

## Expt 1 Basics of Python Programing

To familiarize with input, output and if- else statements

- a. Program to print a message  
# python program to print "Hello World"  
`print("Hello World")`

### Output

Hello World

- b. Program to enter your name  
`name = input("Enter your name")`  
`print("My name is:", name)`  
Output  
Enter your name sree  
My name is: sree
- c. Program to check whether the given number is odd or even  
`num = int(input("Enter the number-"))`  
If `num % 2 == 0`:  
    `print(num, "is even")`  
else:  
    `print(num, "is odd")`

### Output

Enter the number-7

7 is odd

Enter the number-8

8 is even

## Expt2 Data Types in Python

To familiar with data types in Python

- a. Basics of Numeric data types  
`a = 5`  
`print("Type of a: ", type(a))`  
`b = 5.0`  
`print("\nType of b: ", type(b))`  
`c = 2 + 4j`  
`print("\nType of c: ", type(c))`  
`x = 5`  
`y = 10`  
`print(bool(x==y))`  
`x = None`  
`print(bool(x))`  
`x = ()`  
`print(bool(x))`

```
print(bool(y))
```

### Output

```
Type of a: <class 'int'>
Type of b: <class 'float'>
Type of c: <class 'complex'>
False
False
False
True
```

### Expt 3: Develop a calculator using Python

```
# Take input from the user
select = int(input("Select operations form 1, 2, 3, 4 :"))

num_1 = int(input("Enter first number: "))
num_2 = int(input("Enter second number: "))

if select == 1:
    num_3 = num_1 + num_2
    print(num_1, "+", num_2, "=", num_3)

elif select == 2:
    num_3 = num_1 - num_2
    print(num_1, "-", num_2, "=", num_3)
elif select == 3:
    num_3 = num_1 * num_2
    print(num_1, "*", num_2, "=", num_3)

elif select == 4:
    num_3 = num_1 / num_2
    print(num_1, "/", num_2, "=", num_3)
else:
    print("Invalid input")
```

### Output

```
Select operations form 1, 2, 3, 4 :4
Enter first number: 4
Enter second number: 4
4 / 4 = 1.0
```

### Exp t 4: Create, concatenate, and print a string and access a sub-string from a given string

```
# Step 1: Create Strings
```

```
string1 = "Python"
```

```

string2 = "Programing"

# Step 2: Concatenate Strings

con_string = string1 + " " + string2

# Step 3: Print the Concatenated String

print("Concatenated String:", con_string)

# Step 4: Access a Sub-string

# Let's access the substring from the concatenated string

substring1 = con_string[2:11]

substring2 = con_string[8:-1]

print("Sub String1:", substring1, "& Sub String2:", substring2 )

```

### Output

```

Concatenated String: Python Programing

Sub String1: thon Prog & Sub String2: rogramin

```

### Expt 5: . Familiarize time and date in various formats

```

''' Familiarize time and date in various formats '''

import datetime

# to print any date in default format

print("Enter the details of the date")

y = int(input("Enter the year"))

m = int(input("Enter the month"))

d = int(input("Enter the date"))

x = datetime.datetime(y, m, d)

print(x)

# to print the current date and time

y = datetime.datetime.today()

print("Current date is: ",y)

# To print the date in a format

print(x.strftime("formatted date is :%d,%B,%Y"))'''

%a : Abbreviated weekday name.

%b : Abbreviated month name.

```

%d : Day of the month as a decimal number [01,31].  
%w : Weekday as a number 0-6, 0 is Sunday  
%m : Month as number, 1 is February  
%c : mm/dd/yyyy  
%H : Hour (24-hour clock) as a decimal number [00,23].  
%M : Minute as a decimal number [00,59].  
%S : Second as a decimal number [00,61].  
%Z : Time zone name (no characters if no time zone exists).  
%Y : Year with century as a decimal number."

### Output

```
Enter the year2024
Enter the month9
Enter the date1
2024-09-01 00:00:00
Current date is: 2024-09-02 06:36:16.789459
formatted date is :01,September,2024,
```

Expt 6: **Write a program to create, append, and remove lists in Python using numPy.**

Expt 7: **Programs to find the largest of three numbers.**

```
'''largest of three numbers'''
num1 = int(input("Enter the first number-"))
num2 = int(input("Enter the second number-"))
num3 = int(input("Enter the third number-"))
if num1 > num2:
    if num1 > num3:
        print(num1, "is the largest")
    else:
        print(num3, "1is the largest")
elif num2>num1:
    if num2 > num3:
        print(num2, "is the largest")
```

else:

print(num3, "is the largest")

