



COLLEGE OF INFORMATION STUDIES

Introduction to Programming for the Information Professional

INST 630, Fall 2013
Mondays, 2:00-4:45pm
Hornbake 0109

Instructor

Leah Findlater
Office: 2118A
Office Hours: Mondays, 5-6pm. Confirm with me by the end of class.

Course Description

This course is an introduction to computer programming intended for students with no previous programming experience. Topics include fundamentals of programming, such as variables, data types, assignments, selection, nesting, loops, arrays, functions, objects and storage. The course will also touch on current trends in user interface implementation that are relevant to information professionals, such as graphics, multitouch and gestural interaction, and mobile devices.

Student Learning Outcomes

Students will:

- be familiar with general terminology and approaches used by programmers that will enable them to talk to programmers.
- understand and be able to apply the fundamental concepts of computer programming, including variables, data types, assignments, selection, nesting, loops, arrays, functions, objects and storage.
- be able to apply at a basic level of competence the primary concepts of user interface programming, such as event programming, graphics and animation, and toolkits.
- build prototypes at varying levels of fidelity, from low-fidelity prototypes to a fully functional interactive application.

Textbook

Dionisio, J.D., and Toal, R. (2011). Programming with JavaScript: Algorithms and Applications for Desktop and Mobile Browsers, 1st ed. Burlington, MA: Jones & Bartlett Learning. ISBN-13: 978-0763780609.

Some places to get this textbook:

- As paperback or Kindle book from Amazon:
<http://www.amazon.com/Programming-With-Javascript-Algorithms-Applications/dp/076378060X/>
- As eTextbook from CourseSmart (cheaper than Kindle but not as flexible):
<http://www.coursesmart.com/9781449641948>

Tentative Course Schedule

Week	Topics and Assignments	Readings	Out	Due
Sep 9	Course introduction, variables, data types, operators. How HTML, CSS, JavaScript and they fit together.	Chapters 1 & 2	A1	
Sep 16	Conditionals & decision-making. Coding conventions and code quality tools.	Sections 3.1-3.5 & 4.1-4.3		
Sep 23	Iteration and how to approach program design (pseudocode). More in-depth coverage of data types, including string operations, undefined and null.	Sections 4.4-4.5.1 & 4.6	A2	A1
Sep 30	Arrays, Objects I. One-dimensional arrays and the use of for loops for storing and accessing array values. Object data types and basic constructors.	Sections: 3.6-3.9		
Oct 7	Functions I. Defining and calling functions, variable scope.	Chapter 5	A3	A2
Oct 14	Functions II. Validating arguments, preconditions. Passing objects as references.	Chapter 5		
Oct 21	Event programming. Defining user interface elements in HTML and programmatically accessing them, event handlers, timer events.	Chapter 6		A3
Oct 28	Midterm. Also: exception handling using try/catch/finally.			
Nov 4	Objects II. Prototypes, adding functions to objects (methods).	Chapter 7 (but only intro section of 7.7)	A4	
Nov 11	Software design and object-oriented programming. Inheritance, information hiding, and modularity. Performance and efficiency.	Chapter 7 (but only intro section of 7.7)		
Nov 18	Libraries (jQuery), web APIs. Using JavaScript libraries. Fundamentals of distributed computing, including data interchange formats, web communication and accessing web APIs (e.g., REST).	TBD	A5	A4
Nov 25	Graphics and animation. Drawing 2D graphics, transformations, animations, overview of 3D graphics capabilities.	TBD		
Dec 2	Storage. File input and output, local storage within the browser.	TBD		
Dec 9	Mobile applications & looking forward. Transitioning to other programming languages from this class.	TBD		A5
Dec 16	Final Exam			

Other Readings

The supplemental books are free of charge online to UMD students. The first two can be found in the "Safari Tech Books Online" database by first logging into <http://researchport.umd.edu>. Search for the database name, then for the book title once you're redirected to the Safari site:

- Freeman, E.T., and Robson, E. (2011). [Head First HTML5 Programming](#). O'Reilly Media.
- Crockford, D. (2008). [JavaScript: The Good Parts](#). O'Reilly Media.
- Haverbeke, M. (2011). [Eloquent JavaScript: A Modern Introduction to Programming](#). No Starch Press. Free online version: <http://eloquentjavascript.net/>

Many good resources are also available as webpages, including [W3Schools tutorials](#) on [HTML](#), [JavaScript](#), [CSS](#), [JSON](#). The [jQuery](#) webpage should also be useful.

