

Introduction to Java for Python Programmers




















Dominique Thiebaut
CSC212 — Fall 2014



Why Java?

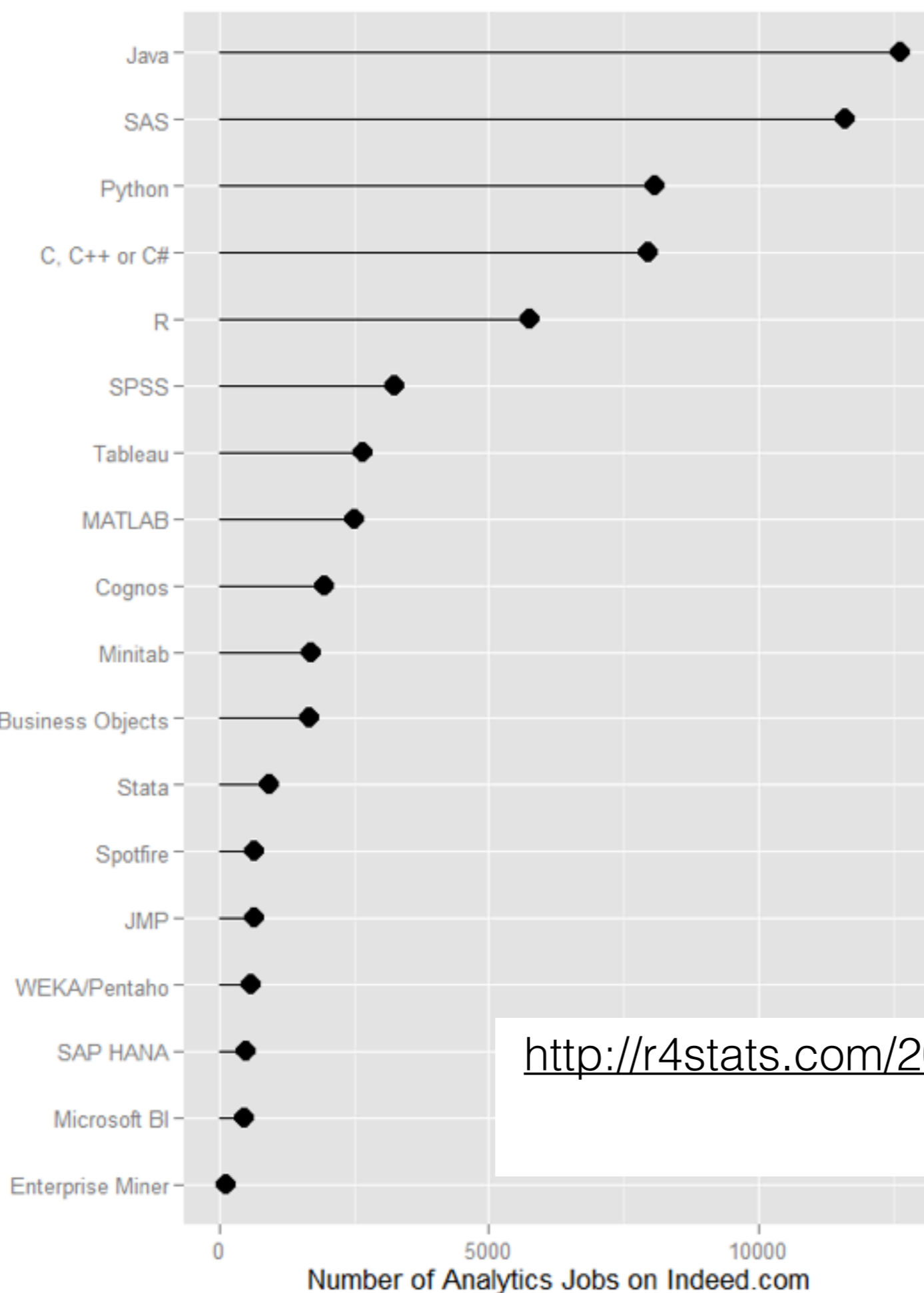
- More adequate for sophisticated projects.
- Python is great for **prototyping**. Java good for **development**
- Java programs run **faster** than Python programs
- **Structured** language: everything is a *class* or an *object*.
- **Easy to maintain** large projects with many classes.
- **Platform independent**

