



Python for Data Science - 2305CS303

Lab - 7

Roll No. : 111

Name : Dhara Maru

1. WAP to count simple interest using function.

```
In [1]: def si(p, r, t):  
        return (p * r * t) / 100  
  
        p = float(input("P: "))  
        r = float(input("R: "))  
        t = float(input("T: "))  
        print("SI:", si(p, r, t))
```

SI: 800.0

2. Write a function to calculate BMI given mass and height. (BMI = mass/h**2)

```
In [2]: def bmi(mass, height):  
        return mass / (height ** 2)  
  
        m = float(input("Mass (kg): "))  
        h = float(input("Height (m): "))  
        print("BMI:", bmi(m, h))
```

BMI: 0.0010957703265395574

3. WAP that defines a function to add first n numbers.

```
In [3]: def add_n(n):  
        return n * (n + 1) // 2  
  
        n = int(input("Enter n: "))  
        print("Sum:", add_n(n))
```

Sum: 55

4. WAP to find maximum number from given two numbers using function.

```
In [4]: def max_num(a, b):  
        return a if a > b else b  
  
x = int(input("First: "))  
y = int(input("Second: "))  
print("Max:", max_num(x, y))
```

Max: 55

5. Write a function that returns True if the given string is Palindrome or False otherwise.

```
In [5]: def is_palindrome(s):  
        return s == s[::-1]  
  
txt = input("Enter string: ")  
print(is_palindrome(txt))
```

True

6. Write a function that returns the sum of all the elements of the list.

```
In [6]: def sum_list(lst):  
        return sum(lst)  
  
nums = list(map(int, input("Enter numbers: ").split()))  
print("Sum:", sum_list(nums))
```

Sum: 113

7. WAP that defines a function which returns 1 if the number is prime otherwise return 0.

```
In [7]: def is_prime(n):  
        if n < 2:  
            return 0  
        for i in range(2, int(n**0.5) + 1):  
            if n % i == 0:  
                return 0  
        return 1  
  
num = int(input("Enter number: "))  
print(is_prime(num))
```

1

8. Write a function that returns the list of Prime numbers between given two numbers.

```
In [8]: def prime_range(a, b):  
        return [x for x in range(a, b+1) if is_prime(x)]
```

```
def is_prime(n):
    if n < 2:
        return 0
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return 0
    return 1

start = int(input("Start: "))
end = int(input("End: "))
print("Primes:", prime_range(start, end))
```

Primes: [11, 13, 17, 19]

9. WAP to generate Fibonacci series of N given number using function name fibbo. (e.g. 0 1 1 2 3 5 8...).

```
In [9]: def fibbo(n):
        a, b = 0, 1
        for _ in range(n):
            print(a, end=' ')
            a, b = b, a + b

        num = int(input("Enter N: "))
        fibbo(num)
```

0 1 1 2 3 5 8 13 21 34

10. WAP to find the factorial of a given number using recursion.

```
In [10]: def fact(n):
        if n == 0 or n == 1:
            return 1
        return n * fact(n - 1)

        num = int(input("Enter number: "))
        print("Factorial:", fact(num))
```

Factorial: 120

11. WAP to implement simple calculator using lamda function.

```
In [11]: add = lambda a, b: a + b
        sub = lambda a, b: a - b
        mul = lambda a, b: a * b
        div = lambda a, b: a / b if b != 0 else 'Error'

        a = float(input("First: "))
        b = float(input("Second: "))
        op = input("Op (+ - * /): ")

        calc = {'+': add, '-': sub, '*': mul, '/': div}
        print("Result:", calc[op](a, b) if op in calc else "Invalid op")
```

Result: 57.0