Grading

Class Participation and Ungraded Exercises (10%)

The class will be a more rewarding experience if everyone actively participates. You will be expected to come to class prepared to contribute constructively to discussions, ask challenging questions, and help your classmates with in-class programming activities. Outside of class, you can participate by posting useful or interesting information on the course discussion website or visiting the instructor during office hours to ask questions or give feedback.

Graded Assignments (55%)

There will be 5-6 graded assignments. Unless specified by the instructor, graded assignments must be completed independently. Each assignment will consist of building an interactive webpage, possibly based on a code template provided by the instructor. Assignments will be skewed toward the first half of the semester to provide students with a base for the larger project. Assignment topics will build on each other, but the rough topic progression will be: (1) data types and conditionals, (2) arrays and loops, (3) functions and events, (4) objects, (5) 2D graphics and libraries. Ungraded programming exercises will also be assigned for the first 4-6 weeks of class, to guide students in practicing the fundamental concepts introduced in those weeks.

Midterm Exam (15%) and Final Exam (20%)

The midterm and final exams will cover all material up to that point.

Syllabus Change Policy

This syllabus is a guide for the course and is subject to change with advance notice.

Academic Integrity

What is academic dishonesty?

Academic dishonesty is a corrosive force in the academic life of a university. It jeopardizes the quality of education and depreciates the genuine achievements of others. Apathy or acquiescence in the presence of academic dishonesty is not a neutral act. All members of the University Community - students, faculty, and staff - share the responsibility to challenge and make known acts of apparent academic dishonesty.

Students have a responsibility to familiarize themselves with violations of the Code of Academic Integrity. Among these include:

- Cheating: "Intentionally using or attempting to use unauthorized materials, information, or study aids in any academic exercise."
- Fabrication: "Intentional and unauthorized falsification or invention of any information or citation in an academic exercise."
- Facilitating Academic Dishonesty: "Intentionally or knowingly helping or attempting to help another to commit an act of academic dishonesty."
- Plagiarism: "Intentionally or knowingly representing the words or ideas of another as one's own in an academic exercise."

For further clarification or information on the Code of Academic Integrity:

<http://www.studenthonorcouncil.umd.edu/code.html>

Students with Disabilities:

The University is legally obligated to provide appropriate accommodations for students with disabilities. The campus' [Disability Support Services Office](#) (DSS) works with students and faculty to address a variety of issues ranging from test anxiety to physical and psychological disabilities. If a student or instructor believes that the student may have a disability, they should consult with DSS (4-7682, email Dissup@umd.edu). Note that to receive accommodations, students must first have their disabilities documented by DSS. The office then prepares an Accommodation Letter for course instructors regarding needed accommodations. Students are responsible for presenting this letter to their instructors.

Classroom Environment

We will do live programming exercises most days, so bring your laptop and be prepared to write code. Any operating system will do. If you don't have access to a laptop, contact the instructor immediately after the first class to arrange for a loaner laptop from the iSchool.

Attendance Policy

[University policy](#) excuses the absences of students for illness, religious observances, participation in University activities at the request of university authorities and compelling circumstances beyond the student's control. Students who miss a single class for a medical reason are not required to provide medical documentation, but students who are absent more than once are responsible for providing various forms of documentation, depending on the nature of the absence.

Extensions

If you have to miss a deadline, you should inform the instructor as soon as possible, indicating when you will submit your work. The instructor will try to accommodate your needs. You should use this clause only for extraordinary personal reasons (e.g., personal illness, death in the family, etc.). The general policy is that late work will be deducted 20% of its total grade per calendar day, starting on the same day it is due. It is at the instructor's discretion to accept late work and assign late penalties.

Emergency Preparedness

<http://www.umd.edu/emergencypreparedness/>

CourseEvalUM

Brief Project Overview

Course evaluations are a part of the process by which the University of Maryland seeks to improve teaching and learning. The University Senate approved the implementation of a standard, online, University-wide course evaluation instrument. Each course evaluation contains a set of universal questions, and some are supplemented by questions from specific colleges. Students who leave no "Pending" evaluations in their Evaluation Dashboard each semester can view the aggregate results of a sub-set of universal items online.

Across the University, course evaluations are being administered through a web-based system dubbed CourseEvalUM. All information submitted to the Evaluation System is confidential. Instructors and academic administrators can only view summarized evaluation results after final grades have been submitted. Instructors and academic administrators cannot identify which submissions belong to which students. This standardized set of evaluation results provides the University with useful information on teaching and student learning across the campus.

For additional info see Student Fast Facts at:

https://www.irpa.umd.edu/Assessment/CourseEval/stdt_faq.shtml