Python Programming

with a simple business-oriented application

Product Design and Ideation

Innovation vs Invention

What is the difference between innovation and invention?

1. What is invention?

Creating something for the first time

2. What is innovation?

Improving or contributing to an existing product

Combine Ideas: Combine form or function of things that already exist

Combining Ideas

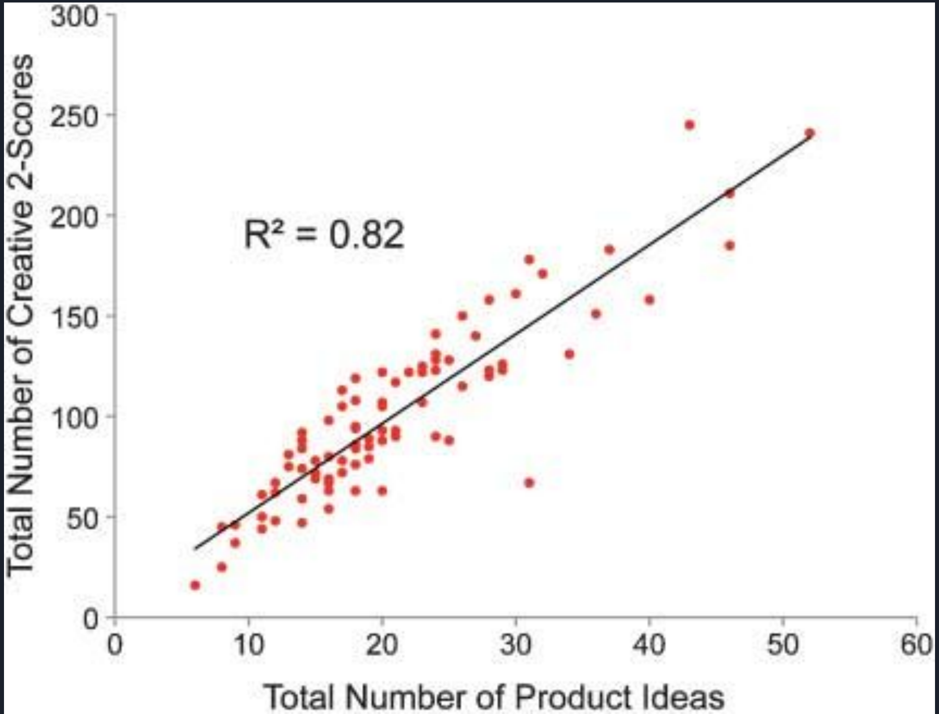
+

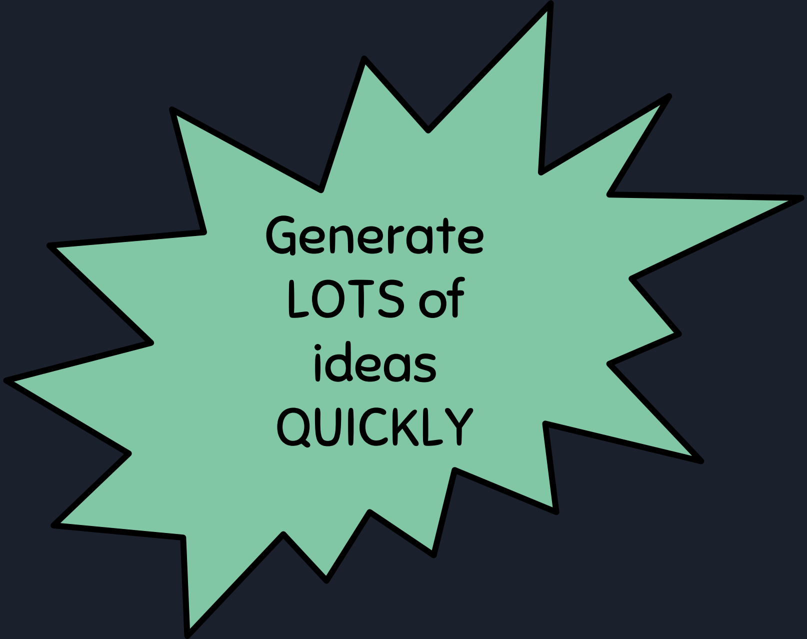
=

+

=

Quantity > Quality





> 1 idea per minute

In 10 minutes you should have at least 10 ideas

"When you come to a roadblock, take a detour."

