

CIS 122

Assignment and onward

Recap

- Storing values in variables
 - `x = 5`
 - `color = "purple"`
- Saving code
 - Separate file
 - **.py** extension
 - Make sure to run code (F5)

Code Files

- Python executes each line in order
 - Performs any assignments
 - Executes all commands
 - Doesn't print out anything unless you ask
- Break up your code with comments
 - **# Python ignores anything following a hash mark**
 - `cowName = "bessie" # Give name to cow`

Assignment Quiz

num1 = 3

string1 = "Hip "

num2 = num1 + num1

string2 = string1 * num1

num1 = num1 + 1

string2 = string1 + string2

Assignment Quiz

num1 = 3

num1 → 3

string1 = "Hip "

num2 = num1 + num1

string2 = string1 * num1

num1 = num1 + 1

string2 = string1 + string2

Assignment Quiz

num1 = 3

num1 → 3

string1 = "Hip "

string1 → "Hip "

num2 = num1 + num1

string2 = string1 * num1

num1 = num1 + 1

string2 = string1 + string2

Assignment Quiz

num1 = 3

num1 → 3

string1 = "Hip "

string1 → "Hip "

num2 = num1 + num1

num2 → 6

string2 = string1 * num1

num1 = num1 + 1

string2 = string1 + string2

Assignment Quiz

num1 = 3

num1 → 3

string1 = "Hip "

string1 → "Hip "

num2 = num1 + num1

num2 → 6

string2 = string1 * num1

string2 → "Hip Hip Hip "

num1 = num1 + 1

string2 = string1 + string2

Assignment Quiz

num1 = 3

num1 → 4

string1 = "Hip "

string1 → "Hip "

num2 = num1 + num1

num2 → 6

string2 = string1 * num1

string2 → "Hip Hip Hip "

num1 = num1 + 1

string2 = string1 + string2

Assignment Quiz

num1 = 3

num1 → 4

string1 = "Hip "

string1 → "Hip "

num2 = num1 + num1

num2 → 6

string2 = string1 * num1

string2 → "Hip Hip Hip Hip "

num1 = num1 + 1

string2 = string1 + string2

Printing Things

- If you want feedback from a code file, use print statements
 - `print "Hello World"`
 - `print 1,2,3`
- What can we print?
 - Any value (ints, floats, strings...)
 - Any variable (as long as it has been defined)
 - Any expression (that can be reduced to a value)
- `print` I hope this prints correctly
 - This will cause a syntax error
 - Why?
 - How could we fix it?

Homework Overview

- 4 parts
 - Part 0 - Getting Started with Python
 - Part 1 - Getting to Know You
 - Part 2 - What's in a Squiggle
 - Part 3 - Some Quick Candy Calculation
- Why start counting at 0?
 - Computer Science convention
- Everyone has done part 0
 - (I hope)

Homework Overview

- Part 1 - Getting to Know You
- Existing code prints out empty info sheet

```
>>> ===== RESTART =====
>>>
Welcome to Python
~~~~~
Name:
Year:
Major:

Why are you taking this class?

What do you hope to take away from this class?

Tell me something interesting about yourself.

~~~~~
>>>
```

Homework Overview

- Part 1 - Getting to Know You
- Change code so it prints out more useful info

```
>>> ===== RESTART =====
>>>
Welcome to Python
~~~~~
Name: Greg
Year: Grad Student
Major: Computer Science

Why are you taking this class?
I love teaching computer science!

What do you hope to take away from this class?
I want to learn to better convey computer science
topics and techniques to students new to programming

Tell me something interesting about yourself.
I used to beatbox in an a capella group
~~~~~
>>>
```

Homework Overview

- Part 2 - What's in a Squiggle
- Two short questions about code from part 1
- Answer in a comment in your code
 - Don't need to print out your answer

Question Prompt...

#

Your answer as a comment...

#

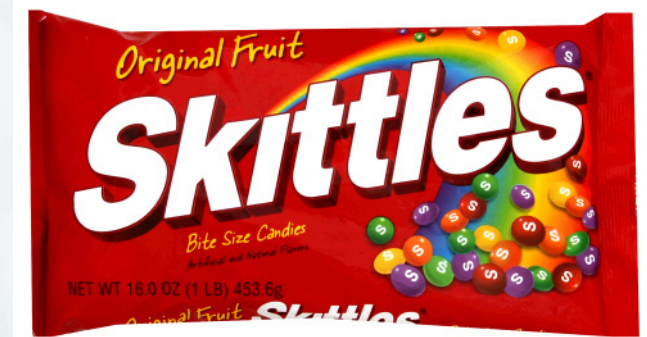
Homework Overview

- Part 3 - Some Quick Candy Calculation
- I have some number of skittles of different colors
 - 7 orange skittles
 - 3 times as many green skittles as orange skittles
 - ...
- Use Python to figure out how many skittles I have of each color
- Print out the results
 - "I have 7 orange skittles"...



Homework Overview

- Part 3 - Some Quick Candy Calculation
- Don't just do the calculations by hand!
 - Use variables
 - Store information
- Why does it matter?
 - "Oops, I only had 6 orange skittles..."



Now for something completely different

- We've seen how to manipulate data with operators
 - +, -, *, ...
 - But there are only so many operators
 - What if we want to do more?
- Functions!
 - Input / Output machines

```
>>> abs(-42)
```

```
42
```

Anatomy of a Function

Function Name

Argument / Parameter

`abs(-42)`

Functions

- Here are some other functions
 - `int(x)` - returns the integer version of x
 - `float(x)` - returns the float version of x
 - `str(x)` - returns the string version of x
 - `round(x)` - returns the whole float closest to x
 - `max(x,y)` - returns the larger of x and y
- Functions can take multiple arguments

Functions

- What can you put in a function?
 - values
 - expressions
 - variables
 - results from other functions!
- What does this return?
>>> `abs(round(-7.9))`
- Use a series of functions to convert `'-42'` to `'42'`

Python Color Codes

"strings"

keywords (e.g. print)

builtin functions (e.g. abs)