

# Welcome to CSSE 220

- We are excited that you are here:
  - Start your computer
  - Pick up a quiz from the back table
    - Answer the first two questions

# Course Introduction, Starting with Java

CSSE 220—Object-Oriented Software Development

Rose-Hulman Institute of Technology

# Agenda

- Instructor intro
- A few administrative details
- Verify Eclipse and Subclipse configuration
- Java *vs.* Python
- Examine and modify simple Java programs

# Instructor Info

- Amanda Stouder
  - Assistant Professor for the Practice of Computer Science and Software Engineering
  - You can call me:
    - Amanda
    - Professor Stouder
  - Don't call me:
    - ~~Mrs. Stouder~~ (please not this)
    - ~~Dr. Stouder~~ (not this either, I'll explain)

# Instructor Info

- Education
  - Rose-Hulman Institute of Technology 2008
    - B.S. in Computer Science and Software Engineering
- Experience:
  - Amadeus Consulting, 2008-2010
  - Software Engineering Professionals (SEP), 2010-2014
  - Stouder Software Consulting, LLC (2014-present)
  - RHIT: Visiting Professor CSSE (2013-2016)

# Instructor Info

- Office Hours:
  - MWF 7:30am – 8:30am, 1:30pm – 4:30pm
  - OFF CAMPUS: Tuesdays and Thursdays
  - Always by my email!
  - If you need a different time, email me and we'll work something out

# Instructor Info

- Laptop Policy
  - I'm trying something new this quarter
  - Why?
  - I will ask you to close/open laptops as we move through the class
  - When closed, I need you to pay attention
    - This also means no phones/other devices
  - Closing far enough so you can't see the screen is fine
  - What if I take notes on my laptop?

# Daily Quizzes

- I expect you to answer every question.
  - Including the last two, at least put N/A
- Stop me if I don't cover a question!

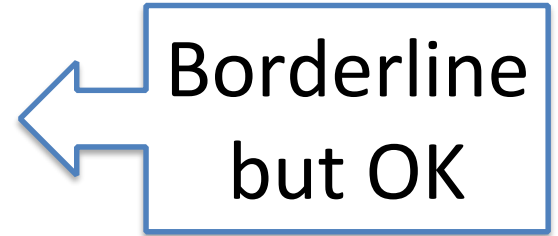


# A Tour of the On-line Course Materials

- Moodle
- Schedule
- Syllabus

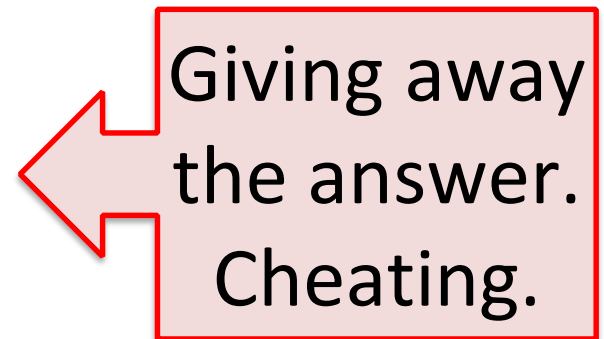
# How much help is too much help?

“So you loop across the array elements, getting each element and seeing if it’s in the hashtable. If it is, you get the value and increment it...”



Borderline  
but OK

“So you write a for loop, 1 to array length. Your key variable is gonna be array[i]. You check if hashMap.get(key) is null, if not, you get the value with get, then hashMap.put(key, oldValue + 1)...”



Giving away  
the answer.  
Cheating.

# Definitely not OK

- Looking at someone else's solution, "just for a reference" as you write your own code
- Pair programming on an individual assignment
- Sitting next to someone as you both work on the assignment, working through any problems you have as a group

# Penalties – they are severe

- Automatic F in the course
- Drop 1 letter grade
- -100% score on assignment

Yes, you can get an automatic F for cheating on one assignment one time.

You should always credit anyone you get help from on an assignment. If you do, it lets me assign one of the weaker penalties.

# Programming is not a spectator sport

- And neither is this course
- Ask, evaluate, respond, comment!
- Interrupt me! Even with statements like, *“I have no idea what you were just talking about.”*
- I do not intend for classroom discussions to go over your head. Don't let them!

