Programming for the WWW

with Emphasis on Java related technologies

Professor Poelman - Fall 2002 NYU
My Philosophy

• My **job** is to teach you as much as you can learn as fast as you can learn it, *whether you like it or not*
• Technology churn is such that what I teach you today will be obsolete in a year!
• Architecture changes slower than application technology
• Teach you to **think** more than teach you a **tech** skill
• Java is a must have technology not esoteric
• Java is your **next legacy technology** – like Assembly, COBOL, Fortran, Basic, PowerBuilder, Pascal, C++, LISP, SmallTalk, …
Philosophy (cont)

• **You’re all adults** – you’re responsible for what you get out of this class
• **Don’t know?** – ASK! (At the appropriate time – think office hours, email). Utilize your classmates.
• Knowledge = Power = Freedom = $$$
• You should **master**, yes, master this material! Know it backwards and forwards, up and down. Know it better than me (when you’re done)!
• **Life is short** – don’t waste time not learning as much as you can, when you can.
• **More to life than technology** – balance is important
Philosophy (cont)

• **Cheating** means you don’t trust ME! And that I can’t trust you. If you aren’t getting it, let me know. Maybe you need it presented in a different way or alternative approach. If you get caught cheating i.e. **not thinking for yourself and doing your own work**, you will get an “F” on that assignment or test. Period.

• **Simple test** to judge whether its cheating “Would you be embarrassed to tell your parents, friends and ME that’s how you did it?” (not a perfect test but pretty easy.) Cheating robs you of the feeling of success. You don’t need to cheat to succeed in this course. You need to work hard (some harder than others).

• **Hardwork and Frustration** – Frustration = large effort with no results, Hardwork = lots of effort with great results. No one is ever proud of their frustration. I am here to work you **hard** NOT to **frustrate** you.
My background

• **Project Expertise** – 15+ years experience in industry
  – Architect on JPMorganChase Global Financial Information cache J2EE + JMS Messaging deployed around the world US, EMEA and Asia
  – Lead Architect on BankBoston Reengineer ($250 MM USD budget)
  – EAI/Messaging Architect Merrill Lynch Direct Markets Web Portal
  – R&D Manager for Electronic Imaging Company
  – Developer of video game Ajax™ on IBM PC 8mHz 8086 1 meg ram,
  – Neuropsychology research in pain – built experimental control / analyzed data
  – Device drivers for DSP60001 board for Mach
  – Device drivers for highspeed image scanners /optical disk jukeboxes/image compression.

• Currently **writing a book**

• **Managing Consultant** for Valtech Technologies, Inc.
  – consulting company in NYC + 15 worldwide offices 1000 people worldwide

• **Languages** — Assembly, Basic, DSP Basic, C, C++, Forth, Snobol, Fortran, VisualBasic, Java, C#, VB.net.

• **Platforms** — DOS, Amiga, Windows 3.x/9x/NT/2K/XP, Unix (BSD/Mach/Solaris/Linux), CICS, Java/J2EE

• **DCT** (Distributed Computing Technology)
  – Sockets, Shared DB/ODBC, COM/DCOM/COM+/.net, CORBA, RMI, EJB/J2EE, Messaging (MQ Series, MSMQ, TIBCo, JMS)
Why are you here?

- Interested in WWW applications
- Interested in Java
- Interested in Architectures
- Seemed like a good idea at the time
- Want to learns something that you can apply
- Other?
Goals of Course

• Understand the architectures of WWW applications
• Understand a selection of technologies
• Understand the value of an OO approach
• Learn Java and gain experience with Java building web applications
• Learning to think in new ways
Technology churn

- **HTML & Web Servers** were advanced technology in 1997
- **CGI & Java** were advanced technology in 1998
- **Servlets & EJB** was advanced technology in 1999
- **XML** was advanced technology in 2000
- **SOAP** was advanced technology in 2001
- **WebServices, Portals & BEEP** were advanced technology in 2002
- **WSRP, WSUI, GRID, Pervasive Computing** will be advanced in 2003
<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture Topic</th>
<th>Homework Assigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sept 9</td>
<td>Class Intro, WWW Architecture Overview</td>
<td></td>
</tr>
<tr>
<td>Sept 16</td>
<td>OO and Java Intro</td>
<td>#0 (Downloads)</td>
</tr>
<tr>
<td>Sept 23</td>
<td>Inheritance, Interfaces, Exceptions</td>
<td></td>
</tr>
<tr>
<td>Sept 30</td>
<td>Swing &amp; JDBC</td>
<td>#1 (Swing)</td>
</tr>
<tr>
<td>Oct 7</td>
<td>Collections &amp; Threads</td>
<td></td>
</tr>
<tr>
<td>Oct 14</td>
<td>Network &amp; RMI</td>
<td>#2 (JDBC)</td>
</tr>
<tr>
<td>Oct 21</td>
<td>Threads</td>
<td></td>
</tr>
<tr>
<td>Oct 28</td>
<td>HTML Pages, Servlets, JSP</td>
<td>#3 (RMI)</td>
</tr>
<tr>
<td>Nov 4</td>
<td>Java Beans, Reflection</td>
<td>MIDTERM Due</td>
</tr>
<tr>
<td>Nov 11</td>
<td>J2EE</td>
<td>#4 (JSP)</td>
</tr>
<tr>
<td>Nov 18</td>
<td>J2EE (Cont)</td>
<td></td>
</tr>
<tr>
<td>Nov 25</td>
<td>Web Services - HTTP, SOAP, WSDL, UDDI</td>
<td>#5 (SOAP)</td>
</tr>
<tr>
<td>Dec 2</td>
<td>Security, SSL, Certs,</td>
<td></td>
</tr>
<tr>
<td>Dec 9</td>
<td>BEEP, XML, XSL/XSLT, XForms</td>
<td></td>
</tr>
<tr>
<td>Dec 16</td>
<td>Misc Discussions</td>
<td>FINAL Due</td>
</tr>
</tbody>
</table>
Misc Class Info