- Mary Kay Ash

PYTHON 101

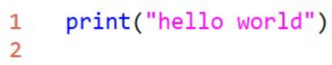
Leon Cheng

Getting Started

- We are using Python 3 (not 2)
- Text Editor
  - Use your editor of choice (don’t have to use IDLE)
  - Suggestions: Sublime, Atom, Emacs, Visual Studio, Vim, Notepad++, etc…
- Running Python
  - Using the terminal

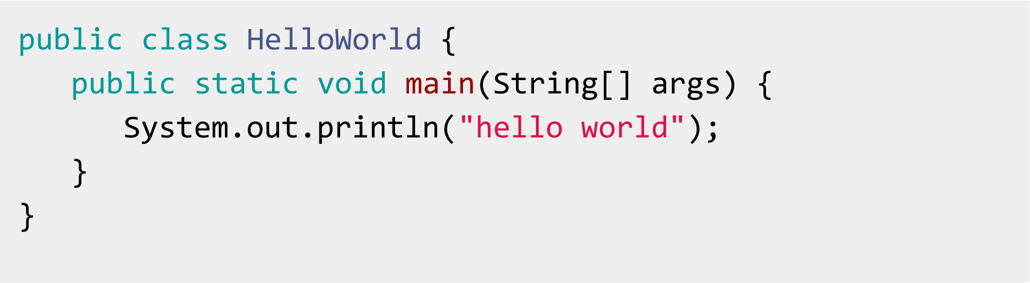
HELLO WORLD

Python:



```python
print("hello world")

```

Java:

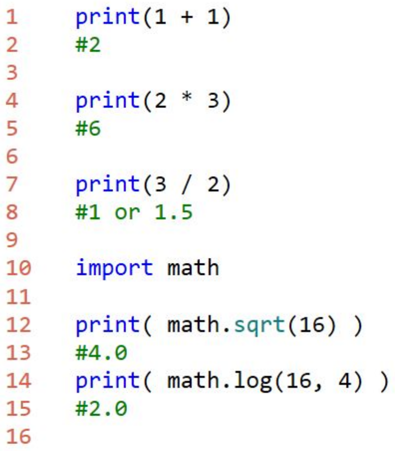


```java
public class HelloWorld {
   public static void main(String[] args) {
      System.out.println("hello world");
   }
}
```

- Very friendly syntax
- Running it
  - Create file: *filename.py*
  - Run: $ *python filename.py*

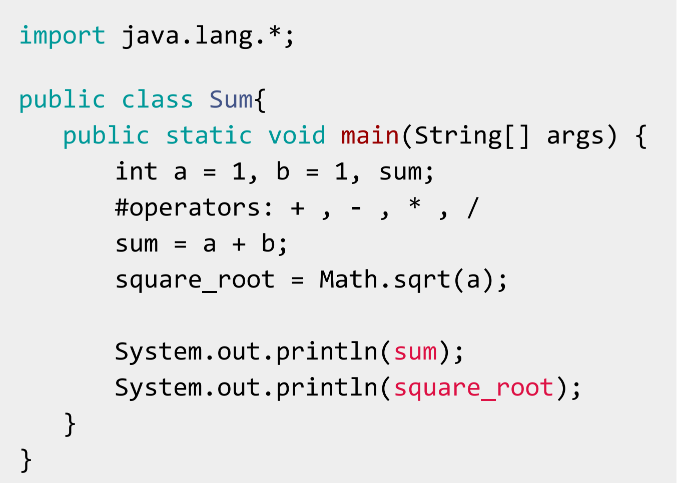
BASIC MATH

Python:



```python
print(1 + 1)
#2

print(2 * 3)
#6

print(3 / 2)
#1 or 1.5

import math

print( math.sqrt(16) )
#4.0
print( math.log(16, 4) )
#2.0

```

Java:

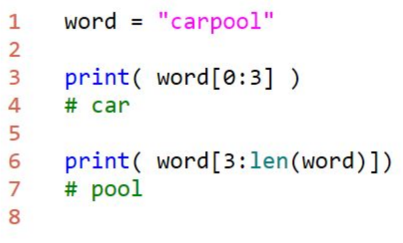


```java
import java.lang.*;

public class Sum{
   public static void main(String[] args) {
      int a = 1, b = 1, sum;
      #operators: + , - , * , /
      sum = a + b;
      square_root = Math.sqrt(a);

      System.out.println(sum);
      System.out.println(square_root);
   }
}
```

- Be careful with integer division
- Some operations require an import

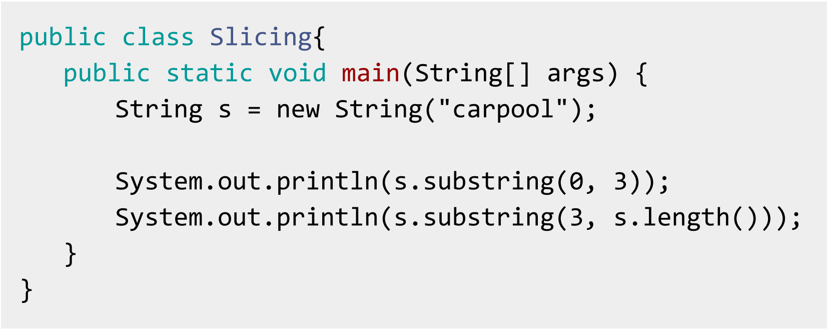
STRINGS

Python:



```python
word = "carpool"

print( word[0:3] )
# car

print( word[3:len(word)])
# pool

```

Java:

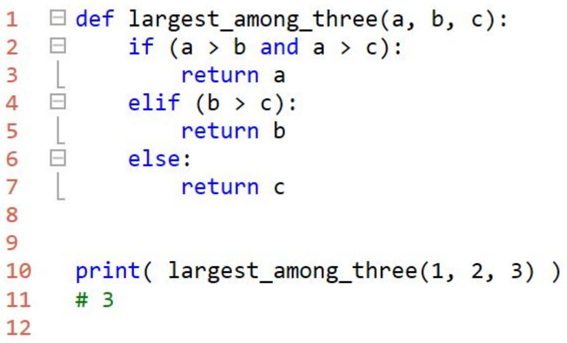


```java
public class Slicing{
   public static void main(String[] args) {
       String s = new String("carpool");

       System.out.println(s.substring(0, 3));
       System.out.println(s.substring(3, s.length()));
   }
}
```

- [inclusive:exclusive]

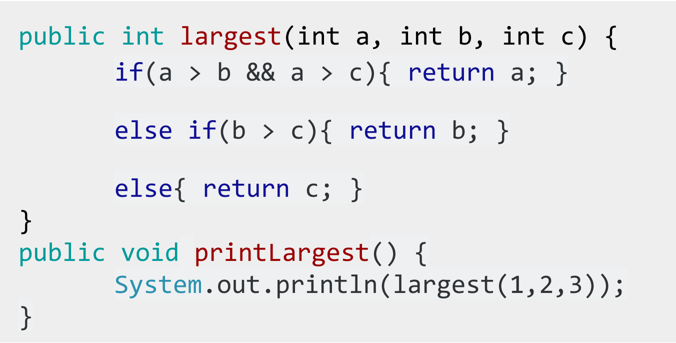
FUNCTIONS AND IF STATEMENTS

Python:



```python
def largest_among_three(a, b, c):
    if (a > b and a > c):
        return a
    elif (b > c):
        return b
    else:
        return c


print( largest_among_three(1, 2, 3) )
# 3

```

Java:

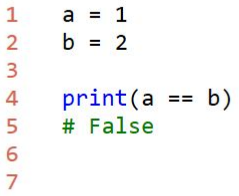


```java
public int largest(int a, int b, int c) {
       if(a > b && a > c){ return a; }

       else if(b > c){ return b; }

       else{ return c; }
}
public void printLargest() {
       System.out.println(largest(1,2,3));
}
```

- Code is processed in order

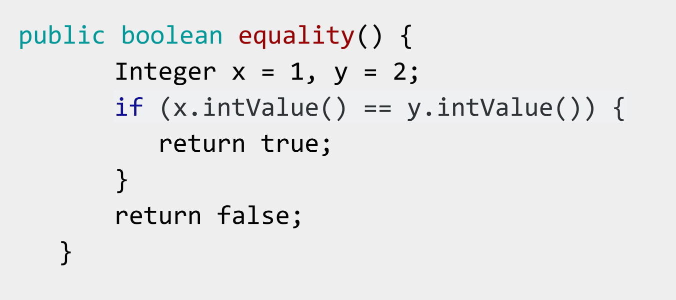
EQUALITY VS ASSIGNMENT

Python:



```python
a = 1
b = 2

print(a == b)
# False
```

Java:

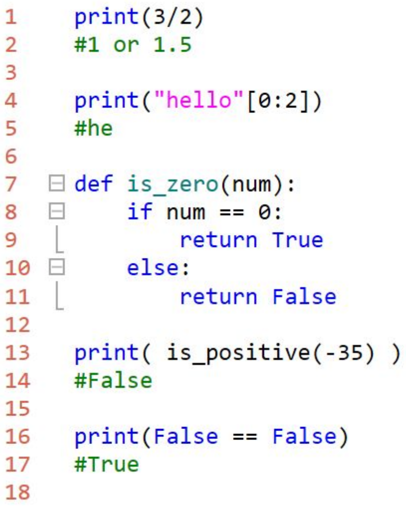


```java
public boolean equality() {
       Integer x = 1, y = 2;
       if (x.intValue() == y.intValue()) {
          return true;
       }
       return false;
   }
```

- Often used for If statements

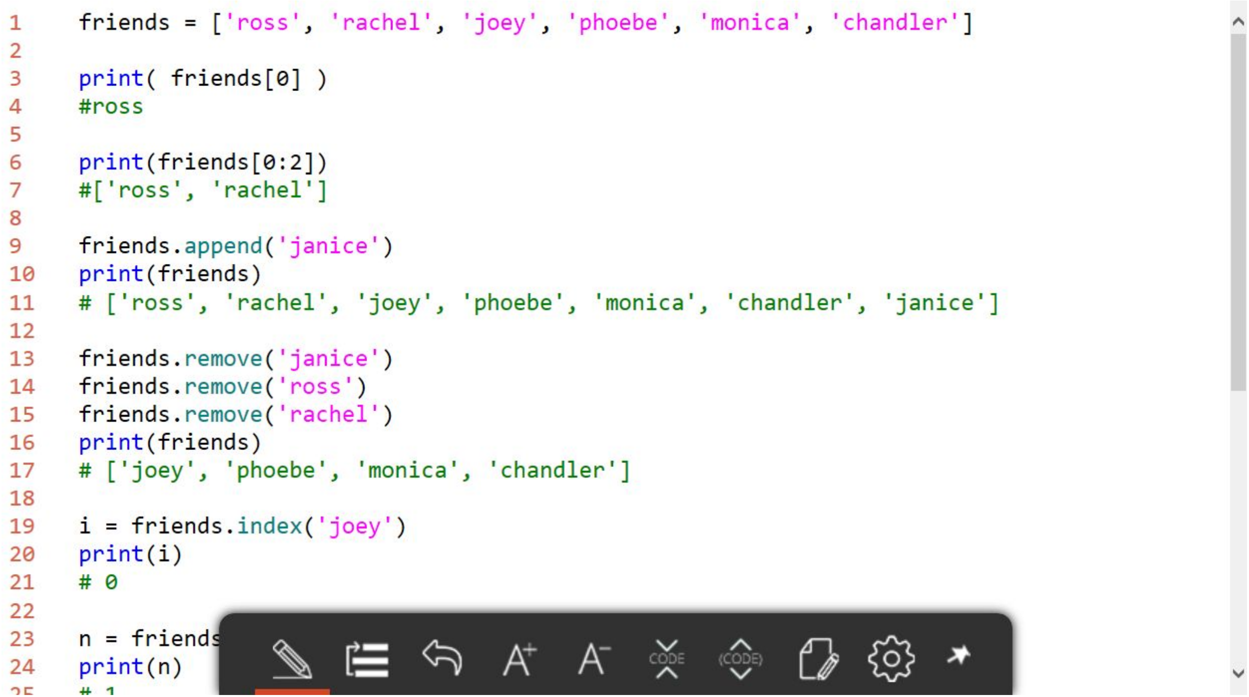
REVIEW

- Basic Math
- Strings
- Functions
- If Statements
- Equality
- Assignment



```python
print(3/2)
#1 or 1.5

print("hello"[0:2])
#he

def is_zero(num):
    if num == 0:
        return True
    else:
        return False

print( is_positive(-35) )
#False

print(False == False)
#True

