

Programming the WWW

Handbook (Full Time)

2008/2009

Module Title: Programming the WWW

Module Code: BBM004

Level: 7

Module Availability: MSc Business Information Technology, MSc Information Technology for Business Management

Staff

Name	Room	EEmail	WWW
Chris Reade	341	Chris.Reade@kingston.ac.uk	http://www.kingston.ac.uk/~ku07009

Aims & Objectives

This module aims to provide students with a solid foundation of practical web programming experience, including:-

Elucidation of concepts and background for programming the World Wide Web.

Giving students *hands-on* experience with Web programming.

Identifying and discussing correctness and security issues related to Web programming.

Learning Outcomes

Students completing the module will be able to plan, design, construct, debug and test active web sites using an appropriate Scripting language.

They will have learned how to design and implement code in a structured way and how to reason about, monitor, debug and test the behaviour of their software.

Students will also be able to make appropriate decisions about client/server software positioning based on performance and security considerations.

Approach to Teaching & Learning

Lectures will be used to introduce key concepts and there will also be exercise classes. However, this module will be mostly laboratory based for hands on programming experience. That is, students will need to spend at least 3 hours a week programming in addition to the lectures and exercise classes. It is in the nature of computing and any sort of programming that you need to implement some small scripts each week so that you become familiar with the environment and (the inevitable) error messages.

You should read the following quote about *debugging* from one of the greatest computer scientists, the late Christopher Strachey, who said in 1966:

Although programming techniques have improved immensely since the early days, the process of finding and correcting errors in programming — known graphically if inelegantly as *debugging* — still remains a most difficult, confused and unsatisfactory operation. The chief impact of this state of affairs is psychological. Although we are happy to pay lip-service to the adage that to err is human, most of us like to make a small private reservation about our own performance on special occasions when we really try. It is somewhat deflating to be shown publicly and incontrovertibly by a machine that even when we do try, we in fact make just as many mistakes as other people. If your pride cannot recover from this blow, you will never make a programmer.

Scientific American 1966 vol 215 (3) September pp112-124

Students will be expected to read manuals as well as recommended texts to prepare for laboratory work. You are required to keep a lab logbook which is important for professional development as well as for progress monitoring.

The logbook should contain a record of work done in labs including:

1. Planning and initial designs.
2. An *Error Log* for each session and package you use — you learn more by paying attention to your errors.
3. You should keep a *Jargon Log* for each session and package. Many words are used to mean *different* things depending on the context and particular package — you only get used to this by recording all the meanings.

The format of the lab logbook is described in Appendix A on page 4 of this document.

The log should build into a valuable reference throughout the year.

Assessment

The assessment will be entirely through practical work (100% Coursework) The assessment takes the form of:

1. An initial assignment which is weighted as one quarter of the total coursework. This will be handed out in week 5 with a submission date of **9:00am Monday 16th March 2009**.
2. A second assignment which is weighted as three-quarters of the total coursework. This will be handed out in week 9 with a submission date of **9:00am Monday 27th April 2009**.
3. The lab log is required to be satisfactorily maintained and completed throughout and will be monitored regularly.

Teaching Programme

Below is the teaching programme for the course — any changes will be reported/recorded in the News section of the main Module Resource Page.

In *Induction Week*, you should have obtained your *Registration password* which will be needed to access a web server (studentnet.king.ac.uk) as well as for using other University machines.

- Week 1. WWW and Programming Concepts** An overview including HTTP, Browsers, HTML document structure (links, frames, forms), and introducing some general programming concepts and Scripting Examples.
- Week 2. Scripting Concepts** Embedded scripts, capabilities and security. Examples introducing Javascript and looking forward at AJAX technology.
- Week 3. Core JavaScript** An introduction to the language and programming concepts. Data types, variables, Statements and Functions with examples.
- Week 4. Core JavaScript** Arrays, Predefined Objects and functions.
- Week 5. Using Arrays and Objects** Examples to illustrate use of Arrays and Objects in solving programming problems.
- Week 6. Reading Week** — *review and work on assignment*
- Week 7. Designing Objects and Programming** Design and development of objects; Subtle language Issues; Methodical programming: Design, Structure, Style, Testing, Documentation.
- Week 8. Client Side Objects** ; (Navigator Objects) Including examples with events and event handling.
- Week 9. Modelling with Objects** Further Design with Objects (object oriented and object based programming techniques.)
- Week 10. Document Object Model** Programming with the DOM and event driven programming. Dynamic HTML.
- Week 11. Advanced Issues** JavaScript Object Model; Prototypes and Inheritance; The growing new technology of AJAX programming (Asynchronous JavaScript and XML); Advanced events and event handling;
- Week 12. Review** Plus work in Lab on assignment.

Reading & Other Resources

The main module resource page *progwwwFullTime* can be found on the Web at:

<http://www.kingston.ac.uk/~ku07009/Teaching/Progwww/progwwwFullTime.html>

This document (the Module Handbook) can be found at:

<http://www.kingston.ac.uk/~ku07009/Teaching/Progwww/progwwwFullTimeHB.pdf>

There is no one manual or book that covers all the material on this course.

Our Web sites contain many links to useful material — in each section of the course we shall indicate which we have found useful — it should also be a course where *you* find useful material we haven't yet seen — the nature of the WWW means that there is more out there than we could possibly have read!

Texts

There are lots of books on the WWW and JavaScript in particular. *Note that you should not use any book that does not cover JavaScript version 1.3 at least, and preferably 1.5* — this is the version that is implemented in recent browsers (Internet Explorer, Netscape, Mozilla Firefox, Opera, Safari).

The main course text is **Flanagan (1)**. (NOTE: the 5th edition) [This has some small changes and additions with some restructuring since the previous edition **Flanagan (2)**.]

