A **Python function** is a named block of code that performs a specific task and only runs when it is called

**Built in functions**
**User-defined functions**

```
#1. len() --> its is used to find out length(number of items ) of:
    # Strings
    # list
    # Dictionaries
    #sets
# syntax: len(object)
# print(len("murali   mohan")) # space is also one character
# print(len([10,20,50,60,7,10,52]))
# print(len({"name":"murali","id":1001}))
# type() --> to display trhe data type of any variable
# print(type("murali   mohan")) # space is also one character
# print(type([10,20,50,60,7,10,52]))
# print(type({"name":"murali","id":1001}))
# str() ,int(), float()
# a="100"
# a=int(a)
# print(a)
# print(type(a))
# b=225
# b=str(b)
# print(b)
# print(type(b))
# price=int(input("enter item price "))
# qty=int(input("enter quantity:"))
# print(f"Total is {price * qty}")
# range() --> used to generate sequence of numbers : range is strating 0 by default . for
stop vales its givinig n-1 value
    # syntax:
    #     range(stop)
    #     range(start,stop)
    #     rang(start,stop,step)
# print(list(range(5)))
# print(list(range(2,10+1)))
# print(list(range(1001,1051)))
# print(list(range(2,21,2)))
# for r in range(100):
#     print("processing record:", r)
#sending OTP 5attemps
# for r in range(1,6):
#     input("enter valid Otp")
#     print("attempt:",r)
# userdefined functions:
#  the functions which are created by user b y using def key word
# syntax  --> def function_name ();
#                 statements
# def message():
#     print("welcome")
# message()#/called
# rules to create the function"
    # 1. must begin with def
    #2. function name must follow identify rules (a-z,A-Z,_,numbers)
    #3. indentation is mandatory
    #4.function should be called to execute
# def emp(i):
#     print("name is murali",i)
```

```
# for i in range(10):
#     emp(i)
# def validate_login(username,password):
#     return username=="admin" and password=="1234"
# print(validate_login("admin","12345"))
#parameters and return types: when defining a function , the variables inside () are called
parameter
            # tehy allows us to pass values (arguments) in to function
# in python 4 types of parameters are supported :
# 1.positional parameters
#  the parameters that must be passed in the same order as defined
# the vales is assigend based on the position not on the name
    #  mandatory
    # order matters
    # most commonly used
# def validate_login(username,password):
#     return username=="admin" and password=="1234"
# uname="admin"
# pwd="1234"
# print(validate_login(pwd,uname))
#default parameters:
    # paramter that have default value if no value is passed
    # used to make parameters optional
# must come after positional parameter
def greet(name,message="welcome "):
    print(message ,name)
# greet("murali")
greet("murali","good evening ") #overrides default values


#3. keyword Parameters
    # when calling a function , you must pass parameter  by name=value .
    # order does not matter
# def student(name,age,course):
#     print(name,age,course)
# student(course="DCME",age=22,name='kiran')
# 4. Variable-length parameter
#     when you dont know howmany number of arguments  will be  passed
    # - *arg(nonkeyword variable length)
    #     accept any numbner of optional arguments
    #     Stored as a tuple
# def total(*numbers):
#     print(numbers)
#     print(f"sum = {sum(numbers)}")
# total(10,20,30,40,5024,42,9,82,245,2,55,225,53,2552,52,22,522,2,2,2532,2,32)
    # **kwargs(keyword variable length)
    # accept any number of keyword arguments
    # stored as dictonary
# def user_information(**details):
#     print("\n",details)
# user_information(name="RITHIKA",cource="PYTHON",age=20,mail="rithika@gmail.com")
# user_information(name="kiran",cource="PYTHON")
# def order(cust_name,Item, qty=1,*addons,):
#     print(f"Name : {cust_name}")
#     print(f"item : {Item}")
#     print(f"Quantity : {qty}")
#     print(f"Addons : {addons}") #alt + shift+down arrow to paste the same line below
#     # print(f"Details : {details}")
# order("murali","Pizza",2,"Extra  cheese","OLIVES","large","true")
#return values:
    # used to send output back from the function
    # if you write return in the function it means it is the end of function no other lines
are executed afterr return line
    #function can return multiple values
# def square(x):
#     return x*x
# result= square(10)
# print(result)
# def stats(a,b):
```

```
#     return a+b, a*b
# print(stats(5,6))
#recursive function
    # a function calling itself repeatedly until a condition is met
#    syntax: def function():
#              function()  recursive call
# it must have 2 parts:
            # 1.base case  --> stop recursion
            # 2. recursive case -- function calling itself
#factorial
def fact(n):
    if n == 1: #base case
        return 1
    else:
        return n * fact(n-1) #recusrive call

print(fact(5))
```