

# CLIENT SIDE PROGRAMMING

Mis 370

COURSE SYLLABUS

SPRING 2018

## MEETING TIMES

TTh, 9:30 - 11:10

## LOCATION

Main Lab, Old Main

## PREREQUISITE

MIS 280 Introduction to Internet Programming

## CREDITS

4

## INSTRUCTOR

Lauren Williams, PhD

## EMAIL

llwilliams@mercyhurst.edu

## OFFICE

Old Main 404

## OFFICE HOURS

Mon: 9 - 10 and 1 - 2  
Tues: 11:30 - 12:30  
Wed: 11 - 11:45  
Thur: 8:30 - 9:15  
Fri: 9 - 11

## GRADING

30%: Midterm Exams  
50%: Assignments  
20%: Final Exam

D D+ C C+ B B+ A  
60 67 70 77 80 87 90

## COURSE DESCRIPTION

Client-side web programming, creating programs executed by the web browser to make dynamic web pages and sites. Separation of content, presentation, and behavior will be a major theme of the course. JavaScript will be the language of choice, and the first half of the course will be focused on its basics. Topics will include statements and commands, data types and variables, arrays, strings, functions, and programming logic. In the second half of the course, a JavaScript framework will be introduced, and the students will use it to make their webpages dynamic and interactive.

## COURSE OBJECTIVES

Upon completion of this course students should be able to:

- Understand the difference between the style, structure, and behavior of a web page, and know what is responsible for each of those areas.
- Write clear JavaScript, HTML, and CSS code that adheres to web standards and promotes readability.
- Understand the basics of JavaScript programming.
- Understand what an event is, and understand how to define user actions in relation to a document.
- Write efficient code that uses modern best practices to solve problems.
- Understand what an object is, and what it means for a programming language to be object oriented.
- Import JavaScript libraries and frameworks, and leverage them to speed up and ease development.
- Know what jQuery is and how it helps ease DOM manipulations.

## IMPORTANT DATES

January	16	First Class Meeting
	19	Last Day to Add/Drop
February	20	Exam I
March	5-11	Mid-Semester Break - No Class
	29	Easter Break - No Class
April	5	Exam II
	10	Advising Day - No Class
	13	Last Day to Declare Pass/Fail
	20	Last Day to Withdraw
May	3	Last Class Meeting
	10	Final Exam, 8 am

## COURSE STRUCTURE

Rather than adhere to a strict schedule of topics, we'll spend time as needed covering the essentials of client side programming.

The outline below contains a few topics that we will cover in class. More topics may be added later as time permits. This is not meant to be a schedule - the topics may not be covered in the order presented here.

- **JavaScript Basics**
  - Uses of JavaScript
  - Best practices in JavaScript and web development
  - Using the console and other tools
  - Adding JavaScript to a web page
  - Basic syntax
  - What JavaScript *cannot* do and things to avoid
  - Finding up to date resources
- **Objects and Functions**
  - Declaring and defining variables
  - Working with built-in objects: numbers, strings, arrays, date/time, and more
  - Defining and calling functions
  - Creating custom objects
  - Object methods and properties
- **Control Statements**
  - If, If-Then, and If-Else
  - For, While, and Do loops
  - Switch statements
  - Understanding control flow and interpreting code logic
- **Events**
  - Adding event listeners to objects on the page
  - Responding to events
  - Timers and Intervals
- **Interacting with the DOM**
  - Changing, adding, and removing HTML elements
  - Altering styles
  - Document and Window objects
- **Libraries and Frameworks**
  - Finding and choosing libraries
  - Reading library documentation
- **Additional Topics**
  - Working with SVG, Canvas, and other graphics
  - Creating, updating, and removing cookies
  - Creating and verifying forms
  - Using regular expressions

## REQUIRED AND SUGGESTED MATERIALS

### TEXTBOOK

There is no textbook required for this course. Rather than purchase a book that will be obsolete in a matter of months, you are encouraged to find answers to questions about syntax, best practices, and more from the wealth of reputable and updated sources on the web. Use caution: many JavaScript tutorials and sites are long out of date, and should not be used as a reference for modern web development. The sites below are highly recommended:

- <https://developer.mozilla.org/JavaScript>  
Mozilla Developer's Network site dedicated to all things JavaScript. Includes well written tutorials at a variety of levels, covering nearly every concept in JavaScript programming. Also includes detailed reference pages on functions, objects, statements, expressions, and more.
- <https://www.javascript.com/>  
Features a Learn JavaScript section with simple examples, a Resources section with links to recommended JavaScript courses around the web, and a News section with posts and updates from the community of JavaScript developers.
- <https://www.w3schools.com/js/>  
Includes some documentation and many examples of important JavaScript concepts. Allows you to experiment with code samples to better understand how it works. Some material may be outdated, so use with caution.

