

CSE 7345/5345

Advanced Application Programming / Fall 2009

Instructor: Dr. Frank P. Coyle
coyle@engr.smu.edu
214.768.3086 (voice) 214.768.3085 (fax)

Web Page: <http://www.engr.smu.edu/~coyle/cse7345>

Course Objectives: Advanced Application Programming is designed to take programmers to the next level of expertise in programming focusing on topics that enable web/internet applications. The emphasis in the course will be on the delivery of web services via Java Servlets, working with threads and concurrency, and working with Flex, the client side frameworks from Adobe.

Prerequisites

- All students are assumed to have basic Java programming skills and understand the basics of object technology.

Text

Goetz, Brian. Java **Concurrency in Practice**. Addison Wesley. 2006.

Recommended:

Cole, Alaric. **Learning Flex3**. O'Reilly. 2008

Online Readings:

Online Java Tutorial. <http://java.sun.com/docs/books/tutorial/java/TOC.html>

Online J2EE Tutorial (for Servlets) <http://java.sun.com/j2ee/1.4/docs/tutorial/doc/>

Grading:

Class participation	5%
Homework/Programming Assignments	35%
Online Quizzes	20%
Midterm Exam	20%
Final Exam	20%

Distance Students

Distance students will be given an extra 5 days to complete assignments and take the midterm.

Class Participation

Class participation can be achieved in two ways:

- asking and answering questions in class
- participating in the Blackboard discussion groups

Class participation can be reduced by gazing at your laptop during class.

Homework/Programming Assignments

Many of the assignments will require the deployment of your program as a java servlet. You will need to have an Lyle School of Engineering account in order to deploy your apps on the engineering school servers. If you need an Engineering account, obtain a form from: <http://lyle.smu.edu/co>

Online Quizzes.

There will be online quizzes associated with various topics. Quizzes will be open notes, book, web, Google -- whatever resources you can muster to answer the questions. Some quizzes may be timed.

Downloads

NetBeans.

<http://www.netbeans.org/downloads>

As of this writing, the official NetBeans versions is 6.7.

You are encouraged to use NetBeans since class examples will use NetBeans. It is advised that you download the version of NetBeans that includes the Apache Server so that you can test your apps locally before uploading to the engineering school servers.

MySQL

<http://dev.mysql.com/downloads/>

Possibly helpful link for MySql installation:

<http://nbguru.wordpress.com/2008/06/01/mysql-admin-and-netbeans/>

Working with MySQL from NetBeans:

<http://www.netbeans.org/kb/60/ide/mysql.html>

Flex

Flex Builder 3 is available for FREE to students.

<http://www.adobe.com/devnet/edu/>

Java Online API documentation

<http://java.sun.com/j2se/1.5.0/docs/api/>

Java Documentation.

It is recommend that you download and install your own copy of the [Java 2 Platform Standard Edition, API Specification](#) for the latest Java version.

<http://java.sun.com/j2se/1.5.0/docs/api/index.html>

The Art of Asking Questions.

- What is the stumbling block? You must be able to articulate it.
- See what Google has to say about your question.
- If you cannot find the answer on Google, try posting your question on one of the class Blackboard discussion groups.
- If you do not get an answer from Blackboard, email drC with the course number (cse7345) in the email title.

Some Tips on Asking Questions.

From: <http://catb.org/~esr/faqs/smart-questions.html>

Be precise and informative about your problem

- Describe the symptoms of your problem or bug carefully and clearly.
- Describe the environment in which it occurs (machine, OS, application, whatever). Provide your vendor's distribution and release level (e.g.: "Fedora Core 4", "Slackware 9.1", etc.).
- Describe the research you did to try and understand the problem before you asked the question.
- Describe the diagnostic steps you took to try and pin down the problem yourself before you asked the question.
- Describe any possibly relevant recent changes in your computer or software configuration.

Describe the goal, not the step

If you are trying to find out how to do something (as opposed to reporting a bug), begin by describing the goal. Only then describe the particular step towards it that you are blocked on.

Often, people who need technical help have a high-level goal in mind and get stuck on what they think is one particular path towards the goal. They come for help with the step, but don't realize that the path is wrong. It can take substantial effort to get past this.

Not Smart:

How do I get the color-picker on the FooDraw program to take a hexadecimal RGB value?

Smart:

I'm trying to replace the color table on an image with values of my choosing. Right now the only way I can see to do this is by editing each table slot, but I can't get FooDraw's color picker to take a hexadecimal RGB value.

The second version of the question is smart. It allows an answer that suggests a tool better suited to the task.

Asking drC for Clarification.

I'm happy to clarify details about any assignment. If you have questions about an assignment, please realize that I do not have every assignment memorized and may be reading your email at a Starbucks. Please provide the context of your question.

Not Smart:

In that last assignment, where do we send stuff?

Smart:

In assignment 3.b, it was posted that:
Please upload source code.
Where do I upload?