To complement this, you may find useful one of the many other JavaScript texts available such as: Moncur (3), Negrino and Smith (4), Powers (5), Powers (6), Vander Veer (7), Morrison (8).

Web & HTML Background/Reference

You will probably need to refer to books on the Web and HTML. *Musciano & Kennedy (9)* is one of the best references on HTML and also introduces XML. *Niederst (10)* and *Spainhour & Eckstein (11)* are good references for HTML, JavaScript, CSS (Cascading Style Sheets), XML and CGI scripts. *Shklar & Rosen (12)* provides motivating background reading for JavaScript and other languages and web application architecture. *Chapman & Chapman (13)* is excellent for web design and has chapters on JavaScript and DOM (Document Object Model) scripting.

AJAX Texts

Many books on AJAX started to come out in 2006. AJAX (Asynchronous JavaScript and XML) concerns advanced use of JavaScript for highly interactive Web 2.0 applications which we will only touch upon at the end. Some example texts are Powers (14), Zakas et al. (15), van der Vlist et al. (16), Moore et al. (17), and Keith (18).

WWW References

<http://staffnet.kingston.ac.uk/~ku00597/JavaScript> (Phil Molyneux's JavaScript directories).

<http://www.ecma-international.org/publications/standards/Ecma-262.htm> This is the official ECMAScript standard.

Tutorials & Examples There are lots on the web – too many to list.

JavaScript Libraries

These are collections of commonly usable function and object definitions for developing larger scale programs in different areas of application (particularly AJAX applications). They are sometimes also called frameworks or widget libraries.

Dojo <http://dojotoolkit.org/>

Ext <http://extjs.com/>

jQuery <http://jquery.com/>

MochiKit <http://mochikit.com/>

MooTools <http://mootools.net/>

Prototype <http://prototypejs.org/>

Rico <http://openrico.org/>

Sarissa <http://dev.abiss.gr/sarissa/>

script.aculo.us <http://script.aculo.us/>

SproutCore <http://www.sproutcore.com/>

YUI <http://developer.yahoo.com/yui/>

Course Texts & Web Sites

- [1] David Flanagan. *JavaScript The Definitive Guide*. O'Reilly, fifth edition, 2006.
- [2] David Flanagan. *JavaScript The Definitive Guide*. O'Reilly, fourth edition, 2002.
- [3] Michael Moncur. *Teach Yourself JavaScript in 24 Hours*. Sams, Indianapolis, Ind., fourth edition, 2006.
- [4] Tom Negrino and Dori Smith. *Javascript and Ajax for the Web*. Peachpit Press, sixth edition, 2006.
- [5] Shelley Powers. *Learning JavaScript*. O'Reilly, 2006.
- [6] Shelley Powers. *Learning JavaScript*. O'Reilly, 2009.
- [7] Emily A. Vander Veer. *JavaScript for Dummies*. John Wiley, fourth edition, 2004.
- [8] Michael Morrison. *Head First JavaScript*. O'Reilly, 2007.
- [9] Chuck Musciano and Bill Kennedy. *HTML and XHTML: The Definitive Guide*. O'Reilly, sixth edition, 2006.
- [10] Jennifer Niederst. *Web Design in a Nutshell*. O'Reilly, third edition, 2006.
- [11] Stephen Spainhour and Robert Eckstein. *Webmaster in a Nutshell*. O'Reilly, third edition, 2003. ISBN 0-596-00357-9.
- [12] Leon Shklar and Richard Rosen. *Web Application Architecture: Principles, Protocols and Practices*. Wiley, 2003. ISBN 0-471-486656-6.
- [13] Nigel Chapman and Jenny Chapman. *Web Design: A Complete Introduction*. Wiley, 2006.
- [14] Shelley Powers. *Adding Ajax*. O'Reilly, 2007.
- [15] Nicholas C. Zakas, Jeremy McPeak, and Joe Fawcett. *Professional Ajax*. Wrox, second edition, 2007.
- [16] Eric van der Vlist, Danny Ayers, Erik Bruchez, Joe Fawcett, and Alessandro Vernet. *Professional Web 2.0 Programming*. John Wiley, 2007.
- [17] Dana Moore, Raymond Budd, and Edward Benson. *Professional Rich Internet Applications: AJAX and Beyond*. John Wiley, 2007.
- [18] Jeremy Keith. *Bulletproof Ajax*. New Riders, 2007.

A Lab Logbook

Keeping Laboratory Log Books

The main purpose of a laboratory log book is to help you become more efficient at learning by recording information about:

- problems
- errors
- where to find information
- plans/designs/aims
- outstanding tasks

It is part of your professional education to keep appropriate records of work you are doing. For example, in Industry, if you needed to be absent for a long period but you were working on critical projects, it is vital that you have sufficient records of progress, plans, unresolved problems and design decisions concerning your work so that someone else could step in and pick up where you left off with minimal disruption.

In an educational environment, you need to practice doing this. In addition, the log book is important for recording your own learning so that you can recall situations and problems you might encounter frequently. By keeping records, you can learn to become systematic and also learn what constitutes a good record from your own experience.

Although you are at liberty to tidy this information up afterwards, this should not be necessary. It is more important to learn to keep tidy notes while you work. It defeats the objective of keeping log books if you only ever write things up after the laboratory work has been done. That is why prior planning should also be recorded as well as problems with things you have not yet understood (such as error messages).

It is recommended that your log book should be a bound book of ruled sheets to encourage you to keep tidy records from the outset and not just collections of loose sheets created by forgetting to bring a log book to the laboratory sessions.

Your diligence in remembering to bring your log book to lab sessions and your record keeping will be monitored frequently during the courses in which you have laboratory sessions.