Loops

For Else

```
a=[1,2,3,4,6,7]
for i in a:
        if i%5=0:
                break
else:
        print("Not Found")
```

Functions

Axpino Technologies

Python Programing

Functions

Function Definition

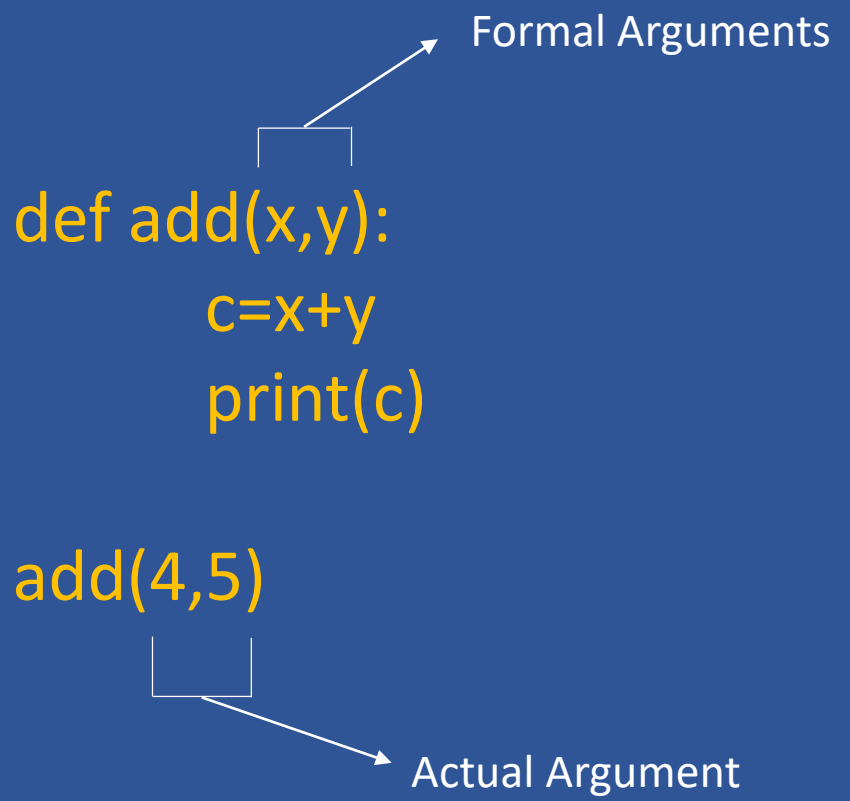
```
def greet():
        print("Hello")
        print("Good Morning")
```

Axpino Technologies

Python Programing

Functions

Passing Parameter to function



Functions

Types of arguments

- Position
- Keyword
- Default
- Variable length

Functions

Position Argument

```
def person(name,age):
    print(name)
    print(age)

person(“Harminder”,29)
```

Functions

Keyword Argument

```
def person(name,age):
        print(name)
        print(age)
person(age=11,name="Harminder")
```

Axpino Technologies

Python Programing

Functions

Default Argument

```
def person(name,age):
        print(name)
        print(age)
person(age=11,name="Harminder")
```

Functions

Variable Length Argument

```
def sum(a,*b):
        for i in b:
                c=a+i
                print(c)

sum(2,4,6)
```

Functions

Keyworded Variable Length Argument

```
def person(a,**b):
        print(a)
        print(b)

person("Harminder",city="faridabad",age=19)
```

Functions

Keyworded Variable Length Argument

```
def person(a,**b):
        print(a)
        for i,j in b.items():
                print(i,j)

person("Harminder",city="faridabad",age=19)
```

Returning values from function

Functions

```
def add(x,y):
        c=x+y
        return c

a=add(4,5)
print(a)
```

Returning multiple values from function

Functions

```
def add_sub(x,y):
        c=x+y
        d=x-y
        return c,d

a,b=add_sub(4,5)
print(a,b)
```

Functions

Global & Local Variables

```
a=10
def hello():
        a=15
        print(a)

hello()

print(a)
```

Functions

Local Variables can only be used inside function

```
def hello():
        a=15
        print(a)

print(a)
```

This code will generate a Error

Functions

Global Variables can be used anywhere in Program

```
a=10
def hello():
        print(a)

hello()
```

Functions

Changing Value of a global Variable

```
a=10
def hello():
        global a
        a=15
        print(a)

hello()
print(a)
```

Functions

Passing List/tuple/set to a function

```
def hello(a,b,c):
        print(a)
        print(b)
        print(c)

a=[1,2,3,4,5]
b=(1,2,3,4,5)
c={1,2,3,4,5}

hello(a,b,c)
```

Functions

Anonymous Function(LAMBDA)

```
f= lambda a,b:a+b

result = f(5,6)

print(result)
```

Functions

Using Filter with lambda

```
nums = [2,3,45,6,7,8,80]

r= filter(lambda n:n%2==0,nums)

for i in r:
        print(i)
```

Functions

Using Map with lambda

```
nums = [2,3,45,6,7,8,80]

r= map(lambda n:n*2,nums)

for i in r:
        print(i)
```

Functions

Using Reduce with lambda

```
from functools import reduce
nums = [2,3,45,6,7,8,80]

r= reduce(lambda a,b:a+b,nums)

print(r)
```

Functions

Creating Modules in Python

```
def add(a,b):
        c=a+b
        return c

def sub(a,b):
        c=a-b
        return c

def mul(a,b):
        c=a*b
        return c
```

Using User Defined Modules in Python

Functions

```
import hello as h

r=h.add(3,4)

print(r)
```

Special Variable

Axpino Technologies

Python Programing

```
import hi as h
def fun1():
        print("This is function 1")
        h.add()

def fun2():
        print("This is function 2")

def main():
        fun1()
        fun2()
main()
```

```
def add():
        print("This is add function")

def mul():
        print("This is mul function")

def hello():
        add()
        mul()

if __name__ == "__main__":
        hello()
```