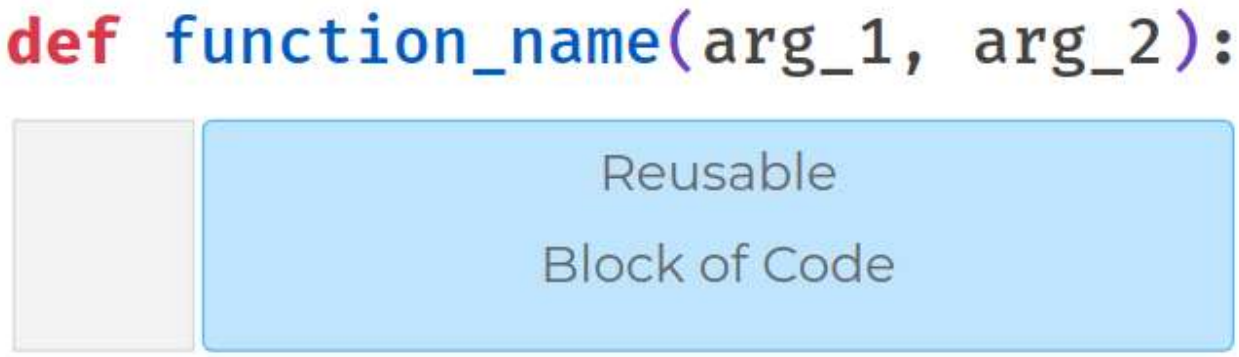
Cheat Sheet

Function Arguments

A function can have more than one argument.



Keyword Arguments

Passing values by their names.

Code

PYTHON

```python
def greet(arg_1, arg_2):
  print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(arg_1=greeting,arg_2=name)
```

Input

```
Good Morning
Ram
```

Output

```
Good Morning Ram
```

Possible Mistakes - Keyword Arguments

Code

PYTHON

```python
def greet(arg_1, arg_2):
    print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(arg_2=name)
```

Input

```
Good Morning
Ram
```

Output

```
TypeError: greet() missing 1 required positional argument: 'arg_1'
```

Positional Arguments

Values can be passed without using argument names.

- These values get assigned according to their position.
- Order of the arguments matters here.

Code

PYTHON

```python
def greet(arg_1, arg_2):
    print(arg_1 + " " + arg_2)
```

```
4  greeting = input()
5  name = input()
6  greet(greeting,name)
```

Input

```
Good Morning
Ram
```

Output

```
Good Morning Ram
```

Possible Mistakes - Positional Arguments

Mistake - 1

Code

PYTHON

```
1  def greet(arg_1, arg_2):
2   print(arg_1 + " " + arg_2)
3
4  greeting = input()
5  name = input()
6  greet(greeting)
```

Input

```
Good Morning
Ram
```

Output

```
TypeError: greet() missing 1 required positional argument: 'arg_2'
```

Mistake - 2

Code

PYTHON

```python
def greet(arg_1, arg_2):
 print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet()
```

Input

```
Good Morning
Ram
```

Output

```
TypeError: greet() missing 2 required positional arguments
```

Default Values

Example - 1

Code

PYTHON

```python
def greet(arg_1="Hi", arg_2="Ram"):
    print(arg_1 + " " + arg_2)

greeting = input()
name = input()
```

```
6  greet()
```

Input

```
Hello
Teja
```

Output

```
Hi Ram
```

Example - 2

Code

PYTHON

```
1  def greet(arg_1="Hi", arg_2="Ram"):
2      print(arg_1 + " " + arg_2)
3
4  greeting = input()
5  name = input()
6  greet(greeting)
```

Input

```
Hello
Teja
```

Output

```
Hello Ram
```

Example - 3

Code

PYTHON

```python
def greet(arg_1="Hi", arg_2="Ram"):
    print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(name)
```

Input

```
Hello
Teja
```

Output

```
Teja Ram
```

Example - 4

Code

PYTHON

```python
def greet(arg_1="Hi", arg_2="Ram"):
    print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(arg_2=name)
```

Input

```
Hello
Teja
```

Output

```
Hi Teja
```

Example - 5

Code

PYTHON

```python
def greet(arg_1="Hi", arg_2):
    print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(arg_2=name)
```

Input

```
Hello
Teja
```

Output

```
SyntaxError:non-default argument follows default argument
```

Non-default arguments cannot follow default arguments.

Example - 6

Code

PYTHON

```python
def greet(arg_2, arg_1="Hi"):
```

```
    print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(arg_2=name)
```

Input

```
Hello
Teja
```

Output

```
Hi Teja
```

Passing Immutable Objects

Code

PYTHON

```
def increment(a):
    a += 1

a = int(input())
increment(a)
print(a)
```

Input

```
5
```

Output

5

Even though variable names are same, they are referring to two different objects.
Changing the value of the variable inside the function will not affect the variable outside.

Submit Feedback