ISIM 2021

International Symposium on
IoT and ML for
Ecosystem Restoration &
Multihazard Resilience
05th to 09th of June 2021

Introduction to Python Programming

Lakshmi S. Gopal

Amrita Center for Wireless Networks & Applications

Amrita School of Engineering

Amrita Vishwa Vidyapeetham

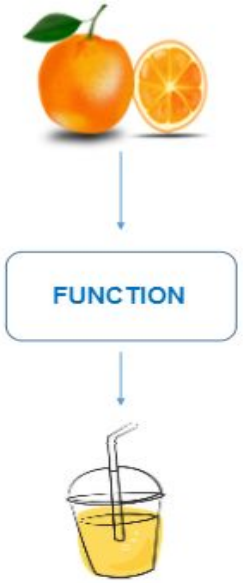
AMRITA VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY

Contents

- Python Functions
- Defining a Function
- Calling a Function
- Function Arguments
- The return statement
- Scope of Variables

Python Functions

- A function is a block of organized, reusable code
- Python gives you many built-in functions like print(), etc. but you can also create your own functions
- These functions are called user-defined functions
- A function consists of a sequence which takes one or more input(s), processes the input and produces the output



Defining and Calling a Function

```
def function_defining():
  print("function definition begins with def")
  print("give necessary function name followed by parantheses and colon")
  print("don't forget the indentation!")
  print("please call the function after defining")

function_defining()  #function call
```

Function Arguments

- The advantage of functions is that it can take multiple inputs
- Arguments are the values that are passed while calling a function
- When arguments are passed, a parameter catches it in the function definition

```
def function_defining(str1): #parameter str1 ; can use any variable name
  print(str1)
str1="python programming"
function_defining(str1)  #passing str1 value as an argument in function call
```

```
python programming
```

Function Arguments

- Different types of function arguments:
  - Required Arguments
    - Required arguments are the arguments passed to a function in correct positional order
    - number of arguments in the function call should match exactly with the function definition
  - Keyword Arguments
    - Keyword arguments are related to the function calls
    - When you use keyword arguments in a function call, the caller identifies the arguments by the parameter name

Function Arguments

- Different types of function arguments:
  - Default arguments
    - A default argument is an argument that assumes a default value if a value is not provided in the function call for that argument
  - Variable-length arguments
    - You may need to process a function for more arguments than you specified while defining the function
    - An asterisk (*) is placed before the variable name
    - remains empty if no additional arguments are specified during the function call

The Return Statement

- The statement return [expression] exits a function, optionally passing back an expression to the caller
- A return statement with no arguments is the same as return None

```
def sum(a,b):
  return a+b #returns a value

addition=sum(5,5) #addition variable catches the return value
print(addition)
#OR
print(sum(5,5)) #can directly print also
```

```
10
10
```

Scope of Variables

- All variables in a program may not be accessible at all locations in that program
- This depends on where you have declared a variable.
- The scope of a variable determines the portion of the program where you can access a particular identifier
- There are two basic scopes of variables in Python :–
  - Global variables
  - Local variables

Scope of Variables

- Variables that are defined inside a function body have a local scope, and those defined outside have a global scope.
- local variables can be accessed only inside the function in which they are declared
- global variables can be accessed throughout the program body by all functions

Thank you !