EE363 Fall 2015

Monday, August 24, 2014

Syllabus

- Will e-mail link shortly…
- Let's review

Course Info

- Timothy C. Fanelli
  CAMP 143 ??
  (315) 261-9711
  tfanelli@{us.ibm.com, clarkson.edu}, tim@fanel.li
- Lecture is M/W/F @ 2:00PM
  Meet in Science Center 356
- We will meet **sometimes** via Web Conference

Grading

- 3 Exams: 2 hourly worth 15% each, 1 cumulative final worth 20%
- 4 (ish) Projects: 2 worth 10%, and 2 worth 15% each (typically)
- Each exam and project will be given a **letter grade**

Grading (cont'd)

- Letter grades will be converted to numerics based on the following table:

A+	A	A-	B+	B	B-	C+	C	D+	D	D-	F
97	93	90	87	84	81	77	74	68	60	50	0

Grading (cont'd)

- Your final grade will be computed, and converted back to a letter grade using the table below.

A+	A	A-	B+	B	B-	C+	C	C-	D	F
97	93	89	86	82	79	76	72	68	60	0

Projects

- We'll have four (ish) projects this semester
  - At least "Project 1" will be made up of various homework assignments through the semester.
- Each project will have a design component, and an implementation component
- The design portion is a written report describing the solution
- The implementation portion is a Java implementation of your solution
- The design and implementation portions may be weighted differently depending on the project.

Ethics

- Anything with your name on it *MUST* represent your own work.
- No excuses, no second chances.

Ok... so, on to it then...

- First things first -- Java?
  - Yes: Java.
  - Java is the new black. It's everywhere, and it's not going anywhere any time soon.

But we've been using C++?

- Not any more... and you're welcome.
- The course description is out-dated, and needs to be changed.
- Using Java in this course:
  - Allows us to focus on higher level design concepts w/o getting wrapped up in language semantics
  - Broadens your knowledge-base as you begin looking at internships and full-time positions.

So C++ Is Dead?

- Absolutely not, and it's not going anywhere either.
  - You should be very familiar with C++ already. You will continue learning C++ in your other courses too.
  - Every design concept we learn this semester is applicable to C++ programming as well
- To set context - I spend significant amount of my professional career in COBOL!

Isn’t Java Dead?

- Absolutely not, and it’s not going anywhere either.
  - Firm foothold in enterprise IT
  - Java EE in Fortune 500’s
  - Hundreds of Millions per year investment!

Computer Languages

- The language itself is just a tool to express an idea.
- Like human languages, it's descriptive power is inherent in its syntax and grammar.
- Most ideas can be described in any language
- Some languages are better at some ideas than others.

Computer Languages

- To be clear:
  - Anyone who tells you C++ is better than Java is blind.
  - Anyone who tells you Java is better than C++ is naive.
  - Anyone who learns COBOL can be filthy rich.
- So why Java? Because I said so.

But I Don't Have a Java IDE?

- No worries... you can get one for free:
  - http://eclipse.org
  - Download Eclipse IDE for Java Developers (Java EE is fine too)
- If you already know Java and use a different IDE: **keep using it.**
  - One of my all-time favorites is JetBrain's Intelli-J Eclipse
- Slides and tutorials will be build on Eclipse.

Hello, Java!

- Live Demo of Eclipse
- We'll create a simple program: Hello, Java!
- Discuss the Eclipse interface components, and the parts of a Java program compared to C++

Watch the Demo Later on the EE363 Video Podcast:

http://ee363f15.fanel.li/

Java Classes and Source Files...

- The organization of source files in Java project may be different than you're used too in other languages...
  - Single class per source file (.java)
  - Single source file per class (ie, no separate "header" file)
  - A class is defined in a package. The package is similar to a "namespace," but also corresponds to the folder under "src" where it is stored.

Define a New Class

- Use the Eclipse “New class” wizard to create a new class.
  - Specify the class name, and package name
  - If the package doesn’t exist, it’ll be created.
  - The source file will be the class name with a “.java” extension

Java Conventions

- A quick note on conventions... the Java specification has a preferred convention to increase readability:
  - Class-names are UpperCamelCase
  - Method and field names are lowerCamelCase

Package Names

- Convention dictates that a developer use their web address components backwards as a top-level package name.
- Use additional qualifiers to differentiate your development efforts.
- For example:

  http://www.timfanelli.com
  com.timfanelli.ee363.hellojava

Package Names

- If you have a personal domain name, use it! If not, I recommend:
  edu.clarkson.<username>.ee363
- Avoid package names like “project1” or “project2” etc...
  - They’re fine... just not very descriptive.

Running a Java Program

- This is where things get very different from what you may be used too...
- ANY class may have a “main” method:

```
public final static void main( String[] args) {
        ...
}
```

- When you invoke your program, you specify *which* class to execute.

Design Patterns vs Principles

Design Principles

- Objectives of good design
- Often idealistic, and sometimes overkill
- We'll learn several principles, and attempt to strike a balance between good design and complexity

Design Patterns

- Well known, documented techniques that are applicable to an entire class of design problem
- We use design patterns to achieve design principles
- Design patterns increase readability and maintainability!

Everything You Know is Wrong

Or, at least, just not good enough…

Liskov Substitution

Let *q(x)* be an property provable about objects *x* of type *T*.
Then, *q(y)* should be provable about objects *y* of type *S*
where S is a subtype of T

Liskov Substitution

Put more simply: squares are not rectangles.