04. FUNCTIONS

T2M2U2

WHAT IS A FUNCTION?

FUNCTION

- A function is a block of code which only runs when it is called.
- You can pass data, known as parameters, into a function.
- A function can return data as a result.

CREATING A FUNCTION

- In Python a function is defined using the **def** keyword:

```
1  def my_function():
2    print("Hello from a function")
```

WHAT IS THE RESULT OF THE CODE?

NOTHING

FIRST WE HAVE TO CALL THE FUNCTION

CALLING A FUNCTION

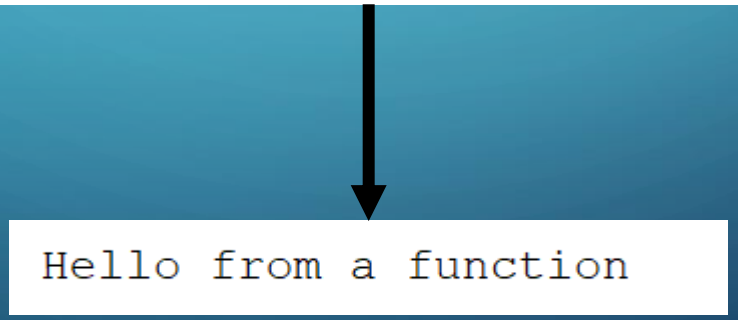
- To call a function, use the function name followed by parenthesis:

```
1  def my_function():
2    print("Hello from a function")
3
4  my_function()
```

CALLING A FUNCTION

- To call a function, use the function name followed by parenthesis:

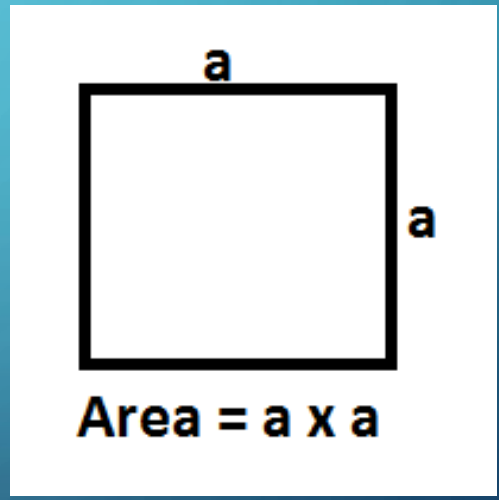
```
1  def my_function():
2    print("Hello from a function")
3
4  my_function()
```



```
Hello from a function
```

TASK 1:AREA OF A SQUARE

- Create a function “areaSquare” that will calculate the area of a square
- Ask the user in the function for size of the side
- Save it in variable a
- Calculate the area and show the result

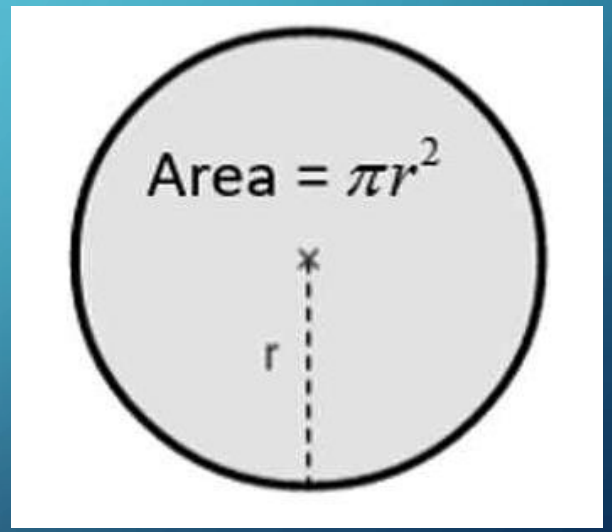


TASK 1:AREA OF A SQUARE

```
1  def areaSquare():
2    a = float(input("What is the size of the side?"))
3    area = a * a
4    print("Area is",area)
5
6  areaSquare()
```

TASK 2:AREA OF A CIRCLE

- Create a function “areaCircle” that will calculate the area of a circle
- Ask the user in the function for the radius of the circle
- Save it in variable r
- Calculate the area and show the result



TASK 2:AREA OF A CIRCLE

```
1  def areaCircle():
2    r = float(input("What is the radius of the circle?"))
3    area = 3.14 * r * r
4    print("Area is",area)
5
6  areaCircle()
```

ARGUMENTS

WHAT IS AN ARGUMENT?

ARGUMENTS

- Information can be passed into functions as arguments
- Arguments are specified after the function name, inside the parentheses
- We can add as many arguments as we want, just separate them with a comma

```
def function_name (arg1,arg2,arg3...):
```

ARGUMENTS

```
1  def my_function(name):
2      print("Your name is",name)
3
4  my_function("Peter")
```

ARGUMENTS

- By default, a function must be called with the correct number of arguments
- If your function expects 2 arguments, you have to call the function with 2 arguments, not more, and not less

```
1  def my_function(fname, lname):
2    print(fname + " " + lname)
3
4  my_function("Peter", "Parker")
```

