

Python Basics!

arguments, parameters, methods, comments

CS101 Lecture #5

Administrivia

Administrivia

- ▣ Homework #2 is due Friday Sep. 9.
- ▣ Labs resume next week.

Warmup Quiz

Question #1

```
s = '%s' + 'i'  
i = 3 / 6  
x = float(s%i) * 2
```

What is the value of x?

- A 0.0
- B '%i%i'
- C 1.0
- D '1.0'

Question #2

```
s = "WATER MAIN"[2:6]
t = int(3.7)
x = s[-1] + s[t-2]
```

What is the value of x?

- A "NA"
- B " E"
- C " R"
- D "ME"

Question #3 (Worked)

```
s = "WATER MAIN"[2:6]
    #0123456789
s = "TER "
t = int(3.7)
t = 3
x = s[-1] + s[t-2]
x = " " + "E"
x = " E"
```

Question #4

```
i = len("WATER MAIN")  
c = (1.0 + 2.0j) * (-i)  
x = abs( min( c.real, -13 ) )
```

What is the value of x?

- A 0
- B 11
- C 12
- D 13

Functions Redux

Functions

- ⌘ A small program (block of code) we can run within Python.
 - ⌘ Saves us from rewriting code
 - ⌘ Don't reinvent the wheel!
- ⌘ Analogy: Functions are more verbs.
- ⌘ Also called subroutine or procedure.

Function calls

- ❖ When we want to execute a function, we call or invoke it.
- ❖ Use name of the function with parentheses.
 - ❖ `print()`
- ❖ Many functions come built-in to Python or in the standard library.
- ❖ Others we will compose at need.

User input

- `input` is a built-in function.
- Argument: string prompting user
- Return value: input from user (as `str`)

Goal

- ✦ A program should achieve a goal.

Goal

- A program should achieve a goal.
- Let's implement the quadratic equation.

Example: Quadratic equation

```
print( "QUADRATIC SOLVER" )  
print( "a x^2 + b x + c = 0" )
```

```
a = float( input( 'a: ' ) )  
b = float( input( 'b: ' ) )  
c = float( input( 'c: ' ) )
```

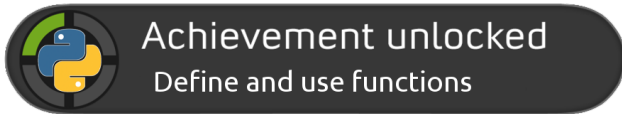
```
root = ( b**2 - 4*a*c ) ** 0.5  
denom = 2 * a
```

```
pos = (-b + root) / denom  
neg = (-b - root) / denom
```

```
message1 = "%.2f + %.2fi" % (pos.real,pos.imag)  
message2 = "%.2f + %.2fi" % (neg.real,neg.imag)
```

```
print("Solution 1: %s" % message1)  
print("Solution 2: %s" % message2)
```

Achievement unlocked!



Methods

Methods

- ▣ Like attributes, functions can be stored inside a type as well.

Methods

- Like attributes, functions can be stored inside a type as well.
- Use attribute operator on the value.

Methods

- Like attributes, functions can be stored inside a type as well.
- Use attribute operator on the value.
`"STOP SHOUTING!".lower()`
`(1 + 1j).conjugate()`

Methods

- Like attributes, functions can be stored inside a type as well.
- Use attribute operator on the value.
`"STOP SHOUTING!".lower()`
`(1 + 1j).conjugate()`
- Value is treated like an argument.

String methods

```
"GATTACA".count('A')  
"MVEMJSUN".find('J')  
"ABACADABRA".replace('AB','G')  
'FNORD'.strip()  
'high king of narnia'.title()  
'wEiRd'.swapcase()
```

Example

```
s = "WATER MAIN"  
x = s[ 0:s.find( ' ' ) ].lower()  
x = x.title().swapcase()
```

What is the value of x?

- A 'wATER'
- B 'Water'
- C 'wATE'
- D 'aTER'

Comments

Methods

- ✦ We can explain our code using comments.

Methods

- ❖ We can explain our code using comments.
- ❖ Comments begin with a `#` sign; Python ignore the rest of the line.

Methods

- ❖ We can explain our code using comments.
- ❖ Comments begin with a `#` sign; Python ignore the rest of the line.
- ❖ Long comments can also be stored as triple-quoted strings.

Methods

- ❖ We can explain our code using comments.
- ❖ Comments begin with a # sign; Python ignore the rest of the line.
- ❖ Long comments can also be stored as triple-quoted strings.

```
dx = 0.01 # grid spacing, m  
V = 14.2 # voltage, V  
,,,,,
```

```
This is an extended comment.  
I can be many lines long.
```

```
Use me to explain functions or formulae, to d  
or to temporarily hide blocks you don't want  
,,,,,
```

Reminders

Reminders

- ▣ Homework #2 is due Friday Sep. 9.
- ▣ Labs resume next week.