

## CSCI 209: Software Development

Sara Sprenkle  
sprenkles@wlu.edu  
September 9, 2011

Sep 9, 2011

Sprenkle - CSCI209

1

## What is Programming?

"If you don't think carefully, you might think that programming is just typing statements in a programming language."

--Ward Cunningham

"Any fool can write code that a computer can understand. Good programmers write code that humans can understand."

-- Martin Fowler

"Refactoring: Improving the Design of Existing Code"

Sep 9, 2011

Sprenkle - CSCI209

2

## Discussion: What Is *Good* Software?

- What are its outcomes?
- What are the characteristics of the software?
- How can we write good software?
- What are short-term vs long-term goals?

Sep 9, 2011

Sprenkle - CSCI209

3

## Characteristics of *Good* Software?

- Free of bugs
  - Robust, reliability, stability
- Code is easy to read, extend, maintain
  - Readability, extensibility, maintainability
- Application is easy to use
  - Usability
- Efficiency
- Scalability

➔ Referred to as the *\*ilities*

Sep 9, 2011

Sprenkle - CSCI209

4

## Course Content

- Software Design Principles
- Java
  - **Statically** typed language
- Software development, productivity tools
  - Eclipse
  - Version Control Systems
  - Some UNIX

Sep 9, 2011

Sprenkle - CSCI209

5

## What to Expect from this Class

- Programming intensive
  - Interesting assignments and projects
  - More freedom in design, *\*ilities*
    - Larger portion of your grade
    - Correctness is **NOT** enough
  - Building on large library of classes
  - Read others' code! *Learn from the good and the bad*
  - Building larger applications
- Compare/Contrast with Python
  - PL design; what's the best PL for your needs
- Learning on your own
  - Online resources

Sep 9, 2011

Sprenkle - CSCI209

6

## Learning Objectives

- Discuss software development and practices **knowledgably**, using appropriate **terminology**
- Design, implement, test, and document efficient applications of **increasing size** and **complexity**
- Understand the designs and implementations of **others**
- Use a **version control system**, such as Subversion or CVS
- Use many of the capabilities of the **Eclipse IDE**
- Test and debug large applications **systematically**, using standard tools
- Understand **design principles** such as DRY and shy
- Discuss the benefits and limitations of a **statically typed** language

Sep 9, 2011

Sprenkle - CSCI209

7

## Class Details

- **Course Web Site** <http://www.cs.wlu.edu/~sprenkle/cs209>
  - **Example code, lecture slides, readings, resources**
  - **Course wiki**
- **Optional Textbooks**
  - **Use plentiful online resources instead!**
- **Participation**
  - **Class discussions**

Sep 9, 2011

Sprenkle - CSCI209

8

## Class Details

- **Programming Assignments**
  - **Various sizes**
- **Reading assignments**
  - **Write up short summaries on Sakai**
  - **Class discussions**
- **Several Projects**
- **2 Exams**

Sep 9, 2011

Sprenkle - CSCI209

9

## Course Dynamics

- |   |   |
|---|---|
| <ul style="list-style-type: none"> <li>• <b>Professor's Responsibilities:</b> <ul style="list-style-type: none"> <li>➤ Be <b>prepared</b> for class</li> <li>➤ Provide non-judgmental feedback to students</li> <li>➤ Treat students with <b>respect</b></li> <li>➤ <b>Challenge</b> and <b>encourage</b> students</li> <li>➤ Make material as clear as possible</li> </ul> </li> </ul> | <ul style="list-style-type: none"> <li>• <b>Student's Responsibilities</b> <ul style="list-style-type: none"> <li>➤ Be <b>prepared</b> for class (do readings and homework)</li> <li>➤ Give <b>attention</b> and <b>effort</b> in class to learning</li> <li>➤ Ask questions (<b>during class</b> and via email)</li> <li>➤ Use professor's office hours</li> <li>➤ Let professor know if something is going wrong</li> <li>➤ Treat other students and professor with <b>respect</b></li> </ul> </li> </ul> |
|---|---|

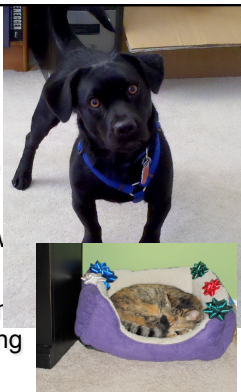
Sep 9, 2011

Sprenkle - CSCI209

10

## My Bio

- From Bellefonte, PA
- College of Delaware
- Gardening



Sep 9, 2011

Sprenkle - CSCI209

11

## My Research Interests

- **General: Software engineering**
- **Automated testing of web applications**
  - **Develop algorithms**
  - **Implement in tools**
  - **Empirical studies**
    - Try ideas out, see what actually happens, analyze

Sep 9, 2011

Sprenkle - CSCI209

12

## Your Bios

- Where you're from
- What activities you're involved in
- What you do in your free time
- Any experience with Java or Python?


Sep 9, 2011

Sprenkle - CSCI209

13

## What is Java?

... and, why should I learn it?