ARGUMENTS

- If we try to call the function with 1 or 3 arguments, we will get an error:

```
1  def my_function(fname, lname):
2    print(fname + " " + lname)
3
4  my_function("Peter")
```

```
Traceback (most recent call last):
  File "E:\Downloads\Reactengle area.py", line 4, in <module>
    my_function("Peter")
TypeError: my_function() missing 1 required positional argument: 'lname'
```

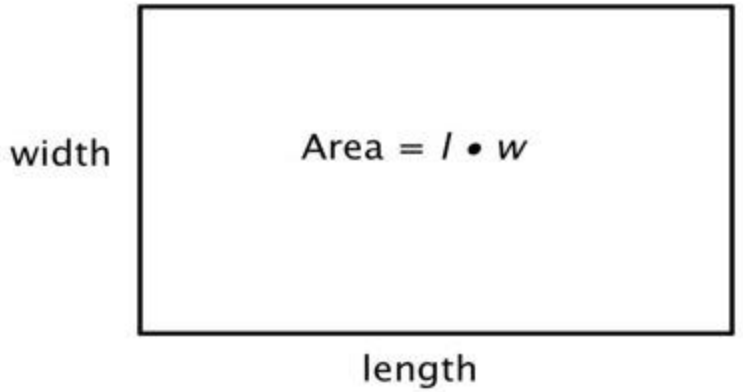
ARGUMENTS

- If we try to call the function with 1 or 3 arguments, we will get an error:

```
1  def my_function(fname):
2    print(fname + " " + lname)
3
4  my_function("Peter", "Parker")
```

```
Traceback (most recent call last):
  File "E:\Downloads\Reactengle area.py", line 4, in <module>
    my_function("Peter", "Parker")
TypeError: my_function() takes 1 positional argument but 2 were given
```

TASK 3: AREA OF A RECTANGLE

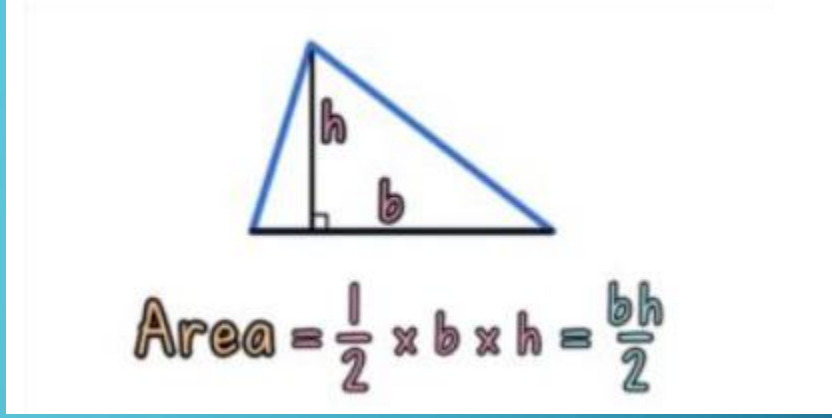


- Create a function “areaRect” that will calculate the area of a rectangle
- Ask the user **outside** of the function for W and L, save them
- Pass the values of W and L into the function by using arguments
- In the function check if W and L are more than 0, if they are calculate
- If they are not, show an error message

TASK 3: AREA OF A RECTANGLE

```
1  def areaRect(W,L):
2      if W > 0 and L > 0:
3          area = W * L
4          print("Area is",area)
5      else:
6          print("W or L is less than 0")
7
8  W = float(input("What is the W of the rect?"))
9  L = float(input("What is the L of the rect?"))
10 areaRect(W,L)
```

TASK 4: AREA OF A TRIANGLE



- Create a function “areaTri” that will calculate the area of a triangle
- Ask the user **outside** of the function for H and B, save them
- Pass the values of H and B into the function by using arguments
- In the function check if H and B are more than 0, if they are calculate
- If they are not, show an error message

TASK 4: AREA OF A TRIANGLE

```
1  def areaTri(H,B):
2      if H > 0 and B > 0:
3          area = 0.5 * H * B
4          print("Area is",area)
5      else:
6          print("H or B is less than 0")
7
8  H = float(input("What is the H of the tri?"))
9  B = float(input("What is the B of the tri?"))
10 areaTri(H,B)
```

DEFAULT ARGUMENT VALUE

- If we call the function without argument, it uses the default value:

```
1  def my_function(country = "Norway"):
2    print("I am from " + country)
3
4  my_function("Sweden")
5  my_function("India")
6  my_function()
7  my_function("Brazil")
```

DEFAULT ARGUMENT VALUE

- If we call the function without argument, it uses the default value:

```
1  def my_function(country = "Norway"):
2    print("I am from " + country)
3
4  my_function("Sweden")
5  my_function("India")
6  my_function()
7  my_function("Brazil")
```

```
I am from Sweden
I am from India
I am from Norway
I am from Brazil
```

RETURN VALUES

- To let a function **return** a value, use the return statement:

```
1  def my_function(x):
2    return 5 * x
3
4  print(my_function(3))
5  print(my_function(5))
6  print(my_function(9))
```

RETURN VALUES

- To let a function **return** a value, use the return statement:

```
1  def my_function(x):
2    return 5 * x
3
4  print(my_function(3))
5  print(my_function(5))
6  print(my_function(9))
```

```
15
25
45
```

ANY QUESTIONS?

THE END