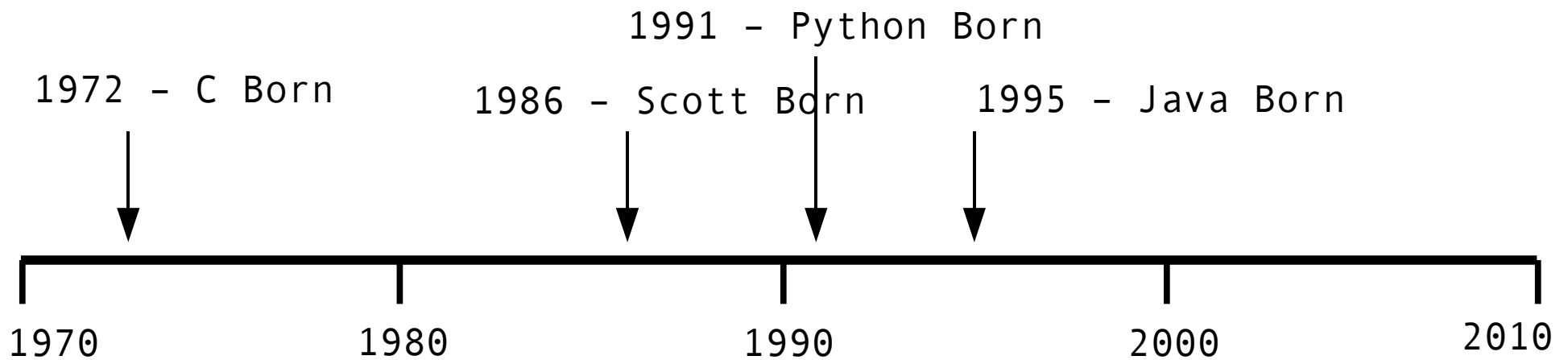python™

8/2/07

>>> About Me

Scott Shawcroft

`tannewt@cs.wa...`

* Junior
* Computer Engineering
* Third Quarter TA

* Creative Commons Intern
* Small-time Open Source Developer

python™

>>> History

1991 – Python Born

1972 – C Born

1986 – Scott Born

1995 – Java Born

1970   1980   1990   2000   2010

python™

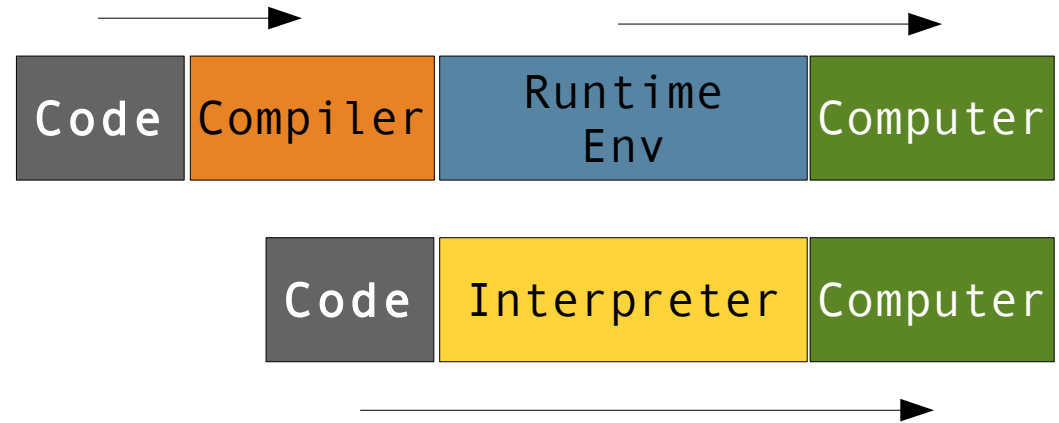# ››› Python Basics

* interpreted
* targeted towards short to
medium sized projects
* useful as a scripting
language

A whaa?

script – short
program meant for
one-time use.

Code	Compiler	Runtime Env	Computer

Code	Interpreter	Computer

python™

››› Getting it Going

Windows

1) Download Python from python.org.
2) Run 'python' using the run command.
       -or-
   Run Idle from the Start Menu.

Mac OSX

1) Python is already installed.
2) Open a terminal and run python or run Idle from finder.

Linux

1) Chances are you already have Python installed. To check run python from the terminal.
2) If not, install python through your distribution's package system.

python™

››› Topic Review

* cover through last week
* printing - System.out.println();
* methods – public static void <name>() ...

