In [3]:
```python
def function1():
    print("aaahhh")
    print('aaahhh2')
print("this code is outside the function")
```

this code is outside the function

In [4]:
```python
function1
```

Out[4]: <function __main__.function1()>

In [5]:
```python
function1()
```

aaahhh
aaahhh2

In [6]:
```python
function1()
```

aaahhh
aaahhh2

In [7]:
```python
def function2(x):
    return 2*x
```

In [8]:
```python
a = function2(3)
```

In [9]:
```python
print("a")
```

a

In [10]:
```python
print(a)
```

6

In [11]:
```python
def y(x):
    return 3*x
```

In [12]:
```python
print (y(3))
```

9

In [22]:
```python
def function3(x, y):
    return x + y
```

In [25]:
```python
c=function3(3,2)
```

In [26]:
```python
print(c)
```

5

In [27]:
```python
def y(x):
    print(x)
    print("still in this function")
    return 3*x
```

In [28]:
```python
y(4)
```

```
4
still in this function
```

Out[28]: 12

In [29]:
```python
f = y(4)
```

```
4
still in this function
```

In [30]:
```python
f
```

Out[30]: 12

In [31]:
```python
print(f)
```

```
12
```

In [32]:
```python
y(4)
```

```
4
still in this function
```

Out[32]: 12

In [35]:
```python
print(y(4))
```

```
4
still in this function
12
```

In [36]:
```python
print (y(4))
```

```
4
still in this function
12
```

In [4]:
```python
x = "KM"
y= 2
z = 90

name2 = "SM"
height2 = 1.7
weight2 = 70

name3 = "JM"
height3 = 2
weight3 = 100
```

In [5]:
```python
def bmi_calc(name, height, weight):
    bmi = weight / (height**2)
    print("bmi: ")
    print(bmi)
    if bmi < 25:
        return name + " is not overweight"
    else:
        return name + " is overweight"
```

In [6]:
```python
result1 = bmi_calc(x, y, z)
result2 = bmi_calc(name2, height2, weight2)
result3 = bmi_calc(name3, height3, weight3)
```

```
bmi:
22.5
bmi:
24.221453287197235
bmi:
25.0
```

In [7]:
```python
print(result1)
```

```
KM is not overweight
```

In [59]:
```python
def convert(miles):
        return miles*1.6
```

In [60]:
```python
km = convert(2)
```

In [61]:
```python
print(km)
```

```
3.2
```

In [ ]: