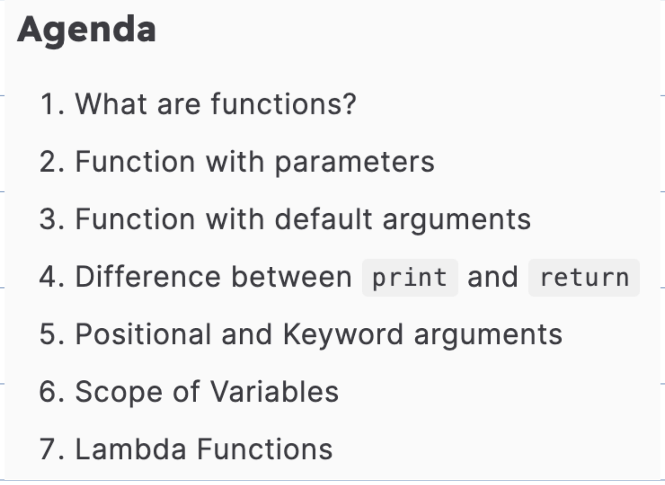
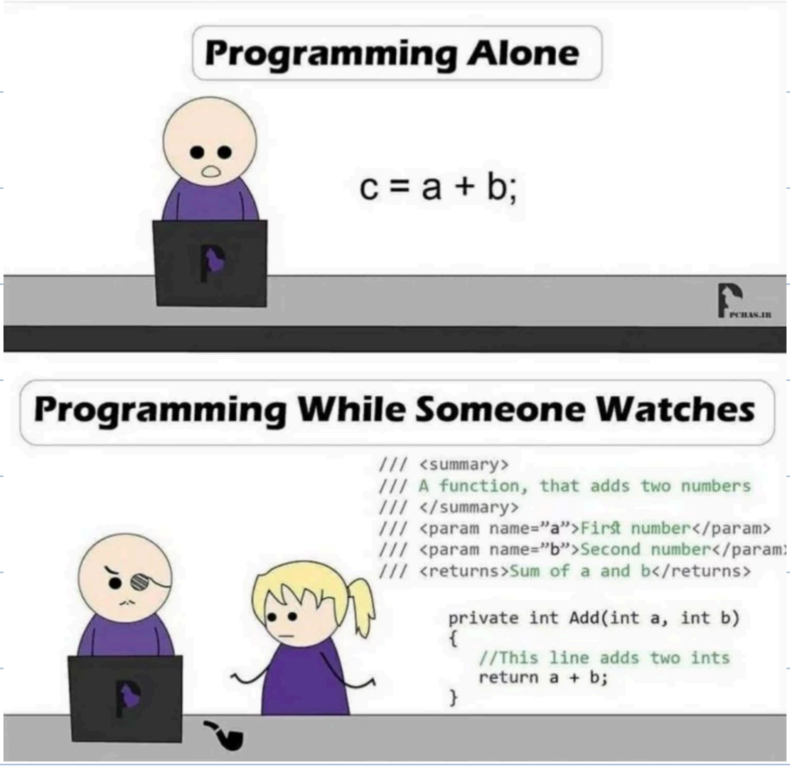


Agenda

1. What are functions?
2. Function with parameters
3. Function with default arguments
4. Difference between `print` and `return`
5. Positional and Keyword arguments
6. Scope of Variables
7. Lambda Functions



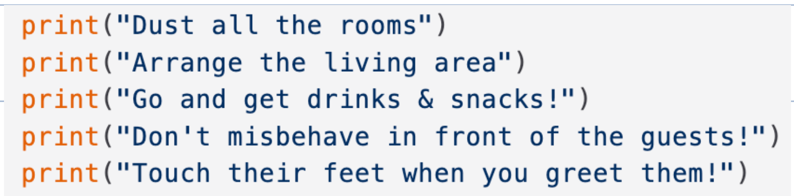
Programming Alone

```
c = a + b;
```

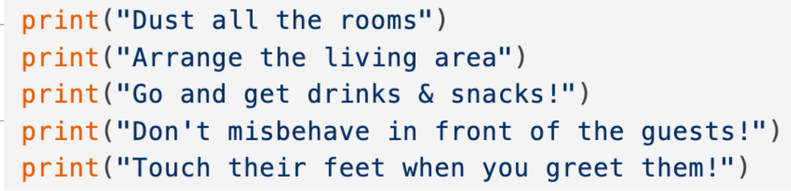
Programming While Someone Watches

```
/// <summary>
/// A function, that adds two numbers
/// </summary>
/// <param name="a">First number</param>
/// <param name="b">Second number</param
/// <returns>Sum of a and b</returns>

    private int Add(int a, int b)
    {
        //This line adds two ints
        return a + b;
    }
```

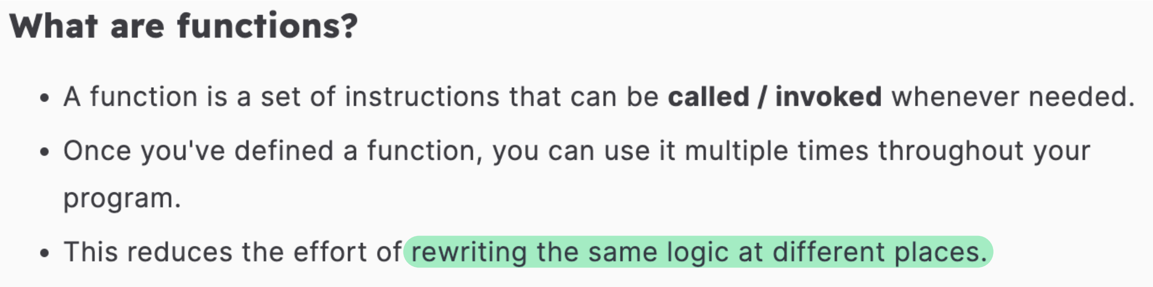
Set of tasks to complete wherever guest comes →



```python
print("Dust all the rooms")
print("Arrange the living area")
print("Go and get drinks & snacks!")
print("Don't misbehave in front of the guests!")
print("Touch their feet when you greet them!")
```

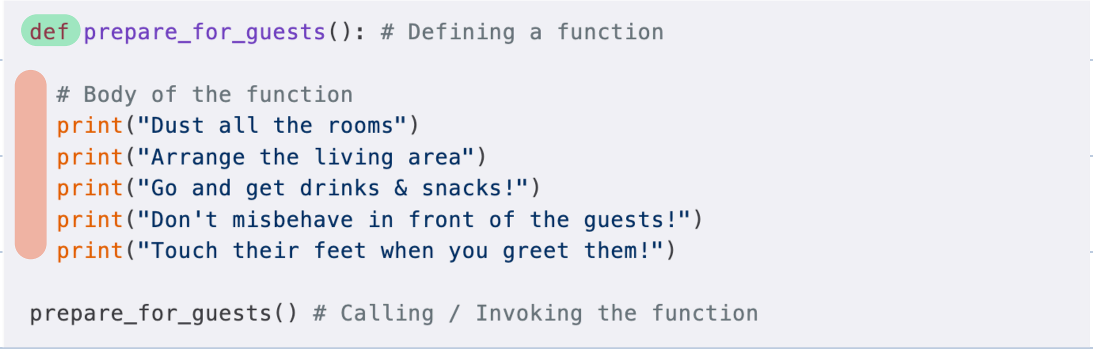


```python
print("Dust all the rooms")
print("Arrange the living area")
print("Go and get drinks & snacks!")
print("Don't misbehave in front of the guests!")
print("Touch their feet when you greet them!")
```



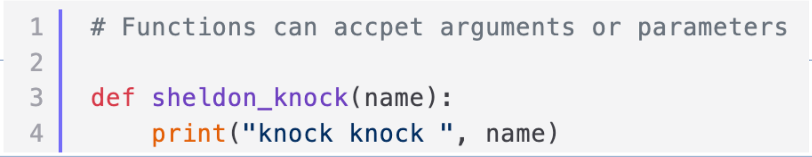
What are functions?

- A function is a set of instructions that can be **called / invoked** whenever needed.
- Once you've defined a function, you can use it multiple times throughout your program.
- This reduces the effort of rewriting the same logic at different places.



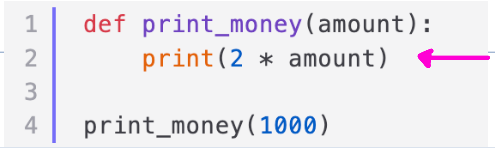
```python
def prepare_for_guests(): # Defining a function

    # Body of the function
    print("Dust all the rooms")
    print("Arrange the living area")
    print("Go and get drinks & snacks!")
    print("Don't misbehave in front of the guests!")
    print("Touch their feet when you greet them!")

prepare_for_guests() # Calling / Invoking the function
```

Functions with parameters

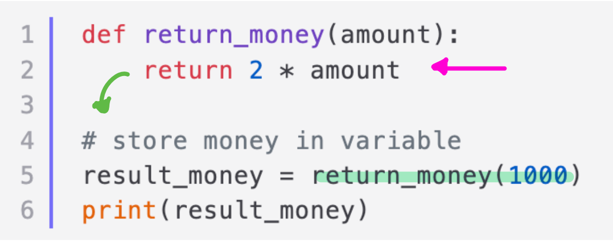


```python
# Functions can accpet arguments or parameters

def sheldon_knock(name):
    print("knock knock ", name)
```

Return Keyword → Return result from function call.



```python
def print_money(amount):
    print(2 * amount)

print_money(1000)
```



