Features of Functions

Spoken Tutorial Project
http://spoken-tutorial.org
National Mission on Education through ICT

http://sakshat.ac.in

Script: Aditya Palaparthy

Narration: Kiran K

IIT Bombay

10 December 2015

In this tutorial we will learn,

Objectives

In this tutorial we will learn,

- ▶ **Assign default values to arguments, when defining functions.**

In this tutorial we will learn,

- ▶ **Assign default values to arguments, when defining functions.**

- ▶ **Define and call functions with keyword arguments.**

Objectives

- **Learn some of the built-in functions available in Python standard library and the scientific computing libraries.**

System Specifications

System Specifications

- **Ubuntu Linux 14.04**

System Specifications

- **Ubuntu Linux 14.04**
- `Python 2.7.6`

System Specifications

- **Ubuntu Linux 14.04**
- `Python 2.7.6`
- `IPython 4.0.0`

Pre-requisite

To practise this tutorial, you should know how to

- ▶ **run basic Python commands on the ipython console.**
- ▶ **use functions**

If not, see the pre-requisite Python tutorials on http://spoken-tutorial.org

Functions

- s.strip() strips on spaces.
- s.strip('@') strips the string of '@' symbols.
- plot(x, y) plots x v/s y using default line style.
- plot(x, y, 'o') plots x v/s y with circle markers.

Functions

- **linspace(0,2*pi,100) returns 100 pts between 0 and 2pi**
- **linspace(0,2*pi) returns 50 pts between 0 and 2pi**

► Redefine the function `welcome`, by interchanging it's arguments. Place the `name` argument with it's default value of "World" before the `greet` argument.

Solution 1

- ```
  def welcome(name="World", greet):
      print greet, name
  ```

Solution 1

- ```
  def welcome(name="World", greet):
      print greet, name
  ```

- **We get an error that reads,**
  ```
  SyntaxError: non-default argument
  follows default argument.
  ```

Solution 1

- ```
  def welcome(name="World", greet):
      print greet, name
  ```

- **We get an error that reads,**
  ```
  SyntaxError: non-default argument
  follows default argument.
  ```

- **When defining a function all the argument with default values should come at the end.**

▶ Redefine the function `welcome` with a default value of "Hello" to the `greet` argument.
Then, call the function without any arguments.

Keyword arguments examples

```
legend(['sin(2y)'], loc = 'center')

plot(y, sin(y), 'g', linewidth = 2)

annotate('local max', xy = (1.5, 1))

pie(science.values(),
            labels = science.keys())
```

Built-in functions

Math functions – abs, sin, ...

Plot functions – plot, bar, pie ...

Boolean functions – and, or, not ...

Classes of functions

- **pylab**
  - **plot, bar, contour, boxplot, errorbar, log, polar, quiver, semilog**

- **scipy (modules)**
  - **fftpack, stats, linalg, ndimage, signal, optimize, integrate**

Summary

In this tutorial, we have learnt to,

- ▶ **Define functions with default arguments.**

Summary

In this tutorial, we have learnt to,

- ▶ **Define functions with default arguments.**
- ▶ **Call functions using keyword arguments.**

Summary

In this tutorial, we have learnt to,

- Define functions with default arguments.
- Call functions using keyword arguments.
- Use the range of functions available in the Python standard library and the Scientific Computing related packages.

1. **All arguments of a function cannot have default values. True or False?**

Evaluation

1. **All arguments of a function cannot have default values. True or False?**
2. **The following is a valid function definition. True or False?**

1. **All arguments of a function cannot have default values. True or False?**
2. **The following is a valid function definition. True or False?**

```python
def separator(count=40, char, show=False):
    if show:
        print char * count
    return char * count
```

3. **When calling a function,**
   - ▸ **the arguments should always be in the order in which they are defined.**

3. **When calling a function,**
   - ▸ **the arguments should always be in the order in which they are defined.**
   - ▸ **only keyword arguments can be in any order, but should be called at the beginning.**

Evaluation

3. **When calling a function,**
   - ► **the arguments should always be in the order in which they are defined.**
   - ► **only keyword arguments can be in any order, but should be called at the beginning.**
   - ► **only keyword arguments can be in any order, but should be called at the end.**

Solutions

1. **False**

1. **False**

2. **False**

Solutions

1. **False**

2. **False**

3. **Only keyword arguments can be in any order, but should be called at the end.**

Forum to answer questions

- **Do you have questions in THIS Spoken Tutorial?**
- **Choose the minute and second where you have the question.**
- **Explain your question briefly.**
- **Someone from the FOSSEE team will answer them. Please visit**

http://forums.spoken-tutorial.org/

Forum to answer questions

- **Questions not related to the Spoken Tutorial?**
- **Do you have general / technical questions on the Software?**
- **Please visit the FOSSEE Forum**
    http://forums.fossee.in/
- **Choose the Software and post your question.**

Textbook Companion Project

- **The FOSSEE team coordinates coding of solved examples of popular books**
- **We give honorarium and certificate to those who do this**

For more details, please visit this site:

http://tbc-python.fossee.in/

Acknowledgements

- **Spoken Tutorial Project is a part of the Talk to a Teacher project**
- **It is supported by the National Mission on Education through ICT, MHRD, Government of India**
- **More information on this Mission is available at:**
  http://spoken-tutorial. org/NMEICT-Intro

THANK YOU!

For more Information, visit our website http://fossee.in/