

Instructor: William Burrows  
Office: 324 Mackenzie, 543-4474, email — burrows@u.washington.edu  
WWW — <http://faculty.washington.edu/burrows>  
Office Hours: T, Th 1:00-3:00 and by appointment  
Text: *Core Java: Volume I-Fundamentals*, Horstman & Cornell  
*Core JSP*, Hougland & Tavistock

**Course Description.** This course is designed to help you gain an understanding of the object-oriented approach to developing and programming computer systems. The course will cover the following topics:

- Programming using Java and JBuilder 5.0
- General object-oriented programming concepts
- General object-oriented design approaches
- Web development using Java Server Pages

Object-oriented programming is a different programming paradigm compared to traditional, non object-oriented languages. It is not simply a few new features added to a programming language — it is a different way of thinking about and developing a programming solution to a problem. The process identifies classes that represent both data and behavior that the class can exhibit. Objects are created using class definitions. These objects are sent messages and they respond to these messages by exhibiting behavior. In addition, classes are members of a class hierarchy that allows a class to inherit the data and behavior of its ancestors in the hierarchy.

Java is a programming language that is both object oriented and rich in modern programming features. It supports the construction of platform independent applications, “applets” that can be embedded in web pages, as well as server-side code (Java Server Pages and Servlets).

Java Server Pages (JSP) provides a technology-neutral way of performing server-side programming using standard HTTP technology. It supports the entire set of classes available within the Java language. This means that JSP provides a very powerful set of classes to generate data aware HTML.

The course uses JBuilder 5.0 Pro from Borland — an object-oriented programming environment for Java. This is a sophisticated and powerful system that provides an environment very similar to Visual Basic to facilitate the construction of event-driven systems with power graphical user interfaces as well as Java Server Pages and Java Servlets.

**Requirements.** The course requires that you complete five programming assignments and two exams. The five assignments give you an opportunity to demonstrate your understanding of the programming concepts. The exams provide feedback on how well you understand the course topics. The assignments count for 50% of your grade and the exams 25% each. Be aware that much of the material covered in lecture is not covered elsewhere (including the textbook). Missed classes will significantly reduce your chances of successfully completing the course requirements and will likely impact your grade.

**Course Schedule (Revised November 1)**

Date	Topics	Readings*	Assignments
10/2 Session 1	Course introduction and introduction to Java and JBuilder 5.0		
10/4 Session 2	Java Fundamentals: Data Types, Operators, Strings, Flow of Control	CJ: Chapt 3	
10/9 Session 3	Java Fundamentals: Continued	CJ: Chapt 3	
10/11	<b>Classes Cancelled</b>		
10/16 Session 4	Objects and Classes	CJ: Chapt 4	<b>Assignment 1</b>
10/18 Session 5	Objects and Classes: Continued	CJ: Chapt 4	
10/23 Session 6	Inheritance	CJ: Chapt 5	
10/25 Session 7	Inheritance: Continued	CJ: Chapt 5	
10/30 Session 8	Composite Objects, Exceptions	CJ: Chapt 6	<b>Assignment 2</b>
11/1 Session 9	Java Interfaces, Hashtable		
11/6 Session 10	HTTP Basics, Scripting Basics, Actions & Directives, JSP Engine	JSP: Chapt 1 - 4	
11/8 Session 11	Retrieving Information	JSP: Chapt 5	
11/13 Session 12	Sending Information	JSP: Chapt 6	<b>Assignment 3</b>
11/15 Session 13	Tracking Sessions	JSP: Chap 7	
11/20 Session 14	Exam		
11/22	Thanksgiving Holiday		
11/27 Session 15	JDBC	JSP: Chapt 9	<b>Assignment 4</b>
11/29 Session 16	JDBC: Continued	JSP: Chapt 9	
12/4 Session 17	JSP & XML		
12/6 Session 18	JSP & XML: Continued		
12/11 Session 19	Course Wrapup		<b>Assignment 5</b>

\* CJ (Core Java), JSP (Core JSP)