### Output

**Enter the first number-5**

**Enter the second number-12**

**Enter the third number-6**

**12 is the largest**

**Expt 8:. Convert temperatures to and from Celsius, and Fahrenheit. [Formula:  $c/5 = f-32/9$ ]**

```
''' Write a Python program to convert temperatures to and from Celsius, Fahrenheit.
```

```
[ Formula:  $c/5 = f-32/9$  ] '''
```

```
print("Enter your choice \n")
```

```
print("1.Convert temperatures from Celsius to Fahrenheit \n")
```

```
print("2.Convert temperatures from Fahrenheit to Celsius \n")
```

```
ch = int(input("Choose any Option(1 or 2) : "))
```

```
if ch == 1:
```

```
    print("Convert temperatures from Celsius to Fahrenheit \n")
```

```
    cel = float(input("Enter Temperature in Celsius: "))
```

```
    fahr = (cel*9/5)+32
```

```
    print("Temperature in Fahrenheit = ",fahr)
```

```
elif ch == 2:
```

```
    print("Convert temperatures from Fahrenheit to Celsius \n")
```

```
    fahr = float(input("Enter Temperature in Fahrenheit: "))
```

```
    cel = (fahr-32)*5/9;
```

```
    print("Temperature in Celsius = ",cel)
```

```
else:
```

```
    print("Invalid Option")
```

### Output

**Enter your choice**

1.Convert temperatures from Celsius to Fahrenheit

2.Convert temperatures from Fahrenheit to Celsius

Choose any Option(1 or 2) : 2

Convert temperatures from Fahrenheit to Celsius

Enter Temperature in Fahrenheit: 97

Temperature in Celsius = 36.111111111111114

**Expt 9: Program to construct the stars (\*) pattern, using a nested for loop**

```
''' Program to construct the stars (*) pattern, using a nested for loop '''
```

```
n = int(input("Enter the number of rows-"))
for i in range(n+1):
    for j in range(i):
        print("* ", end="")
    print()
```

**Output**

**Enter the number of rows-5**

```
*
* *
* * *
* * * *
* * * * *
```

**Expt 10:**

**a. Program to check whether the given number is prime or not**

```
def is_prime(n):
    # Check if number is less than or equal to 1
    if n <= 1:
        return False
    # Check if number is 2 or 3, which are prime
    if n <= 3:
        return True
    # Eliminate even numbers and multiples of 3
    if n % 2 == 0 or n % 3 == 0:
        return False

    # Only check up to the square root of the number
    # and skip even numbers and multiples of 3
    i = 5
    while i <= n/2:
        if n % i == 0:
            return False
```

```

        i += 1

    return True

# Get user input
number = int(input("Enter a number: "))

# Check if the number is prime and print the result
if is_prime(number):
    print(number, " is a prime number.")
else:
    print(number, " is not a prime number.")

```

### Output

```

Enter a number: 4
4 is not a prime number.

```

### b. Program that prints prime numbers less than 20

```

"""Write a Python script that prints prime numbers less than 20"""

print("Prime numbers between 1 and 20 are:")
ulmt=20;
for num in range(ulmt):
    # prime numbers are greater than 1
    if num > 1:
        for i in range(2,num):
            if (num % i) == 0:
                break
        else:
            print(num)

```

### Output

```

Prime numbers between 1 and 20 are:

2
3
5
7
11
13
17
19

```

### Expt 11: . Program to find the factorial of a number using Recursion.

```

num = int(input("Enter a number: "))
def factorial(n):
    if n < 0:

```

```

        return "Factorial is not defined for negative numbers."
# Base case: if n is 0 or 1, the factorial is 1
elif n == 0 or n == 1:
    return 1
else:
    return n * factorial(n - 1)

# Calculate the factorial using the recursive function
result = factorial(num)
print(result, "is the factorial of", num)

```

### Output

Enter a number: 5  
120 is the factorial of 5

## 12. Recursive function to add two positive numbers.

```

def rec_sum(a, b):
    # Base case: if b is 0, return a
    if b == 0:
        return a
    else:
        return rec_sum(a + 1, b - 1)

num1 = int(input("Enter the first positive number: "))
num2 = int(input("Enter the second positive number: "))
if num1 < 0 or num2 < 0:
    print("Please enter positive numbers only.")
else:
    result = rec_sum(num1, num2)
    print(num1, " + ", num2, " = ", result)

```

### Result

Enter the first positive number: 23  
Enter the second positive number: 32  
23 + 32 = 55

## 13. Recursive function to multiply two positive numbers

```

def rec_mul(a, b):
    # Base case: if b is 0, return a
    if b == 0:
        return 0
    else:
        return a + rec_mul(a, b - 1)

num1 = int(input("Enter the first positive number: "))
num2 = int(input("Enter the second positive number: "))
if num1 < 0 or num2 < 0:
    print("Please enter positive numbers only.")
else:

```



```
result = rec_mul(num1, num2)
print(num1, " * ", num2, " = ", result)
```

**Result**

```
>>> %Run traila.py
Enter the first positive number: 4
Enter the second positive number: 67
4 * 67 = 268
```

**14. Recursive function to the greatest common divisor of two positive numbers.**