Ranking

Language Rank	Types	Custom Ranking
1. Java	  	100.0
2. Python	 	93.4
3. C#	  	92.2
4. PHP		84.6
5. Javascript	 	84.3
6. Ruby		78.6
7. PERL	 	70.1
8. HTML		66.1
9. Scala	 	62.5
10. Go	 	60.9

<http://spectrum.ieee.org/static/interactive-the-top-programming-languages>

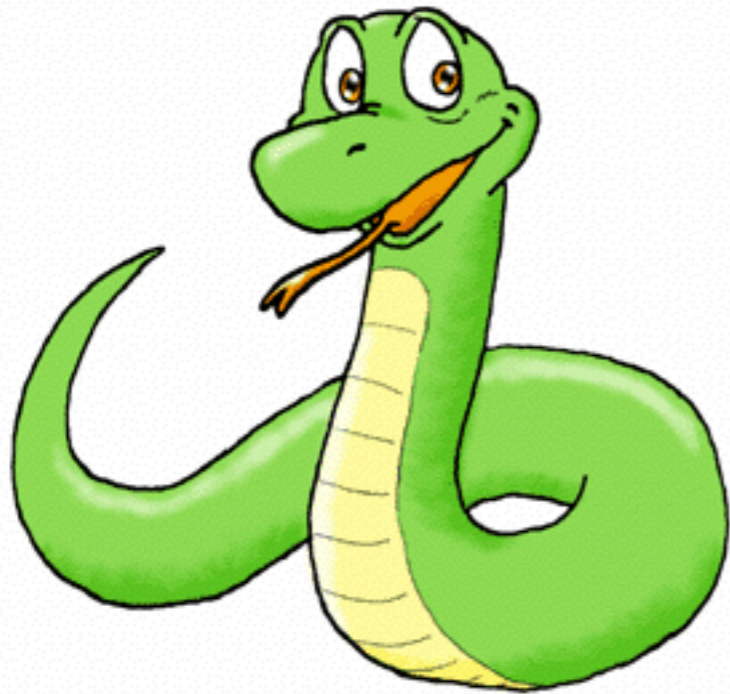
The More Popular Analytics Software



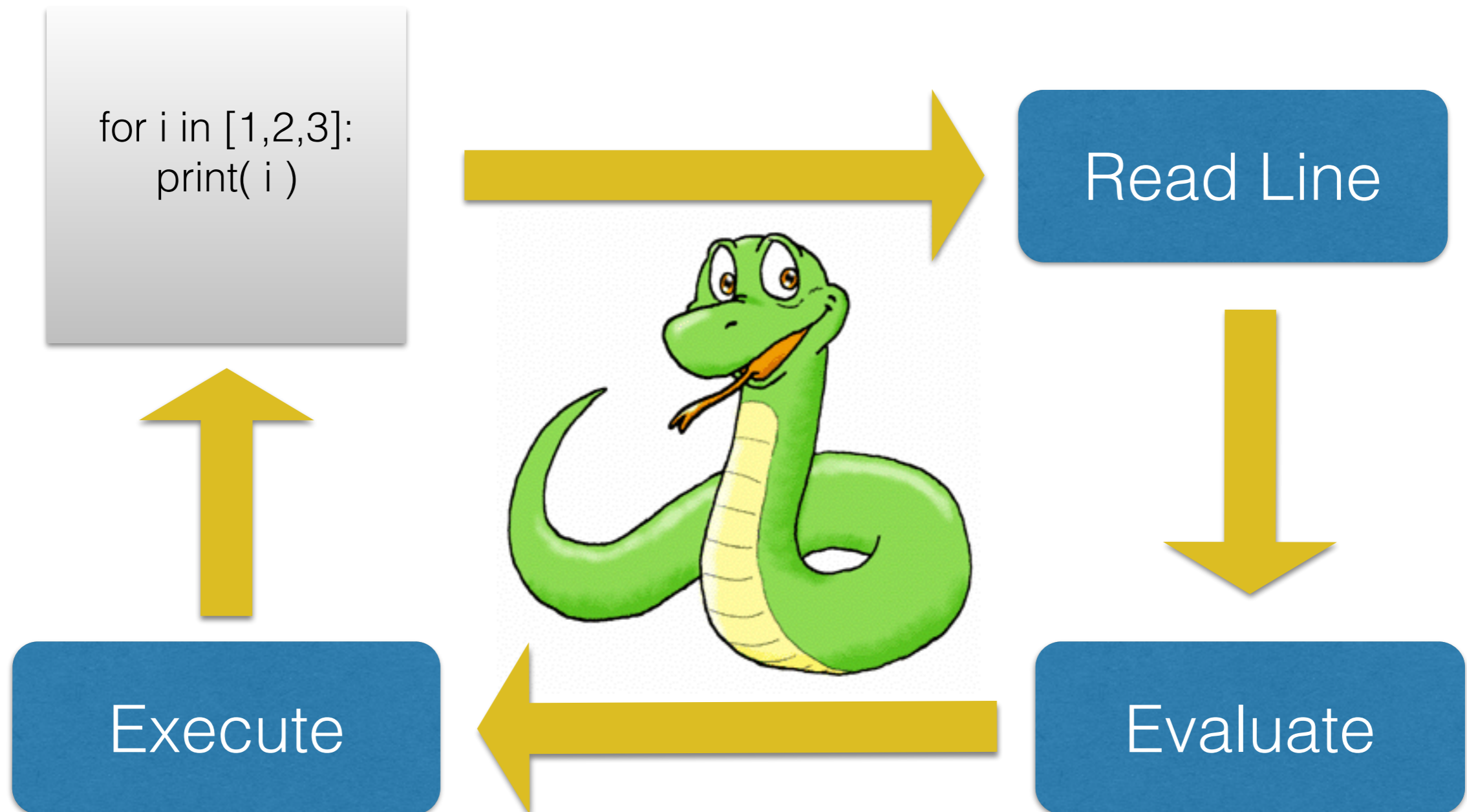
Jobs

<http://r4stats.com/2014/02/25/job-trends-improved/>
(Feb 2014)

