* Statements
  - break
  - continue
  - pass.

* Break
  - Terminate 'im between the loop

Eg. code:

a)

```
for i in range (1, 10):
    if (i == 5):
        break
    print (i)
```

b)

```
for i in range (1, 10):
    for j in range (2, 4):
        if (i == j):
            break
        print (i * j)
```

• the break statement terminates the execution of the program before the test expression evaluates to false.

* Continue.
  - Skips the particular test conditions iteration

Eg:

```
for i in range (1, 10):
    if (i == 5):
        continue
    print (i)
```

* Pass

- Temporary passing.
- That does nothing
- null statement
- no operation.

functions.

- Block of code - task that are specific
- length - reduced.
- Code reusability

Types :

a) Built-in functions b) User-defined functions.

Advantages

- Increases readability
- Reduces code length
- Increases reusability.
- Easy to divide and conquer.

a) Built-in function

- readmade functions.

Eg: import(), input(), pow(), set()

b) User-defined function

- Elements
  1. Function definition [keyword : def]
  2. Function call.

① Syntax :

```
def function_Name (Parameters) :
    function_body
```

(2) Syntax:

Function - Name ([arguments])

Q1. write a python program to exchange two values using function.

```
def exchange(a,b):
        a,b = b,a
        print("After exchange")
        print("A= ", a)
        print('B= ', b)
a = input("Enter A : ")
b = input("Enter B : ")
print("Before exchange")
print("A = ", a)
print("B = ", b)
exchange(a,b)
```

Fruitful function.

Q2. Write a python program to find area of circle.

```
def area(r):
        a = math.pi * r**2
return a            # return (3.14 * r**2)
import math
r = int(input("Enter a number : "))
x = area(r)
print("area = ", x)
```