### COMPUTER ACCESS

A computer and internet access will be required to complete and submit assignments. The lab machines are available according to the schedules posted on the lab door if you do not own a computer. If you do own a laptop, you are welcome to use it in lieu of the lab machines during class. Any internet connected machine running Windows, Linux, or Mac OS is fine. Chromebooks are not recommended.

### SERVER ACCOUNT

All assignments and projects must be uploaded to your directory on our department's server. If you have taken MIS 280 on the Erie campus, you already have an account. If you do not have an account, or no longer remember your password, please inquire about getting an account within the first week of the semester.

### SOFTWARE

You will need at least two pieces of software: one to access your directory on the server, and another to write your code. While premium software is available to accomplish these tasks, there is an abundance of free software that can handle more than you will need for this course.

To access your server account and transfer files to and from your directory, you will need an SCP or FTP client. A few suggestions are below, but you are free to use any program you like.

- WinSCP  
<https://winscp.net/eng/download.php>  
This client is installed on the lab machines and is recommended for Windows users (not available for Mac or Linux). Free and open source.
- Filezilla Client  
<https://filezilla-project.org/>  
Can be installed on Windows, Mac, or Linux. Free and open source version is recommended. There is a Pro version that does not have any features we would use in this class.

To write and edit your code, you will only need a text editor, though you may prefer something more robust with helpful features like color highlighting, line numbering, code completion, etc. There are too many to list, but we will discuss recommendations in class.

## ASSIGNMENTS

Throughout the semester, you will be given brief assignments based on class topics. Some of these assignments will not be collected, but you are still expected to be familiar with the techniques necessary to complete them. If an assignment is collected for a grade, you will generally have one week to complete your assignment.

Projects will be based on JavaScript, not HTML or CSS. You will not be held responsible for reflexive design, the use of advanced frameworks, or styling options unless specified, though your assignment grades will reflect the overall effort put into all aspects of web development. Best practices for all languages used should be observed. Your work will be expected to follow standard guidelines for HTML, CSS, JavaScript, and any supporting libraries or frameworks. This includes proper indentation, variable declaration, commenting, and safe coding techniques. This will be an increasingly important consideration in your assignment grades throughout the semester.

Before beginning any work in this course, please review the collaboration guidelines.

## ASSIGNMENT REQUIREMENTS AND GRADING

All assignments and projects must be uploaded to your public html directory on the department server by the assignment due date. Projects submitted via email or other means will not be accepted. Using the server is recommended while developing your projects, as you are far less likely to lose your work. It is also the fastest way for me to assist with projects as you work.

A basis for each project's grade will be provided with the project description. You should take this criteria into account when developing your work, and note that the requirements may differ from one project to the next. It is expected that you follow the style and best practices guide for all assignments.

Part of most project grades (about 15%) will be based on the overall level of effort put into it. Generally, a grade of B+ or A indicates that you have gone beyond the minimum requirements and have produced truly outstanding work.

## LATE ASSIGNMENTS

All files related to your assignment must be located within your public directory by the time and date indicated on the assignment. If all or part of your assignment is not received by this deadline, you may still submit your work, with a penalty. Work received within 24 hours of the due time will receive an automatic 15% deduction. Work received 24-48 hours after the due time will receive a 30% deduction. Work received more than 48 hours after the deadline will not be accepted.

## EXAMS

There will be two midterm exams in addition to a cumulative final exam. There are no make up exams, though you may arrange to take an exam early if you know in advance you will not be able to attend class for a scheduled exam. If you miss an exam, that exam grade will be replaced by the final exam grade (your final exam would then be worth 35% of your final grade).

## EXAM DATES

- Thursday, March 1
- Tuesday, April 17
- Final Exam: Thursday, May 10, 8:00 - 10:00

## OTHER COURSE POLICIES

### ATTENDANCE AND WORK EXPECTATIONS

- Attendance is not part of your grade, but you are encouraged to attend class as often as possible. Many of the examples we go over in class on will be useful when working on homework.
- Most of the examples from our class meetings will be available on the course website, as are all assignments. If you miss class for any reason, it is your responsibility to check these resources for what you may have missed, and to ask a classmate if you missed any announcements in class. Please do not email with the question “what did I miss?” or worse, “did we do anything important today?”.
- Since this is a 4 credit course, you should expect to spend approximately 8-12 hours per week on homework and studying, in addition to time in class.

