AMRITA VISHWA VIDYAPEETHAM

Python Input, Output and Import

Anoop S Babu
Faculty Associate
Dept. of Computer Science & Engineering
bsanoop@am.amrita.edu

Python Input

- **input()** function is use take the input from the user.

Syntax

```
input([prompt])
```

  – where **prompt** is the string we wish to display on the screen. It is optional

Example

```
>>> num = input('Enter a number: ')
Enter a number: 10
>>> num
'10'
```

Python Output Using print() function

- **print()** function to output data to the standard output device (screen).

Example 1:

```
print('This sentence is output to the screen')
```

Output

```
This sentence is output to the screen
```

Example 2:

```
a = 5
print('The value of a is', a)
```

Output

```
The value of a is 5
```

In Example 2, **space** was added between the string and the value of variable a. This is by default.

Syntax of the print() function

- The actual syntax of print() function is,

```
print(*objects, sep=' ', end='\n', file=sys.stdout, flush=False)
```

  - **objects** is the value(s) to be printed.
  - **sep** separator is used between the values. It defaults into a **space character**.
  - After all values are printed, **end** is printed. It defaults into a **new line**.
  - **file** is the object where the values are printed and its default value is sys.stdout (screen).
  - **flush** is a Boolean, specifying if the output is flushed (True) or buffered (False). Default is **False**

Syntax of the print() function

Example

```
print(1, 2, 3, 4)
print(1, 2, 3, 4, sep='*')
print(1, 2, 3, 4, sep='#', end='&')
```

Output

```
1 2 3 4
1*2*3*4
1#2#3#4&
```

Python Import

- When our program grows bigger, it is a good idea to break it into different modules.
- A module is a file containing Python definitions and statements. Python modules have a **filename** and end with the extension **.py**.
- Definitions inside a module can be **imported to another module** or the interactive interpreter in Python. We use the **import** keyword to do this.

Example

```
import math
print(math.pi)
```

Python Import

Example 1

```
import math
print(math.pi)
```

Output

```
3.141592653589793
```

- In Example 1, **all the definitions** inside math module are available in our scope.
- We can also import some **specific attributes** and **functions only,** using the **from** keyword

Example 2

```
>>> from math import pi
>>> pi
3.141592653589793
```