





Computer Science (COMP) 266

Introduction to Web Programming (Revision 3)

Status: Replaced with new revision, see the [course listing](#)  for the current revision 

Delivery mode: [Individualized study online](#) 

Credits: 3

Area of study: Science

Prerequisites: Students are expected to be proficient users of the Internet and to be able to use word processors, text editors and file manipulation tools (including zip compression) effectively. If you do not have these skills you will not succeed and we highly recommend that you should take **COMP 200**, **COMP 210** or equivalent courses before starting this one.

Precluded: None

Challenge: COMP 266 has a challenge for credit option.

Faculty: [Faculty of Science and Technology](#) 

Notes:

Students who are concerned about not meeting the prerequisites for this course are encouraged to contact the **course coordinator** before registering.

Overview

COMP 266 is a beginners' course in programming using JavaScript, together with some HTML and CSS. It follows a problem-based approach which requires you to design and create a website of ever-increasing sophistication as the course progresses while creating design documentation, reflecting on the process, and (optionally) sharing and communicating with others on the course. The output of your work will be presented as a publicly accessible website, and you will submit a portfolio that maps what you have done to the course learning outcomes.

Outline

The units are weighted with respect to their relative contribution to the total portfolio grade.

Unit 0: Orientation

This unit has the information you need to successfully complete COMP 266.

Unit 1: Site Design — Weight: **10%**

Identify the audience, purpose, uses, and structure of your site. This is the basis for all the units that follow, each of which will relate explicitly to the personas, scenarios, and purposes that you create here.

Unit 2: HTML Site Building — Weight: **15%**

Learn the elements of HTML, using it to add content to your original design in the form of web pages.

Unit 3: CSS Site Styling — Weight: **15%**

Apply styles to improve the look, feel, and functionality of the content developed in Unit 2 as well as prepare for the programming you will learn in later units.

Unit 4: Script Use and Augmentation — Weight: **10%**

Find appropriate snippets of JavaScript code and to adapt them to work with your site as well as learn to read and critique JavaScript code.

Unit 5: Writing JavaScript — Weight: **30%**

This is by far the largest unit of the course, requiring you to add functionality to your site using JavaScript code that you have created yourself. It requires you to become proficient in the use of JavaScript commands, objects, functions, and tools. Topics addressed include loops, iterations, variables, arrays, objects, functions, regular expressions, events, manipulating the DOM, and recursion.

Unit 6: Using Libraries — Weight: **10%**

Extend the functionality of your site using external libraries such as JQuery.

Unit 7: Using External Site Data — Weight: **10%**

Use data from other sites to build mashups using AJAX to access APIs provided by those sites.

Learning outcomes

At the end of this course the successful student will be able to:

- apply a structured approach to identifying needs, interests, and functionality of a website.
- design dynamic websites that meet specified needs and interests.
- write well-structured, easily maintained, standards-compliant, accessible HTML code.
- write well-structured, easily maintained, standards-compliant CSS code to present HTML pages in different ways.
- use JavaScript to add dynamic content to pages.
- critique JavaScript code written by others, identifying examples of both good and bad practice.
- select appropriate HTML, CSS, and JavaScript code from public repositories of open-source and free scripts that enhances the experience of site visitors.
- modify existing HTML, CSS, and JavaScript code to extend and alter its functionality, and to correct errors and cases of poor practice.
- write well-structured, easily maintained JavaScript code following accepted good practice.

- write JavaScript code that works in all major browsers (including IE, Mozilla-based browsers such as Firefox, Opera, Konqueror, Safari, Chrome).
- effectively debug JavaScript code, making use of good practice and debugging tools.
- use JavaScript libraries (e.g. JQuery) to create dynamic pages.
- use JavaScript to access and use web services for dynamic content (AJAX, JSON, etc.).

Evaluation

Tutor approval is required at key points in the course before further units may be submitted.


Assessment for this course is based entirely on a portfolio of evidence, explicitly matched to the learning outcomes of the course and including personal reflections on the learning process kept in a reflective learning diary. Each student's work will be unique, so we encourage sharing and mutual support in this process, and allow you to such support and sharing with others as evidence that you have met the learning outcomes.