### ASSIGNMENTS AND HELP

- You will be largely responsible for the prerequisite material for this course. It is expected that you are familiar with working on the server, creating and publishing HTML files, and styling websites with CSS. If you do not have a strong background in internet programming, there are many online resources and textbooks that may be of assistance.
- I am happy to help with questions related to assignment requirements, but you will be largely on your own when working on “extra features” such as CSS or the use of other libraries.
- All assignments must be uploaded to the server in the required directory. I will not accept assignments as email attachments or via BlackBoard.
- If you have a question about your assignment and cannot come to office hours, please upload your work to the server and send me a link to what you have so far. Be specific in what you are asking; include any errors and/or a description of the problem.
- Programming questions may be answered with a suggestion for a helpful resource, such as a link to a page on the Mozilla Developer’s Network site with relevant information. Being able to quickly find answers to syntax questions is an important part of being a developer, and I will encourage you to use relevant and trustworthy sources.
- It is assumed that you have read the course Style and Best Practices Guide as well as the Collaboration Guidelines. Ignorance of a policy is not an excuse - please review these materials before beginning any work on assignments.

### EMAIL

- Email is best for simple questions. Please come to my office hours for in depth help with assignments or other concerns.
- For the best chance of a reply, use your Mercyhurst email account. You are also expected to check this account regularly.
- Many of the answers to “frequently asked questions” are available on the course website or in this syllabus. If you have a question about the schedule, grading, or other issues pertaining to the course, please check these sources first.
- DO NOT send email with homework questions the morning the homework is due. I will not have an opportunity to respond in time. If you have a last minute question, come to office hours for the best chance of getting help. I will expect that the majority of the work is complete in these cases.
- Please allow at least 24 hours for an email reply, particularly one that requires more than a simple answer. Occasionally, notes get buried, lost, or never make it to my inbox. If you do not hear back from me after 24 hours, please send a reminder.

## LEARNING DIFFERENCES

In keeping with college policy, any student with a disability who needs academic accommodations must call Learning Differences Program secretary at 824-3017, to arrange a confidential appointment with the director of the Learning Differences Program during the first week of classes.

## COLLABORATION POLICIES

Programmers use each other's code all the time - in fact, sharing code is how they learn from each other and develop their own skills. It's also how they construct new projects efficiently, without needlessly writing code that's been written many times before. However, professional coders also recognize that they cannot take credit for a project that is not almost entirely their own, and that they must give credit to those whose work they used or modified.

In this class, it will be tempting to use code you find online, in a book, or obtain from another student. Many assignments will be simple and based on common projects for JavaScript beginners. You will have no trouble finding a source with full solutions in many cases. However, keep in mind that you want to learn for yourself how to approach a coding project, how the code works, and how to correct issues. All code you submit for assignments must be your own.

Remember, you are not taking this course for the purpose of satisfying a degree requirement or maintaining your GPA. You are taking this course to develop and refine skills that you will be expected to have in your career. It is imperative that you follow the guidelines below to ensure your time, money, and effort in the course have been well spent.

## LIBRARIES AND FRAMEWORKS

For many assignments, you will not be required nor permitted to use JavaScript libraries or frameworks, including jQuery, React, and Angular. The focus of this course will be on "vanilla" JavaScript, as a strong familiarity with the underlying language is the only way to truly take advantage of the libraries.

Some assignments will require the use of specific libraries. In these cases, the libraries will be clearly stated in the assignment description. For other assignments, you will be permitted to use libraries of your choice (with some stated exceptions).

If the source of the library is not stated within the .js file, include a citation as a comment within the script tag. For example,

```
<script src="../../../js/math.js">//see mathjs.org</script>
```

## SHARING CODE WITH CLASSMATES

You will not be allowed to use or share code for an assignment or project with any other students in the class without explicit approval before submission. While you are encouraged to help each other, you should not be asking another student to do your work, nor doing the work of another student. The line between helping and cheating can sometimes be hard to see - please ask me when you're not sure.

For a few examples, you may assist each other with software, server issues, suggestions for references, or finding syntax or other minor errors in your code. It is not acceptable to ask to view another student's work to see how to complete a portion or all of an assignment, ask another student to write code for you, or offer to write code for another student.

## PENALTIES

If you are found to be

- using uncredited code from another student or source, without instructor permission to do so, or
- allowing another student to use your work, without instructor permission to do so, or
- using unauthorized notes, materials, another student, or any other form of cheating on an exam

you will receive a score of -20% on that assignment or a 0% on the exam (it would be better for your final grade to skip an assignment or exam than to cheat). In the case of an exam, an exam grade of 0% due to academic dishonesty will *not* be replaced by your final exam grade.

Subsequent offenses may result in an Academic Dishonesty report and an F for your overall course grade.