# Ok, let's write our first Java program!

- Hello world

# Check the Repository Folder

- Click Start → Computer
- Double-Click “Local Disk (C:)”
- Double-Click “EclipseWorkspaces”
  - If it doesn’t exist, create it
- Verify that you have a folder named “csse220”
  - If it doesn’t exist, create it
- If you have taken the course before:
  - Rename the existing folder to “csse220-old”
  - Create a new folder named “csse220”

# Opening Eclipse

- Start Eclipse
  - Go to C:\Program Files\eclipse
  - Double-click “eclipse.exe”
- When prompted for the workspace, enter:
  - C:\EclipseWorkspaces\csse220
- If not prompted for the workspace, after Eclipse loads:
  - Click File → Switch Workspaces → Other
  - Enter path above



# Select Perspective

- Look at the top-right corner of Eclipse
- If “Java” is selected, do nothing and wait for next slide
- Otherwise:
  - Click Window → Perspective → Other...
  - Select “Java”
  - Click OK

# Get SVN Menu

- If SVN menu not shown at the top of the screen:
  - Click Window → Perspective → Customize Perspective
  - Click “Command Groups Availability” OR “Action Set Availability”
  - Scroll down and check “SVN”
  - Click “OK”

# SVN Menu Not Available

- You need to install Subclipse
  - Help → Install New Software
  - Add...
    - Name: Subclipse
    - Location:  
[http://subclipse.tigris.org/update\\_1.8.x](http://subclipse.tigris.org/update_1.8.x)
  - Check Subclipse AND SVNKit
  - Continue clicking Next, Accepting terms, and then click Finish when available

# SVN Repositories Window

- You can also display the SVN Repositories Window by doing the following:
  - Click Window → Show View → Other...
  - Expand SVN
  - Select “SVN Repositories”
  - Click OK

# Add Your Repository

- Click SVN → “Checkout projects from SVN”
  - Select “Create a new repository location”
- Click Next
- Type the following URL, replace the **user** in blue with your username:  
`http://svn.csse.rose-hulman.edu/repos/csse220-201720-user`  
Mine would be:  
`http://svn.csse.rose-hulman.edu/repos/csse220-201720-stouder`
- Click Next

# Checkout Project for Today

- If you received an error at the end of the last slide,
  - let myself or a TA know immediately
  - Use <https://svn.csse.rose-hulman.edu/password/> to reset your SVN password
- Otherwise, expand your repository and select “JavaIntro”
- Click Finish
  - Do the same for HW1 now if you’d like, or you can wait and check it out later

# Show Package Explorer

- If JavaIntro did not show up in the Package Explorer (defaults to the left):
  - Click Window → Show View → Package Explorer

# HelloPrinter.java

- To run a Java program:
  - Right-click the .java file in Package Explorer view
  - Choose **Run As → Java Application**
- Change the program to say hello to a person next to you
- Introduce an error in the program
  - See if you can come up with a different error than the person next to you
- Fix the error that the person next to you introduced



# A First Java Program

In Java, all variable and function definitions are inside *class* definitions

main is where we start

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

**System.out** is Java's standard output stream. This is the variable called **out** in the **System** class.

**System.out** is an *object* from the **PrintStream** class. **PrintStream** has a *method* called **println()**.

# Introduction to Java

# Things Java Has in Common with Python

- Classes and objects
- Lists (but no special language syntax for them like Python)
- Standard ways of doing graphics and GUIs
- A huge library of classes/functions that make many tasks easier
- Nice integration with the Eclipse IDE

# Why Java?

- Widely used in industry for large projects
  - From cell phones
    - including smart phones—Android platform
  - To global medical records
- Highlights essential topic of the class – Object Orientation
- Similar to other popular languages C#, Objective-C
- Less complex than C++
- Most popular language according to the TIOBE Programming Community Index [March 2016]

<http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>

## Interlude: JavaScript and Java

# Java is to Javascript as Ham is to Hamster

From Wikipedia (edited, bullets added to enhance PowerPoint readability):

- The change of name to JavaScript roughly coincided with Netscape adding support for Java technology in its web browser.
- The name caused confusion, giving the impression that JavaScript was a spin-off of Java.
- The choice has been characterized by many as a marketing ploy by Netscape to give JavaScript the cachet of what was then the hot new web-programming language.
- It has also been claimed that the language's name is the result of a co-marketing deal between Netscape and Sun, in exchange for Netscape bundling Sun's Java runtime with its then-dominant browser.