```

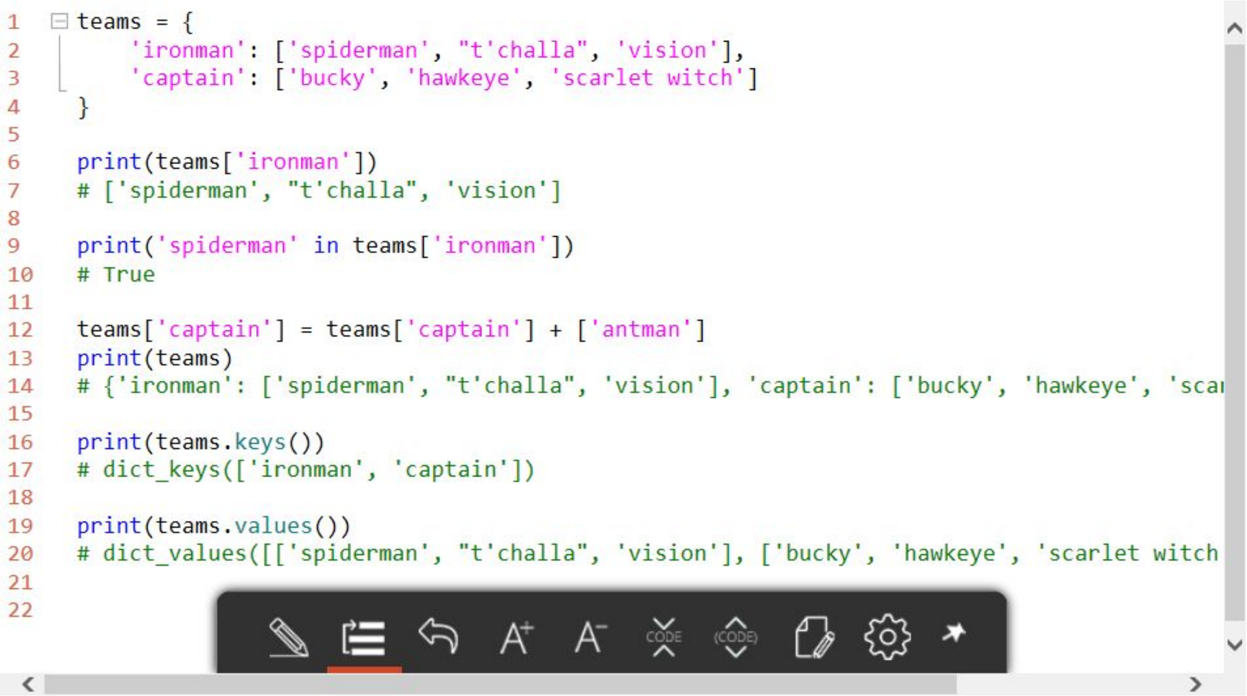
LISTS



```
friends = ['ross', 'rachel', 'joey', 'phoebe', 'monica', 'chandler']

print( friends[0] )
#ross

print(friends[0:2])
#['ross', 'rachel']

friends.append('janice')
print(friends)
# ['ross', 'rachel', 'joey', 'phoebe', 'monica', 'chandler', 'janice']

friends.remove('janice')
friends.remove('ross')
friends.remove('rachel')
print(friends)
# ['joey', 'phoebe', 'monica', 'chandler']

i = friends.index('joey')
print(i)
# 0

n = friends
print(n)
# 1
```