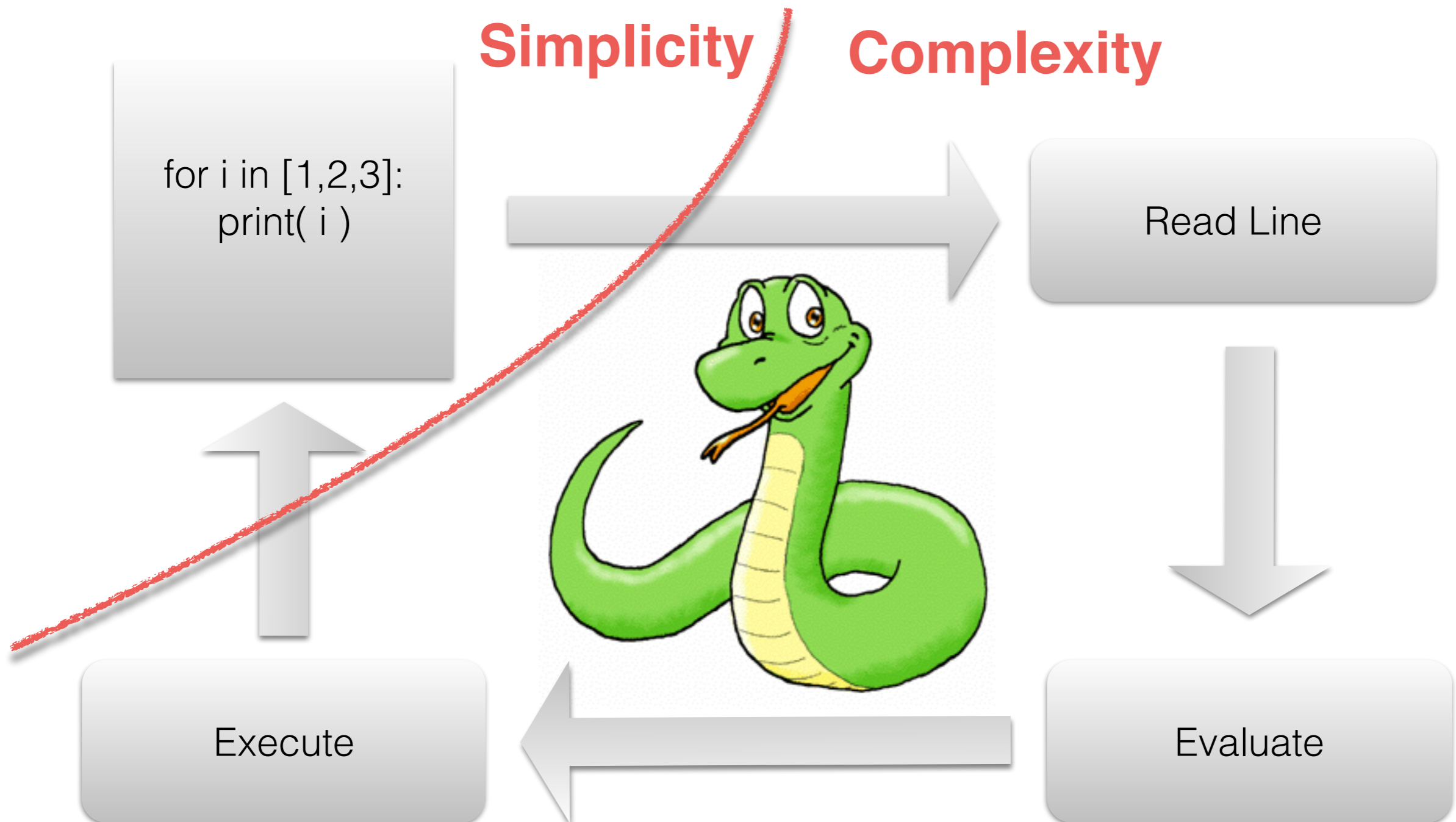
Interpreted vs Compiled



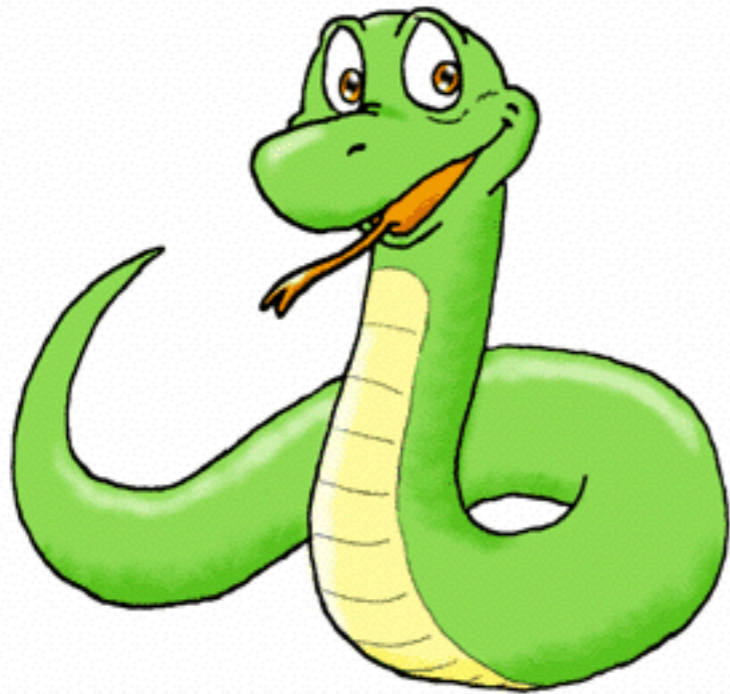
Interpreted vs Compiled



Interpreted vs Compiled



Interpreted vs Compiled



MyProgram.java

```
for ( int i=0; i<3; i++ ) {  
    System.out.println( i );  
}
```



Compiler



MyProgram.class

```
01010101 01000111 10101001  
01010001 00100101 11110011  
11010111 10101010 11010101  
01011111 00111011
```