Evidence will consist of

- your website.
- your reflective learning diary, including reflections on the process, design documentation, and archives of earlier iterations of your site.
- supporting evidence, including contributions to discussions, help given to others, resources shared, questions asked.
- mapping of learning outcomes, in which you explicitly demonstrate how the work you have done (including reflections and other contributions as well as your site itself) proves that you have met the course learning outcomes. You are expected to estimate your own grades for this, which the marker may or may not agree with.


Note that there are several places along the way where formal feedback is given but no marks are allocated: this is intended to help you gauge your progress and to ensure that the work you engage with allows you to meet the outcomes as effectively as possible.

To **receive credit** [↗](#) for COMP 266, you must submit all course assignments and complete them to the satisfaction of your tutor. You must also achieve a

grade of 50 per cent on the Portfolio, and a course composite grade of at least **D (50 percent)** .

To learn more about assignments and examinations, please refer to Athabasca University's **online Calendar** .

Materials

This course either does not have a course package or the textbooks are open-source material and available to students at no cost. This course has a **Course Administration and Technology Fee** , but students are not charged the Course Materials Fee.

This course is entirely supported by online resources, some of which are provided by the course team, others that will be discovered and shared by students in the course.

Other Materials

A server space for your website will be provided along with a social environment which will allow you to create and share your reflective learning diary, design documentation, and resources you have discovered in your learning journey.

Special Course Features

Please note that at least some of the work you produce must be shared with others and that we strongly encourage you to share as much as possible with students in this course, current and future.

The course makes extensive use of social technologies such as blogs, wikis, bookmark sharing, file sharing, and discussion spaces to enable learning in a manner that reduces isolation and allows students to learn from one another.

Please be aware that this is primarily a course about programming for the Web. It is not an introduction to website design and simple page building which, though covered to some extent, are only introduced to the level needed to create effective programs for the Web.

Special Instructional Features

This course is strongly problem based, which means that although it follows a guided and highly scaffolded process, you are not told precisely what to do and how to do it, as might be the case in a typical textbook. The onus is on the student to seek out or follow links to web-based resources from which to learn

the detailed process of programming. This is, however, a supported process: advice is given regarding the overall structure of development and the process of learning; recommendations are given about appropriate resources; a rich social space is provided in which to discuss issues, ask questions, and receive answers; and feedback is provided along the way on work that is done. Where possible, students are encouraged to help and support one another: such support can be rewarding not just as a learning activity in itself but also as a means of showing that the learning outcomes have been achieved.

Challenge for credit

Overview

The challenge for credit process allows you to demonstrate that you have acquired a command of the general subject matter, knowledge, intellectual and/or other skills that would normally be found in a university-level course.

Full information about **challenge for credit** [↗](#) can be found in the Undergraduate Calendar.

Evaluation





The challenge assessment for COMP 266 involves the production of a portfolio of evidence, including a website, design documentation, design artifacts, and reflective commentary, mapped to intended course learning outcomes, demonstrating competence in the design and construction of web technologies including JavaScript, HTML and CSS, as well as an online oral examination based on the work submitted.

The evaluation will be based entirely on evidence presented in the portfolio and oral examination, mapped to the intended learning outcomes of the course. All components must be completed in order to pass the assessment but there is no explicit weighting given to any particular component: grades are given based on evidence provided of achievement of learning outcomes, not on specific deliverables.

To **receive credit** [↗](#) for the COMP 266 challenge registration, you must achieve a grade of at least **D (50 percent)** [📄](#) on the portfolio and oral examination.

[📄](#) **Challenge for credit course registration form**

Important links

- › [Academic advising](#) 
- › [Program planning](#) 
- › [Request assistance](#) 
- › [Support services](#) 

Athabasca University reserves the right to amend course outlines occasionally and without notice. Courses offered by other delivery methods may vary from their individualized study counterparts.

Opened in Revision 3, April 4, 2012

Updated January 20, 2025

View [previous revision](#) 