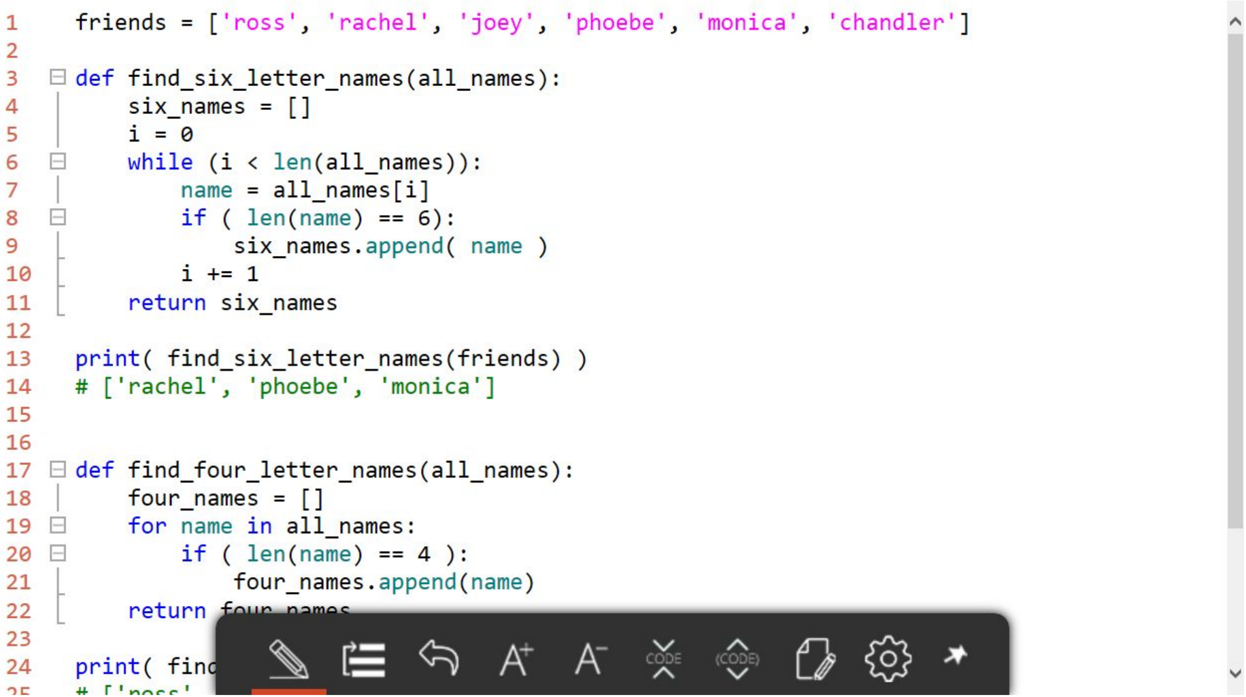
- Can store any type of object
- Lists are indexed like strings

DICTIONARIES



```python
teams = {
    'ironman': ['spiderman', "t'challa", 'vision'],
    'captain': ['bucky', 'hawkeye', 'scarlet witch']
}

print(teams['ironman'])
# ['spiderman', "t'challa", 'vision']

print('spiderman' in teams['ironman'])
# True

teams['captain'] = teams['captain'] + ['antman']
print(teams)
# {'ironman': ['spiderman', "t'challa", 'vision'], 'captain': ['bucky', 'hawkeye', 'scar

print(teams.keys())
# dict_keys(['ironman', 'captain'])

print(teams.values())
# dict_values([['spiderman', "t'challa", 'vision'], ['bucky', 'hawkeye', 'scarlet witch
```

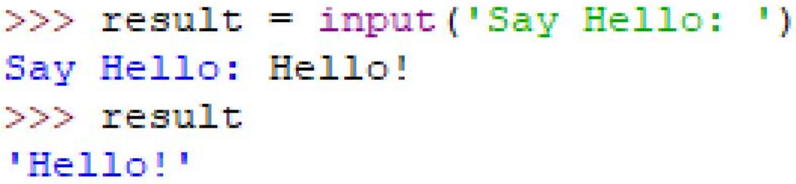
FOR AND WHILE LOOPS



```
friends = ['ross', 'rachel', 'joey', 'phoebe', 'monica', 'chandler']

def find_six_letter_names(all_names):
    six_names = []
    i = 0
    while (i < len(all_names)):
        name = all_names[i]
        if ( len(name) == 6):
            six_names.append( name )
        i += 1
    return six_names

print( find_six_letter_names(friends) )
# ['rachel', 'phoebe', 'monica']


def find_four_letter_names(all_names):
    four_names = []
    for name in all_names:
        if ( len(name) == 4 ):
            four_names.append(name)
    return four_names

print( fin
# ['ross'
```

