

1.

Write a program to read and print values of variables of different data types.

```
str = "flashroot"
num = 100
rc = 1.7
complex_num = 19j + 2j
player_list = ["VR", "RS", "IP", "PR"]
bat_name = ("A", "B", "C", "D")
duckworth = {"A": "a", "B": "b", "C": "c", "D": "d"}
lbev = ('Y', 'N', 'T', 'F')
print("The type is:", type(str))
print("The type is:", type(num))
print("The type is:", type(rc))
print("The type is:", type(complex_num))
print("The type is:", type(player_list))
print("The type is:", type(bat_name))
print("The type is:", type(duckworth))
print("The type is:", type(lbev))
```

Output:

The type is : <class 'str'>

The type is : <class 'int'>

The type is : <class 'float'>

The type is : <class 'complex'>

The type is : <class 'list'>

The type is : <class 'tuple'>

The type is : <class 'dict'>

The type is : <class 'set'>

2.

# Write a program to perform addition, subtraction, multiplication,  
# division and modulo division on two integers.

```
print("Enter First Number: ")  
numOne = int(input())  
print("Enter Second Number: ")  
numTwo = int(input())  
print("\nAddition")  
res = numOne + numTwo  
print("\nAddition Result = ", res)  
res = numOne - numTwo  
print("Subtraction Result = ", res)  
res = numOne * numTwo  
print("Multiplication Result = ", res)  
res = numOne // numTwo  
print("Division Result = ", res)  
res = numOne % numTwo  
print("Modulo division Result = ", res).
```

Output:

Enter First Number:

10

Enter Second Number:

2

Addition Result = 12

Subtraction Result = 8

Multiplication Result = 20

Division Result = 5.0

Modulo division Result = 0

3.

# Write a program to input two numbers and check whether they are equal or not

a = 8

b = 2

if (a == b):

    print('The two numbers are equal.')

else:

    print('The two numbers are not equal.')

Output:

The two numbers are not equal



4.

# Write a program that prompts user to enter a character  
# (O,A,B,C,F). Then using if-else-if-else construct display  
# Outstanding, Very Good, Average and Fail respectively

if --name-- = "--~~main~~--"

marks = [95, 88, 98, 93, 92, 96]

max-marks = len(marks) \* 100

total = 0

grade = 'F'

for i in range(len(marks)):

    total += marks[i]

percentage = ((total) / max-marks) \* 100

if (percentage >= 90):

    grade = 'O(Outstanding)'

else :

    if (percentage >= 80 and

        percentage <= 89):

        grade = 'A(Very Good)'

else :

    if (percentage >= 60 and

        percentage <= 79):

        grade = 'B(Good)'

else:

if(percentage >= 33 and  
percentage <= 59):

grade = 'C(Average)',

else

grade = 'F(Fail)',

print(grade)

Output:

A

5.

# Write a program to print Fibonacci series using recursion.

```
def recurr-fibo(n):
    if n <= 1:
        return n
    else:
        return(recurr-fibo(n-1)+recurr-fibo(n-2))
```

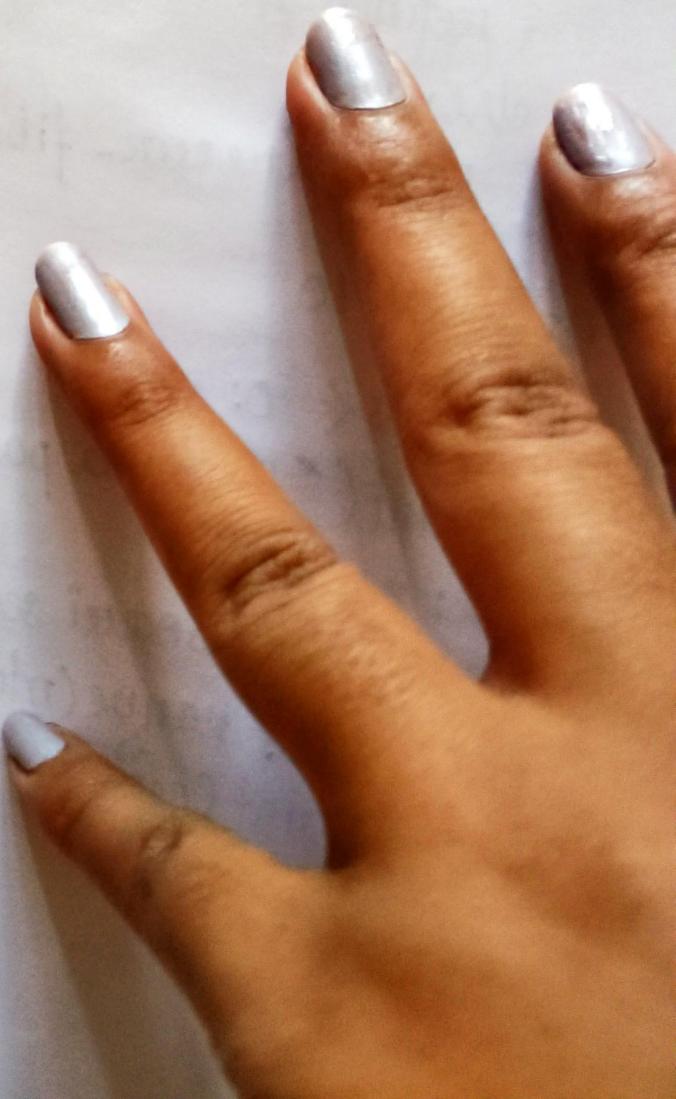
nterms = 10

```
if nterms <= 0:
    print("Enter a positive integer")
else:
    print("Fibonacci sequence:")
    for i in range(nterms):
        print(recurr-fibo(i))
```

Output:

Fibonacci sequence:

0  
1  
1  
2  
3  
5  
8  
13  
21  
34.



6.

# Write a program that prints absolute value, square root and  
# cube root of a number.

from numpy import cbrt

```
n=int(input('Enter a number: '))
print('Square root of the number:', n**(1/2))
print('Cube root of the number:', cbrt(n))
print('Absolute value of the number:', abs(n))
```

Output:-

Enter a number: 49

Square root of the number: 7.0

Cube root of the number: 3.65.

Absolute value of the number: 49.

7.  
# Write a Program that finds the greatest of three given numbers using  
# functions. pass three arguments.

$$a = 10$$

$$b = 1$$

$$c = 17$$

print(max(a, b, c)).

Output:

It

is going to take me all day to  
draw this

Output:

17



# Write a program to get a string made of the first 2 and last 2  
# characters from a given string. If the string length is less than 2,  
# return empty string.

```
def string_both_ends(str):  
    if len(str) < 2:  
        return ''  
    return str[0:2] + str[-2:]  
print(string_both_ends('iti'))  
print(string_both_ends('iti01'))  
print(string_both_ends('i'))
```

O<sub>c</sub>

Output:

itti  
itol.

10.

# Write a program to find the resolution of an image.

```
def jpeg-res(filename):
    with open(filename, 'rb') as img-file:
        img-file.seek(163)
        a = img-file.read(2)
        height = (a[0] << 8 + a[1])
        a = img-file.read(2)
        width = (a[0] << 8) + a[1]
    print("The resolution of the image is ", width, "x", height)
    jpeg-res("Test.jpg")
```

Output:

The resolution of the image is  $280 \times 280$