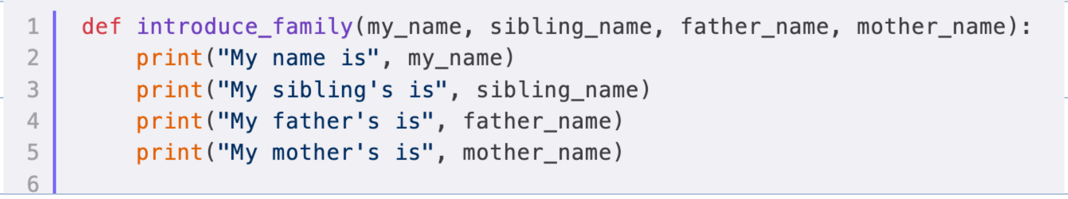
```python
def return_money(amount):
    return 2 * amount

# store money in variable
result_money = return_money(1000)
print(result_money)
```

print ("xyz")

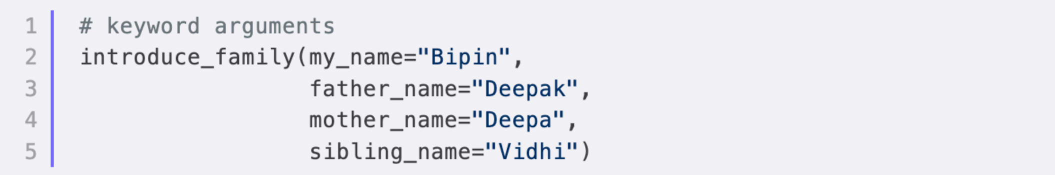
Help Function → Used to read documentation of the function.

Eg → help (print)

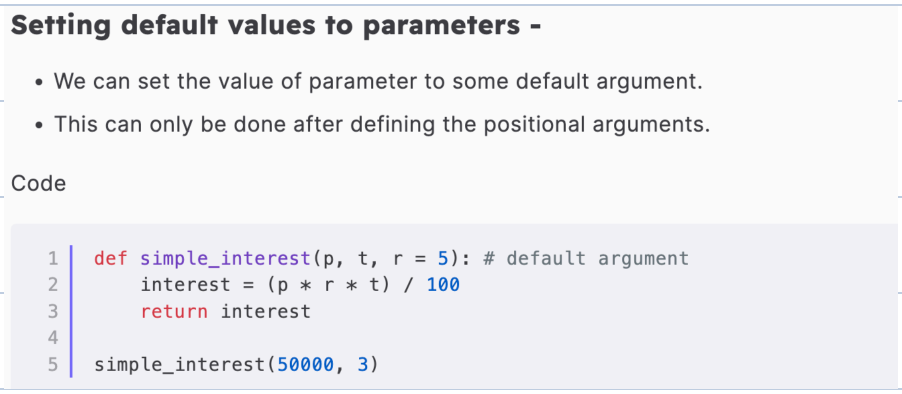
Positional & Keyword Arguments



```python
def introduce_family(my_name, sibling_name, father_name, mother_name):
    print("My name is", my_name)
    print("My sibling's is", sibling_name)
    print("My father's is", father_name)
    print("My mother's is", mother_name)

```



```python
# keyword arguments
introduce_family(my_name="Bipin",
                 father_name="Deepak",
                 mother_name="Deepa",
                 sibling_name="Vidhi")
```



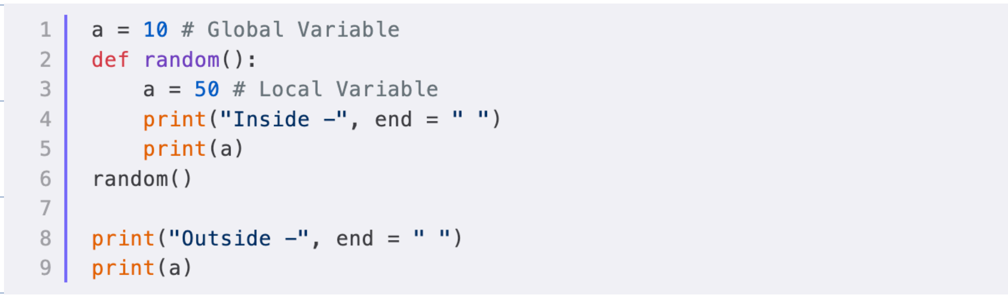
Setting default values to parameters -

- We can set the value of parameter to some default argument.
- This can only be done after defining the positional arguments.

Code

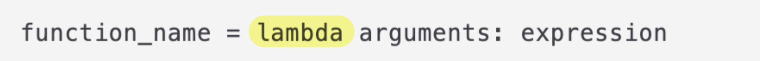
```python
def simple_interest(p, t, r = 5): # default argument
    interest = (p * r * t) / 100
    return interest

simple_interest(50000, 3)
```

Scope of a variable



```python
a = 10 # Global Variable
def random():
    a = 50 # Local Variable
    print("Inside -", end = " ")
    print(a)
random()

print("Outside -", end = " ")
print(a)
```

Lambda Function



```
function_name = lambda arguments: expression
```