- For while loops, don’t forget to increment the counter

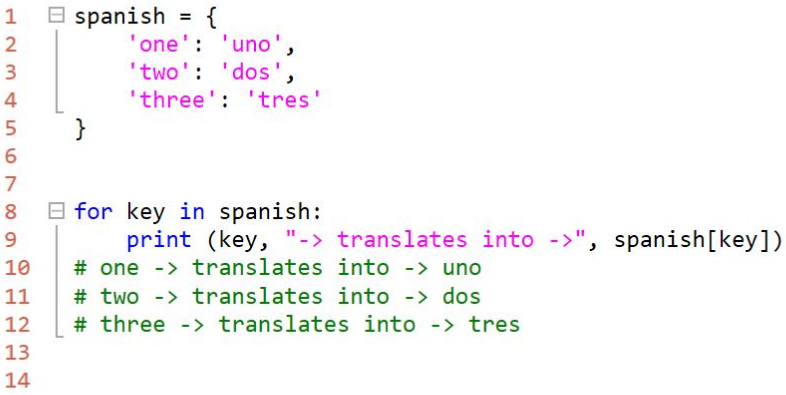
Input



```
>>> result = input('Say Hello: ')
Say Hello: Hello!
>>> result
'Hello!'
```

REVIEW

- Lists
- Dictionaries
- For loops
- While loops

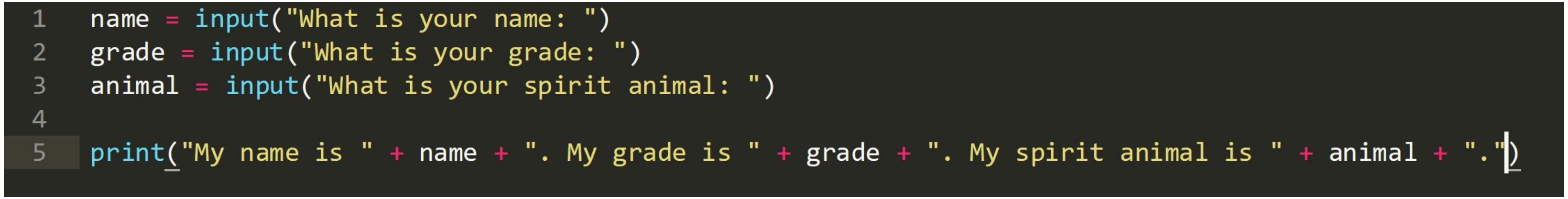


```python
spanish = {
    'one': 'uno',
    'two': 'dos',
    'three': 'tres'
}


for key in spanish:
    print (key, "-> translates into ->", spanish[key])
# one -> translates into -> uno
# two -> translates into -> dos
# three -> translates into -> tres


```

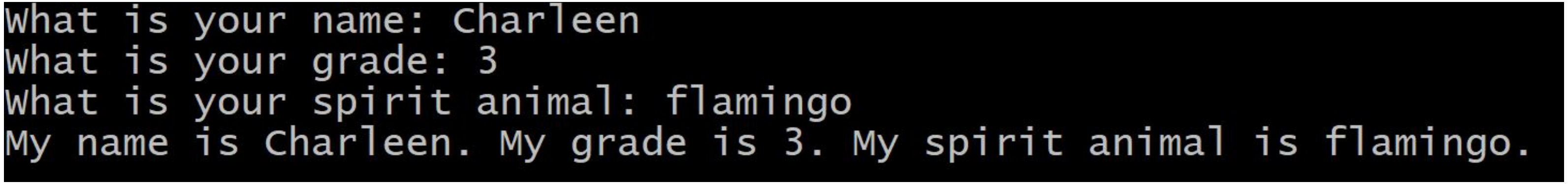
PRACTICE PYTHON!

- Lists
- Dictionaries
- For loops
- While loops

Introductions!