(byte code)



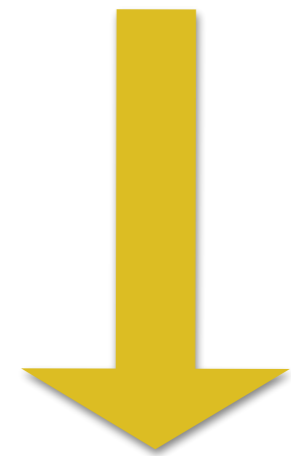
JAVA VIRTUAL
MACHINE
(JVM)

MyProgram.java

```
for ( int i=0; i<3; i++ ) {  
    System.out.println( i );  
}
```



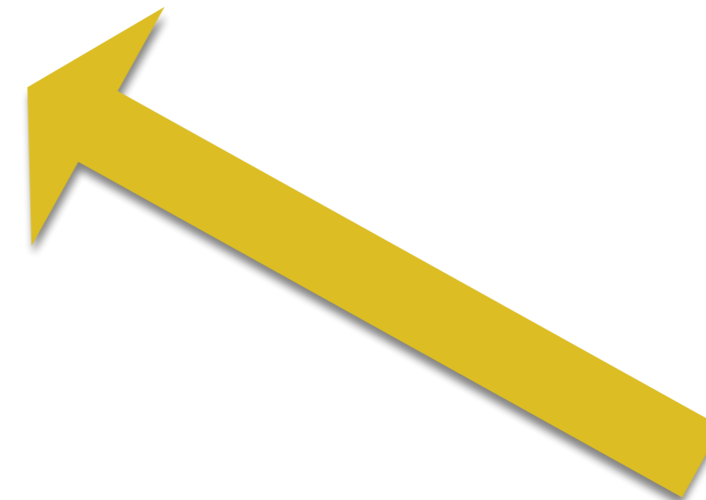
Compiler



MyProgram.class

```
01010101 01000111 10101001  
01010001 00100101 11110011  
11010111 10101010 11010101  
01011111 00111011
```

(byte code)



P
C

JAVA VIRTUAL
MACHINE
(JVM)

M
A
C

JAVA VIRTUAL
MACHINE
(JVM)

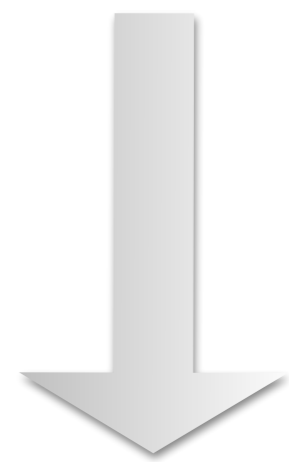
MyProgram.java

```
for ( int i=0; i<3; i++ ) {  
    System.out.println( i );  
}
```

