**24/9/25** **PYTHON**

**Functions:** A function is a block of code which is use to perform a specific task.

We represent a function using '()'.

Uses:

- Reusability of code increases
- Functions make our program more organised and they are more readable
- Reduces no. of lines of code and reduces reptation.

**Types of function:**

- Built-in function: Functions which are predefined or already present in python. Eg: print(), input() etc.
- User defined function: Functions created by user using 'def' keyword.
  Syntax: def fun_name(parameters):

  ```
                  Statements

                  Return value
  ```

- Lambda function: It's the anonymous (nameless) function written in single line using the 'lambda' keyword.

We can use functions in 4 ways:

- Function without input and without return.
- Function with input and without return.
- Function without input and with return.
- Function with input and with return.

1.Function without input and without return: Whatever we do we should do within that function only.

```
Syntax: def fun_name():

                Statements
```

```
Eg: def si1():

        p=float(input())

        t=float(input())

        r=float(input())
```

```
        si=(p*t*r)/100
        print(f’{si} for pa={p},time={t},roi={r}”)
    si1();
```

[We can’t access the variables like p,t,r outside the function since they are local variables, if we try to access them it gives ‘name error’]

2.Function with input and without return.

```
Syntax: def fun_name(p1, p2……pn):
                    Statements
```

```
Eg: def si2(p, t, r):
        si=(p*t*r)/100
        print (f” {si} for pa={p}, time={t}, roi={r}”)
    si2()
```

3.Function without input and with return.

```
Syntax: def fun_name():
                    Statements
                    Return value
```

```
Eg: def si3():
         p=float(input())
         t=float(input())
         r=float(input())
         si=(p*t*r)/100
        print (f” {si} for pa={p}, time={t}, roi={r}”)
         return si3()
```

4.Function with input and with return.

```
Syntax: def fun_name(p1, p2, ……pn):
                    Statements
```

## Return value

```
Eg: def si4(p. t. r):
        si=(p*t*r)/100
       print (f” {si} for pa={p}, time={t}, roi={r}”)
        return si4()
```