Hello.java

```
1  public class Hello {
2      public static void main(String[] args){
3          System.out.println("Hello world!");
4      }
5  }
```

Hello.java

```
1  public class Hello {
2      public static void main(String[] args){
3       hello();
4      }
5      public static void hello(){
6       System.out.println("Hello \"world\"!");
7      }
8  }
```

python

>>> Missing Main

Hello.java

```
1  public class Hello {
2      public static void main(String[] args){
3       System.out.println("Hello world!");
4      }
5  }
```

The entire file is interpreted as if it was typed into the interpreter. This makes it easy to write scripts with Python.

hello.py

```
1  print "Hello world!"
2
3
4
5
```

python™

>>> Printing

Hello.java

```java
1  public class Hello {
2      public static void main(String[] args){
3       System.out.println("Hello world!");
4       System.out.println();
5       System.out.println("Suppose two " +
6       "swallows carry it together.");
7      }
8  }
```

Escape sequences:
* \t – tab
* \n – new line
* \" – "
* \\ – \

hello.py

```python
1  print "Hello world!"
2  print
3  print "Suppose two swallows carry it
   together."
4
```

python™

>>> Methods

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

hello.py

```python
def hello():
    print "Hello \"world\"!"

hello()
```

>>> Whitespace

Unlike in Java, in Python whitespace (think tabs and spaces) matters. Instead of using braces ({}) to designate code blocks, Python uses the indentation. This was done to promote readable code. In Java you may or may not indent and it will work. In Python you **must** indent.

Hello.java

```java
1  public class Hello {
2      public static void main(String[] args){
3       System.out.println();
4      }
5  }
```

hello.py

```python
1  def hello():
2      print "Hello \"world\"!"
3
4  def does_nothing():
5      pass
6
7  hello()
```

>>> Comments

```java
1  // This is a file full of comments.
2  /* oooh
3     multi-line */
```

While commenting is flexible it is good practice to comment code with **#** comments before code in question. Comment methods with a doc string (**"""**) on the first line in a method that way it will be used in **help**().

comments.py

```python
1  # This is a file full of comments and code.
2
3  def hello():
4      """print 'Hello world'
5      this comment can be multi-line"""
6      print "Hello world" # this is in-line
7  hello()
```

python™

>>> help()

Python can read the
function comments and
make them available
when you need them.
To access these
comments call
help(<function>).

comments.py

```python
1  # This is a file full of comments and code.
2
3  def hello():
4      """prints 'Hello world'
5      this comment can be multi-line"""
6      print "Hello world" # this is in-line
7  hello()
```

```
>>> hello()
Hello world
>>> help(hello)
Help on function hello in module __main__:

hello()
    prints 'Hello world'
    this comment can be multi-line
```

python™

›› Example 1

```
scott @ yossarian ~ $ python  quote.py
A "quoted" String is
'much' better if you learn
the rules of "escape sequences."

Also, "" represents an empty String.
Don't forget: use \" instead of " !
'' is not the same as "
```

>>> Example 2

```java
// Marty Stepp, CSE 142, Autumn 2007
// This is the final version of the Figures program.
// It uses static methods to capture structure and remove redundancy.
//
// This text is a comment.  Comments let us document or explain our programs.
// The Java compiler ignores comments.  (They aren't printed on the console.)
//
public class Figures3 {
    public static void main(String[] args) {
        egg();
        cup();
        stop();
        hat();
    }

    public static void egg() {
        eggTop();
        System.out.println("\\        /");
        System.out.println(" \_____/");
    }

    public static void eggTop() {
        System.out.println("  _____");
        System.out.println(" /      \\");
        System.out.println("/        \\");
    }

    public static void cup() {
        System.out.println("\\        /");
        System.out.println(" \_____/");
        System.out.println("+--------+");
    }

    public static void stop() {
        eggTop();
        System.out.println("|  STOP  |");
        System.out.println("\\        /");
        System.out.println(" \_____/");
    }

    public static void hat() {
        System.out.println("  _____");
        System.out.println(" /      \\");
        System.out.println("/        \\");
        System.out.println("+--------+");
    }
}
```
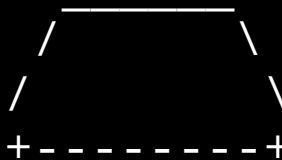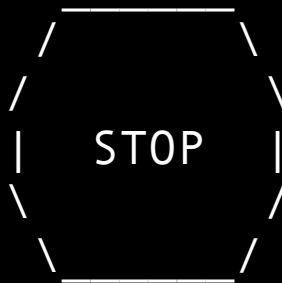
```
scott @ yossarian ~ $ python  quote.py

       _____
      /      \
     /        \
     \        /
      _____/

     \        /
      _____/
     +--------+

       _____
      /      \
     /        \
     |  STOP  |
      \        /
       _____/

       _____
      /      \
     /        \
     +--------+
```

Python® and the Python logo are either a registered trademark or
trademark of the Python Software Foundation. Java™ is a trademark or
registered trademark of Sun Microsystems, Inc. in the United States
and other countries.