    print(i, ":", a[i])
```

5B

Aim

Write program to Create the tuple to store Employee Details for the five Employee. and Display the Employee Details in tabular format.

5bJ

Employee details

EID	Ename	Desg	Dep	smif
1	Rahul	Data Scientist	IT	2020-2021
2	Rupesh	Data Analytics	IT	2018-2019
3	Vivek	Data Engineer	IT	2019-2020
4	Rohit	Data Administrator	IT	2019-2020
5	Ash	Data Architect	IT	2019-2020

5C

bmw
SUV
1990

```
emp=[{1,'Rahul','Data Scientist','IT'),  
(2,'Rupesh','Data Analytics','IT'),  
(3,'Vivek','Data Engineer','IT'),  
(4,'Rohit','Data Administrator','IT'),  
(5,'Ash','Data Architect','IT'),
```

```
print("Employee details:\n")  
print("EID Ename Desg Dept:\n")  
for i in emp:  
    print(i[0]," ",i[1]," ",i[2]," ",i[3])
```

5C

Aim

Create a dictionary that stores Vechical Details like, brand model and year and perform following operation

1. Display all the values from a dictionary
2. Update the dictionary by adding one more feature of Vechical
3. Remove year element from dictionary
4. Make a copy of Vechical dictionary
5. Remove last element from the copied dictionary
6. Remove all the element from dictionary
7. Delete the dictionary object.

```
→ Vec1={brand:'bmw',model:'SUV',year:1990}  
for i in Vec1:  
    print(Vec1[i])
```

```
{'brand': 'bmw', 'model': 'Suv', 'year': 1990, 'color': 'blue'}  
{'brand': 'bmw', 'model': 'Suv', 'color': 'black'}  
{'brand': 'bmw', 'model': 'Suv', 'color': 'black'}  
{'brand': 'bmw', 'model': 'Suv'}  
S?
```

```
Traceback (most recent call last):  
  file "C:\Users\S1214v\Downloads\Project 5-C(1).py"  
    line 33, in <module>  
      print(v)  
NameError: name 'v' is not defined
```

```
vec1['color'] = 'black'  
print(vec1)
```

```
del vec1['year']  
print(vec1)
```

```
v = vec1  
print(v)
```

```
v.pop('color')  
print(v)
```

```
v.clear()  
print(v)
```

```
def v  
print(v)
```

6A

File Created

Content Written Successfull

6B)

Append Successfull

Practice 6A

Aim

```
fobj = open ("Syt.txt", "w")  
print ('file created')
```

```
fobj.write ("welcome to file Handling \n")  
print ("Content Written Successfull")  
fobj.close()
```

Practice 6B

Aim

```
fobj = open ("Syt.txt", "a")  
fobj.write ("\n the new Content is append")  
fobj.close()
```

6C

welcome to file Handling
we are learning file handing through python

line 1
line 2
line 3

6D

[welcome to file Manding -], [we are learning through python,] ['lines1', 'lines2', 'lines3'] []

Enter the last line you want to Print 3
line 1
line 2
line 3

Practical 6C

Aim:

```
fobj = open ("Sgit.txt", "r")  
print (fobj.read ())  
fobj.close ()
```

Practical 6D

Aim:

```
fobj = open ("Sgit.txt", "r")  
line = fobj.read ().splitlines ()  
print (line)  
n = int (input ("enter the last line you want to head"))  
for i in range (1, n):  
    print (line [-i])  
fobj.close ()
```

Parent Class

Brand Name : BMW

Color : black

Brand Name : Audi

Color : white

Child 1 of vehicle

Brand Name : Mercedes

Color : Gray

Child 2 of vehicle

Display

Honda black Zee A4

Child of Bike

Brand Name : Stylish

Color : purple.

Practical 7



OOP WAP to Implement Inheritance with following
Types ① Single Inheritance

② Multiple Inheritance

③ Multilevel Inheritance

Polymorphism through Method Overriding

Class vehicle

```
print("In parent class\n")
```

```
def __init__(self, brand, color):
```

```
    self.brand = brand
```

```
    self.color = color
```

```
def display(self):
```

```
    print("Brand Name:", self.brand)
```

```
    print("Color : ", self.color)
```

```
P1 = Vehicle("BMW", "black")
```

```
P2 = Vehicle("Audi", "white")
```

```
P1.display()
```

```
P2.display()
```

Class Car(vehicle):

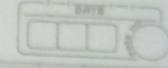
```
    print("In child 1 of Vehicle.")
```

```
pass
```

```
S1 = Car("Mercedes", "Gray")
```

```
S1.display()
```

FOR EDUCATIONAL USE



Class bike (Vehicle):

```
print ("In child 2 of Vehicle")
def __init__(self, brand, Color, Speed, model):
    self.Speed = Speed
    self.Model = Model
```

```
Vehicle.__init__(self, brand, Color)
def display(self):
    print ("Display")
```

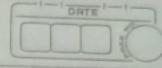
```
def Show(self):
    print (self.brand, self.Color, self.Speed, self.model)
```

```
b1 = bike ('Honda', 'black', 200, 'A4')
b1.display()
b1.Show()
```

Class cycle (bike):

```
pass
print ("In child of Bike")
f1 = Car ("Stylish", "Purple")
f1.display()
```

Practical 8



Aim: Create a Modular Shapes which Consist of Methods that Calculates area of Circle, Square and Rectangle

WAP to use the Method from the Shapes modul.

→ (i) Create a first file (File name → Shapes)

import math

def Square (Side):

return Side ** 2.

def Rectangle (length, breadth):
return length * Breadth

def Circle (radius):

return math.pi * radius ** 2

(ii) Create a Seconds file (file name → Shape demo)

>>>

import Shapes

print (Shapes.Square(2))

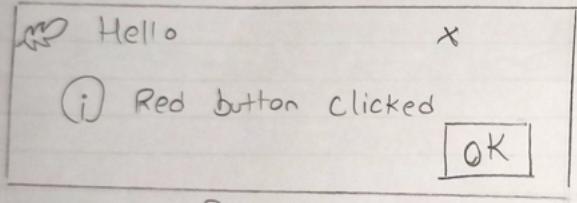
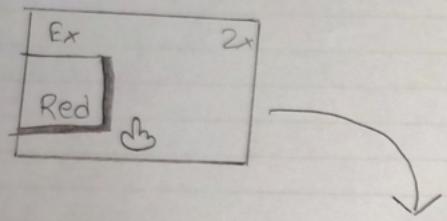
print (Shapes.Circle(6))

print (Shapes.Rectangle(4,6))

4

113.0973355

24



== == RESTART:D://Rahul:/ GUI practiced

Button clicked!!!

Practical 9

```
from tkinter import *  
import tkinter  
from tkinter import messagebox
```

```
top = TK()  
top.geometry ("200x100")  
def func():  
    messagebox.showinfo("Hello", "Red Button Clicked")  
    # print ("Button Clicked !!!")
```

```
b1= Button (top, text = "Red", command = func, activeforeground = "Red", activebackground = "Pink",  
            padx = 10 )
```

```
b1.pack  
b1.pack (Side = LEFT)
```

```
top.mainloop()
```