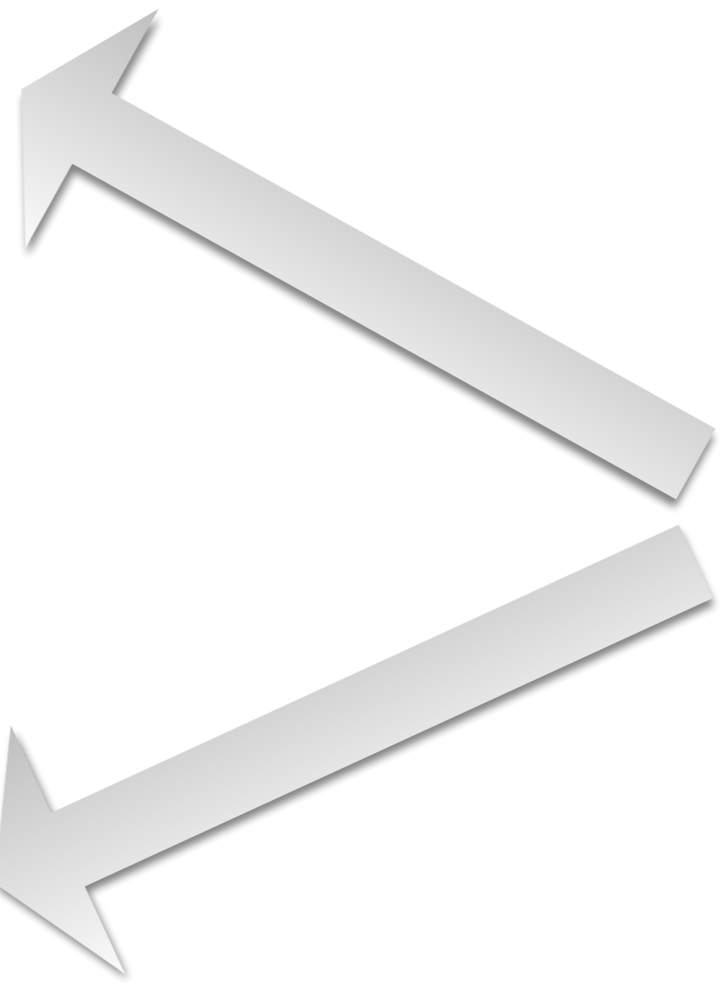
More Complex



Compiler



Simpler/Faster



**P
C**

JAVA VIRTUAL
MACHINE
(JVM)

**M
A
C**

JAVA VIRTUAL
MACHINE
(JVM)

MyProgram.class

```
01010101 01000111 10101001  
01010001 00100101 11110011  
11010111 10101010 11010101  
01011111 00111011
```

(byte code)

An Example:
Hello World!

An Example: Hello World!

- 1) Write the code
- 2) Analyze the different parts
- 3) Create & run real program

Hello World!

HelloWorld.java

```
class HelloWorld {  
  
    public static void main(String[] args) {  
        System.out.println( "Hello, World!" );  
    }  
  
}
```

```
emacs HelloWorld.java
```

```
javac HelloWorld.java
```

```
java HelloWorld
```



```
Hello, World!
```

```
emacs HelloWorld.java
```

```
javac HelloWorld.java
```

```
java HelloWorld
```



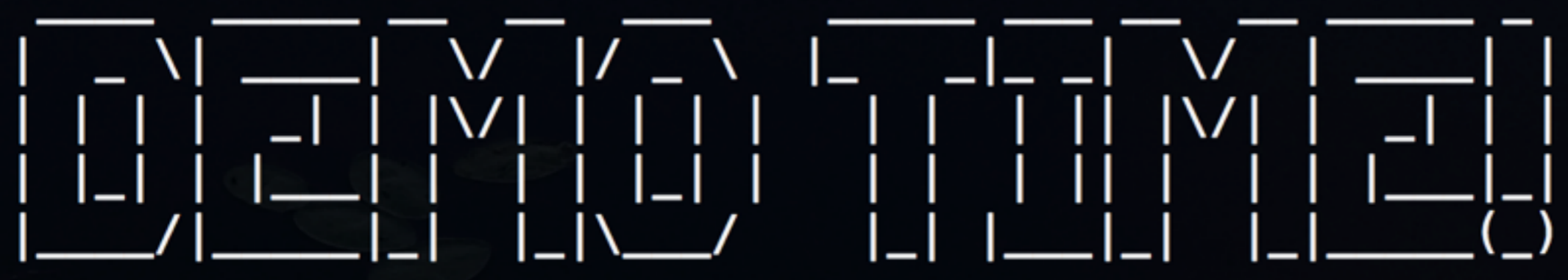
```
Hello, World!
```


bash

bash

[MacDom15]

[21:04:50] ~\$: figlet DEMO TIME!



[MacDom15]

[21:04:52] ~\$: █

```

// Lecture1.java
// D. Thiebaut
// A program that uses some simple Java features.

public class Lecture1 {

    // main program
    public static void main(String[] args) {
        // local variables
        String name = "Sophia Smith";
        int age = 24;
        double height = 94.5; // inches

        // printout a message on the screen
        System.out.println( name + " is " + age + " years old." );
        System.out.println( "Welcome to Smith College!" );
        System.out.println( "=====" );
    }
}

```



**KEEP
CALM
AND
JUST
CODE IT!**

CODELLI