# Basic Java Functions and Conditionals

- Let's go through the ConditionalExamples.java file

# What are Types?

- All variables in Java have a “type”
- Describes the data that can be stored in a variable
  - String – text only
  - short/int/long – whole numbers only
  - float/double – numbers with decimals
  - boolean – true or false
  - char – a single text character
- Classes – Class names are also types, let you define your own, more complex, types

# Strings


- `String myString = "hello";`
- `String otherString = new String("hello2");`
- Java's way of storing text data
- Has many handy functions like `substring`, `charAt`, etc. that you will slowly learn
- But how do you find out about these cool functions?



# Java API Documentation

- What's an API?
  - Application Programming Interface
- The Java API on-line
  - Google for: **java api documentation 7**
  - Or go to: <http://download.oracle.com/javase/7/docs/api/>
  - Also hopefully on your computer at  
C:\Program Files\Java\jdk1.7.0\_9\docs\api\index.html

You need the 7 (or 8) to get the current version of Java

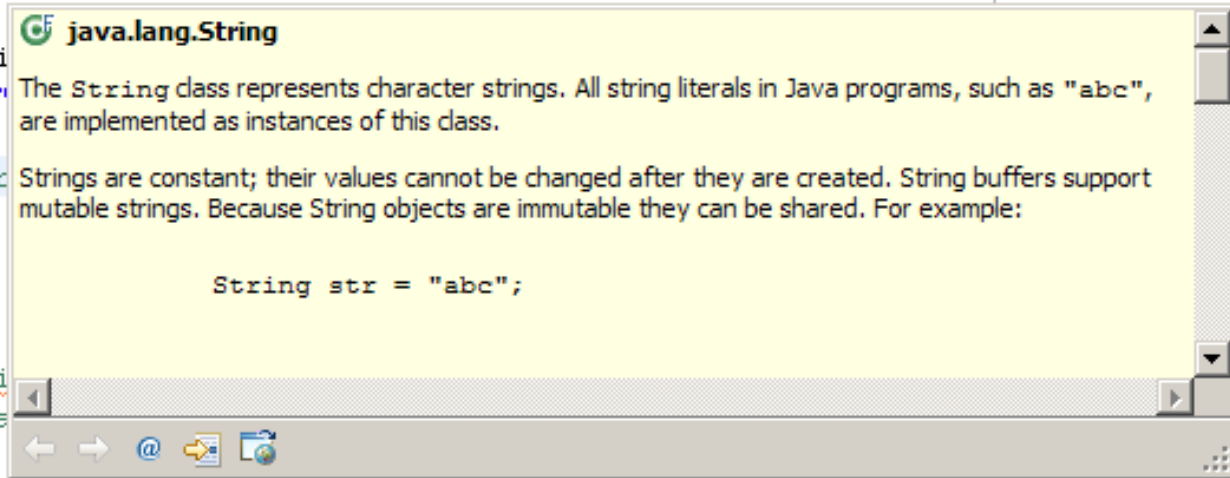


**Note:** Your version may be something other than 7.0\_9. We recommend that you bookmark this page in your browser, so you can refer to it quickly, with or without an internet connection.

# Java Documentation in Eclipse

- Setting up Java API documentation in Eclipse
  - Should be done already,
- Using the API documentation in Eclipse
  - Hover text
  - Open external documentation (Shift-F2)

```
main(String[] args) {  
    is a  
    JOpti  
    at ln("   
    VG: Pr  
    ring:  
    Verldi  
    e's e
```



**java.lang.String**

The `String` class represents character strings. All string literals in Java programs, such as `"abc"`, are implemented as instances of this class.

Strings are constant; their values cannot be changed after they are created. String buffers support mutable strings. Because `String` objects are immutable they can be shared. For example:

```
String str = "abc";
```

# Exercise

- Work on StringProbs

**HW1 DUE  
BEFORE NEXT SESSION**

**IT'S ON THE SCHEDULE PAGE.**

**(IT IS YOUR RESPONSIBILITY TO KEEP UP WITH THE SCHEDULE  
PAGE)**

**AS ALWAYS, EMAIL ME IF YOU  
HAVE ANY QUESTIONS**