• **Office Hours**
  – 6-7pm Mondays 4th Floor Warren Weaver Room 401

• **Class Web Page**
  – check it every week

• **You needs a CIMS account**
  – email [csgrad@cs.nyu.edu](mailto:csgrad@cs.nyu.edu) to request a new account.
Misc Class Info

• My email
  – poelman@cs.nyu.edu send me an email 1st week of class from your official email account

• Email distribution list
  – http://www.cs.nyu.edu/mailman/listinfo/g22_3033_009_fa02

• My cell phone
  – 917-783-7477 (emergencies only)
TAs

• Kranthi
  – kranthi@cs.nyu.edu
  – Office: #530, WWH

• Prashant
  – puniya@cs.nyu.edu
  – Office: #505, WWH

• TA office hours will be posted by second week of class on the class website.
Class Lecture & Tests

• **All** lecture notes will be available on or BEFORE the class – reading/browsing ahead is recommended
• **All** homework assignments will be posted by the third week – start early, avoid the rush.
• Midterm and Final will be administered via the website!
• Check the site Monday before class EVERY week for updates, changes, notes, etc! – this is REQUIRED!
Text Books

• None are Required!?
  – I’ll give recommendations but most info can be gotten on the web (free) or at B&N for browsing
  – Reserved copies at library
  – Look before you buy – it’s your cash
  – Different people prefer different presentation formats for info – tutorial, hands on, theoretical, “for dummies”, - you make the call
  – Just because they aren’t required doesn’t mean you don’t need to get the info from somewhere – borrow, buy, browse, download and print, xerox a friend’s copy, hack the code, …
Suggested Text Books

• “Java in a Nutshell” – O’Reilly (basic and J2EE versions)
• “Learning Java” – O’Reilly
• “Taming Java Threads” – Apress
• “Just Java and Beyond” – Prentice Hall / Sun
• Get it with Java 1.4 version info
• Rule of thumb - Never buy a computer book with a copyright of > 2 years ago
Grading Policy

- **5 Homework Assignments (5*6 = 30 points total)**
  - 2 points if it compiles
  - 2 points if it runs and matches functionality required
  - 1 point for readability/comments in code
  - 1 structure of code

- **2 Exams (30 + 40 = 70 points)**
  - Midterm (30 points) – take home/open web
  - Final (40 points) - take home/open web
  - emailed to you via the mailing list **Monday Night** after lecture
  - Open web you have until midnight Tuesday night
  - Web based exams – take and submit via email to your TA
Grading Policy (cont)

• Initial Grading Curve:
  – $\geq 90 \rightarrow A$
  – $\geq 80 \rightarrow B$
  – $\geq 70 \rightarrow C$
  – $\geq 60 \rightarrow D$
  – $\leq 60 \rightarrow F$

• Most of you should get A’s!
Homework Assignments

- Based on a fictitious bookstore Greenwich Village Books online
- Homeworks (tentatively):
  - Download and install the JDK, JRE and NetBeans IDE
  - A Swing UI based Java App that is a bookstore application that uses file IO.
  - Now using JDBC and a threaded application layer.
  - Swing UI talking RMI to a business logic layer talking to RDBMS Data Layer - server based 3 tier application.
  - Servlet based UI componentized 3 tier application talking RMI to existing server code.
  - JSP UI to the existing system.
Homework is due:

• Due 7 days after assigned in class (next lecture period)
• Late up to 14 days after assigned in class (2 lecture periods later). Late homework gets 50% of points it would have gotten if on time.
• After that 0 points, still need to submit them.
• Submit via the web to your TAs.
• Having problems contact your TAs! Or me (try them first).
Homework #0

• You need to download the Java 1.4 JDK and JRE
• You need to download the NetBeans IDE
• You need to install them
• You need to create a simple java program called “HelloWorld”
• You need to do this by Sept 25th!
Start NetBeans and choose “New” off the file menu
Click “new” on NetBeans objects

Create a new NetBeans object, for example:
- A Swing Form
- An Applet
- A JavaBean
- A Java Server Page

Or, select from the complete set of object templates.

You can double-click a method or variable in the Explorer to open the Source Editor and jump straight to that point in the source.
Go to “classes” and choose “main”
Then give it the name “HelloWorld” (exact case)
Click “next”
Click “next”
Click “next”
NetBeans generates a class called "HelloWorld.java"

/*
 * HelloWorld.java
 * Created on September 7, 2002, 2:26 PM
 */

/**
 * @author logan.poelman
 */
public class HelloWorld
{

 /** Creates a new instance of HelloWorld */
 public HelloWorld()
 {
 
 }

 /**
 * @param args the command line arguments
 */
 public static void main(String[] args)
 {
  System.out.println("Hello NYU");
  }

}
Then go to the “Build” menu and select “Execute” (or press F6). It will compile your HelloWorld.java file into a file called HelloWorld.class and then execute it invoking the method called “main()”
Files generated by the NetBeans wizard
Or you can compile your java file and then execute it from a command prompt (or shell).
useful URLs

- http://www.cs.nyu.edu/courses/fall02/G22.3033-009
- http://www.cs.nyu.edu/mailman/listinfo/g22_3033_009_fa02
- http://www.netbeans.org/