Functions and Documentation Strings

School of Information Studies
Syracuse University

Predefined Functions

Functions we have used

- Type—to determine the type of variable
- Len—to determine number of elements

Dot notation functions

- Random numbers
  - >>>import random
  - >>>random.randint (1,10)

Functions used directly

- Var1.upper
- Var1.lower

Defining Our Own Functions

Function with no parameters

```
>>>def hello ():
…       print('Hello!')
…
>>>hello()                #calling the function
Hello!
>>>type(hello)
<class 'function'>
```

Defining Our Own Functions

Function with two parameters

```
>>>def area (width, height):
…       a = width * height
…       return a
…
>>>area (5, 7)
35
```

Scope of Variables

Defined within function

- Local scope
- Not available outside function

Defined outside of function

- Global scope
- Can be used outside of function

Keyword Arguments

Positional arguments—like our area function

- Def area (width, height):
- A = width * height
- Return a

Keyword arguments

- >>>def printtentuples (xtuple, beginindex=0, endindex=9):
- … print (xtuple[beginindex:endindex+1])

Using the Arguments (or Not)

Missing an argument

- >>>tentuple = (3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
- >>>len (tentuple)
- 10
- >>>printtentuples (tentuple)
- (3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
- >>>printtentuples (tentuple, endindex – 4)
- (3, 4, 5, 6, 7)

All arguments specified

- >>>printtentuples (tentuple, beginindex=3, endindex=4)
- (6, 7)

Documentation Strings

Use of docstrings to document functions

- >>>def area (width, height):
- … ”’compute area by taking two int or float args and returns int or float result”’
- … return width*height
- …
- >>>area (6, 8)
- 48