- From Sun Microsystems 
  - 1995, James Gosling and Patrick Naughton
  - Specifications
- Object-oriented
- Rich and **large** library
- Develop cross-platform applications
  - Web, desktop, embedded
  - Frameworks
- Widely used



Sep 9, 2011

Sprenkle - CSCI209

## What is Java?

- Java Programming Language
- Java Virtual Machine
- Java Class Libraries

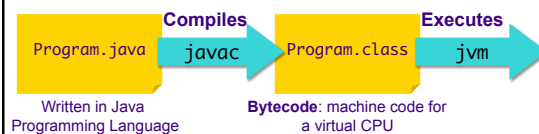
Sep 9, 2011

Sprenkle - CSCI209

15

## Overview:

### Writing, Executing Java Programs



Sep 9, 2011

Sprenkle - CSCI209

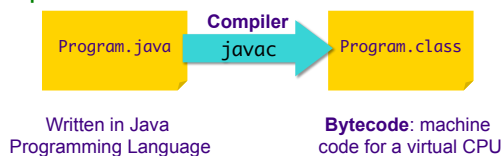
16

## Java Programming Language

- Similar to Python
- But *entirely* object-oriented\*

Label the  
Python equivalents

### Step 1:



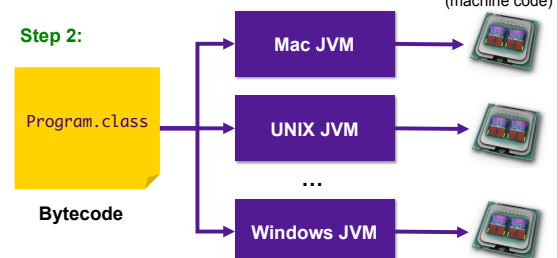
Sep 9, 2011

Sprenkle - CSCI209

17

## Java Virtual Machine (JVM)

### Step 2:



- Same **bytecode** executes on each platform
- Don't need to provide the source code

Sep 9, 2011

Sprenkle - CSCI209

18

## Java Virtual Machine (JVM)

- Emulates the CPU, usually specified in software
- Executes the program's **bytecode**
  - Bytecode: virtual machine code
- Different versions for each platform Java supports
  - Enables program portability
- Sun's Hotspot VM
  - Code dynamically compiled to machine code
- Garbage Collection

Sep 9, 2011

Sprenkle - CSCI209

19

## Traditional Way



- Executable is not portable

- How does Java's approach ease distribution of software?
- How is (I) Java and (II) the traditional approach the same and different from Python's approach?

Sep 9, 2011

Sprenkle - CSCI209

20

## Java Editions

- Edition*: Version of Java
- 3 different editions
  - SE: Standard Edition
  - EE: Enterprise Edition
    - Server-side applications
    - Web applications, Communication (mail)
  - ME: Micro Edition
    - For PDAs, mobile devices, etc.

Sep 9, 2011

Sprenkle - CSCI209

21

## Java Class Libraries

- Pre-defined classes
  - Included with Java Software Development Kit (SDK) and Java Runtime Environment (JRE)
  - View the available classes online: <http://download.oracle.com/javase/6/docs/api/>
- Similar in purpose to modules included with Python

Sep 9, 2011

Sprenkle - CSCI209

22

## Summary of Java Platform SE 6.0

Java Language	Java Language										
Tools & Tool APIs	java	javac	javadoc	apt	jar	javap	JPDA	jconsole			
	Security	Int'l	RMI	IDL	Deploy	Monitoring	Troubleshoot	Scripting	JVM TI		
Deployment Technologies	Deployment				Java Web Start				Java Plug-in		
User Interface Tools	AWT				Swing				Java 2D		
	Accessibility	Drag n Drop		Input Methods		Image I/O		Print Service		Sound	
Integration Libraries	IDL	JDBC™		JNDI™		RMI	RMI-IIOP		Scripting		
Other Basic Libraries	Beans	Intl Support		I/O		JMX	JNI		Math		
	Networking	Override Mechanism			Security	Serialization	Extension Mechanism			XML JAXP	
lang and util Basic Libraries	lang and util	Collections		Concurrency Utilities			JAR	Logging		Management	
	Preferences API	Ref Objects		Reflection		Regular Expressions			Versioning		Zip Instrument
Java Virtual Machine	Java HotSpot™ Client VM					Java HotSpot™ Server VM					
Platforms	Solaris™			Linux			Windows			Other	

Image from Oracle's site

Image from Oracle's site

Java 7 was released in July 2011

Sep 9, 2011

Sprenkle - CSCI209

23

## Benefits of Java

- Rapid development of programs
  - Large library of classes, including GUIs, Enterprise-level applications, Web applications
- Portability
  - Run program on multiple platforms without recompiling
- Statically-typed language
  - Compiling: find some errors before execution!
  - Can give performance boost by doing optimizations

Sep 9, 2011

Sprenkle - CSCI209

24

## Which 'Java' is this class about?

- Java programming language
- Java class libraries
- Use the JVM but won't learn about how it works
  - For more information:  
<http://java.sun.com/docs/books/vmspec/>

Sep 9, 2011

Sprenkle - CSCI209

25

## Summary

- Today
  - Overview of Java programming language
  - Example program
- Next time:
  - Data types
  - Control structures
- Your To Do
  - Review UNIX commands
  - Get comfortable with the command line
    - Run your Python scripts from the command line
  - Check out course web site

Sep 9, 2011

Sprenkle - CSCI209

26