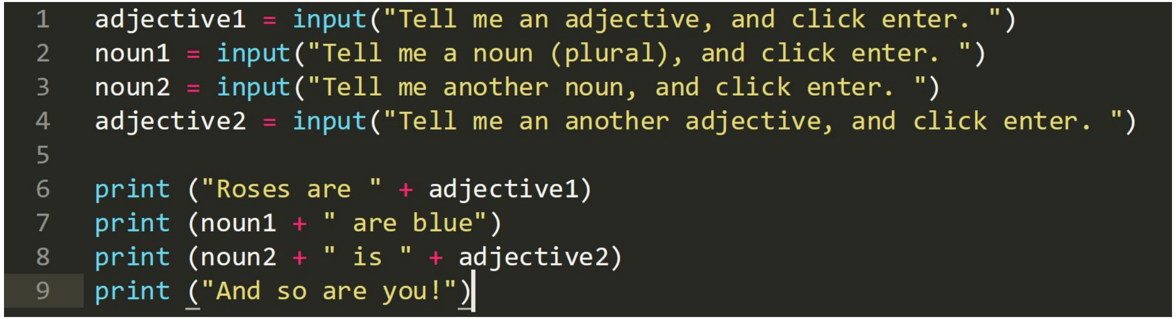


```python
name = input("What is your name: ")
grade = input("What is your grade: ")
animal = input("What is your spirit animal: ")

print("My name is " + name + ". My grade is " + grade + ". My spirit animal is " + animal + ".")
```

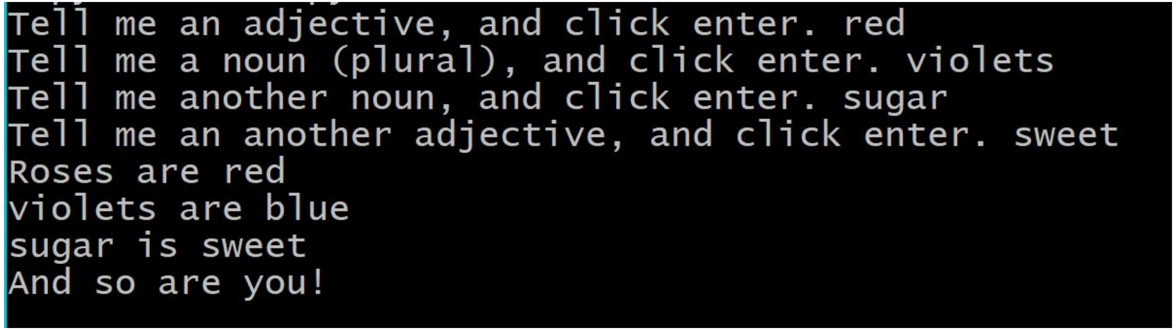


```
What is your name: Charleen
What is your grade: 3
What is your spirit animal: flamingo
My name is Charleen. My grade is 3. My spirit animal is flamingo.
```

Mad Libs!



```python
adjective1 = input("Tell me an adjective, and click enter. ")
noun1 = input("Tell me a noun (plural), and click enter. ")
noun2 = input("Tell me another noun, and click enter. ")
adjective2 = input("Tell me an another adjective, and click enter. ")

print ("Roses are " + adjective1)
print (noun1 + " are blue")
print (noun2 + " is " + adjective2)
print ("And so are you!")
```



```
Tell me an adjective, and click enter. red
Tell me a noun (plural), and click enter. violets
Tell me another noun, and click enter. sugar
Tell me an another adjective, and click enter. sweet
Roses are red
violets are blue
sugar is sweet
And so are you!
```

Choose your own adventure!

POSSIBILITIES

- Simple Games
  - Hangman (http://inventwithpython.com/chapter9.html)
  - Mad Libs Generator
  - Text Based Adventure Game
  - Tic Tac Toe
- Websites (Flask example)
- Parsing Files (https://github.com/kakyoin01/python-web-scraper-example)

OTHER TOPICS AND APPLICATIONS

- List Comprehensions
- Recursion
- Documentation
- Classes
- Inheritance
- Efficiency
- Debugging

- *Django* and *Flask*

(Frameworks for Web Development)

- *BeautifulSoup*

(HTML parsing)

- *Tensorflow, numpy, matplotlib*

(Machine Learning)

- *Pandas*

(Data analysis)

Brainstorming Time!

Brainstorming

Defer judgement!

There's no such thing as a bad idea

Encourage wild ideas

Build off each other

Challenge assumptions

Theme:
Community

15 minutes

Write down as many ideas as you can

Team with most ideas gets a prize!

> I'll bring something tomorrow :)

Refining Ideas

Go through your list of ideas

Highlight your favorite ones

Tomorrow will be idea selection!