

# WEB1100

Basic front-end programming II

# Description

This course introduces advance concepts that will be used throughout the rest of the program. We delve in topics such as understanding a programs stack, understanding JavaScript single thread and how it differs from other programming languages, understanding recursion and why it's necessary, functional programming, and Object-Oriented programming and how it differs in the JavaScript language. We also accomplish creating a bootstrap portfolio and linking our projects. We also do parts of Wes Bos’ JavaScript 30 to introduce more advanced concepts and also Code schools Introduction to ES6 videos to help understand what is ECMAScript 5 and why they are implemented.

# Day 1

## Summary

Review of what was accomplished in WEB1010 and introduce JavaScript 30 and the ES6 videos. The first ES6 videos should be on using const and let. After the video and class discussion on it, students should not be allowed to use var as a variable declaration, only to use const and let. Then we will do the To-do-list application, however will be timed and meant to be done independently. Will go over answers later another day in class.

## Objectives

* Review the concepts from WEB1010 and what MUST BE KNOWN to be successful in this class and throughout the program.
* Assess the student's current abilities by doing a timed assessment like what they will be getting as a professional. They will have one hour to complete a to-do-list application with vanilla JS. They will not be required to turn it in after the hour time because most won't be able to finish but later in the class, it will be done together so that can be turned in. Turn assignment in under "To Do List" in Canvas.

## Deliverables

* ( <https://github.com/FVITech/Web1100--old-web-2010/tree/master/WEB1010-review-quiz> ) GitHub location of the WEB1010 review quiz
* ( <https://www.codeschool.com/courses/es2015-the-shape-of-javascript-to-come/videos> ) Videos from Code School or ES6

# Day 2

## Summary

We begin this day by understanding the JavaScript class. We both discuss and watch the videos, [*Asynchronous JavaScript*](https://www.youtube.com/watch?v=vMfg0xGjcOI) and explain how we have used it in previous apps. Also, [*What the heck is the Event Loop Anyway?*](https://www.youtube.com/watch?v=8aGhZQkoFbQ). Then we do the “Around the Board” app. If it's not finished today, we complete tomorrow.

## Objectives

* Understand how the JavaScript stack works. Being that it’s single threaded programming language and meant to handle asynchronous calls, students will learn how their browser handles dealing with these issues and the other programs involved. We will watch two videos explaining how it is done.
* Begin the Around the Board App. Once completed turn it in in Canvas under "Dice Roll Game"

## Deliverables

* ( <https://www.youtube.com/watch?v=vMfg0xGjcOI> ) Understanding Asynchronous JavaScript Video
* ( <https://www.youtube.com/watch?v=8aGhZQkoFbQ> ) What the heck is the Event Loop Anyway Video
* ( <https://github.com/JAnderson88/around_the_board_starter> ) GitHub location for the Around the Board starter app

# Day 3

## Summary

We begin on [Recursion](https://docs.google.com/presentation/d/1UYrhGqQdHogVWpzmpHy6C8ewe3EFrYYUuAZgwo_ct7Q/edit#slide=id.p). We define recursion as a function which calls itself and solve recursive definitions of factorial, Fibonacci, sum 1 to n, sum multiples of 3 or 5. We will showcase this explanation of recursion: <http://everything2.com/index.pl?node_id=477013>

## Objectives

* Review about what was learned about the JavaScript Call Stack, WEB API, Task Queue, and the Event Loop
* Recursion Presentation and the exercises that go along with it. They keep all these examples in a folder called Recursion for studying purposes.
* Students will be tasked with creating a function that removes all the Vowels from a given string as an assignment. They will do on their own but will be given the answer later. They will turn this assignment in under "String Mingling" in Canvas

## Deliverables

* ( <https://docs.google.com/presentation/d/1UYrhGqQdHogVWpzmpHy6C8ewe3EFrYYUuAZgwo_ct7Q/edit#slide=id.p>) Recursion presentation

# Day 4

## Summary

We go over the concept of Object-Oriented Programming and Inheritance, making sure to preference that in JavaScript it is not a traditional way of inheriting which programmers need to account for. We use the classes presentation to show the difference between traditional prototypal inheritance and the new ES6 class structure. We introduce the instructions for test number 1.

## Objectives

* Go over classes presentation and do the examples and exercises that come with it.
* Introduce the project that must be completed for Test 1

## Deliverables

* ( <https://docs.google.com/presentation/d/1ujsYUixURjSrr5ILTaRW78XkvmOBWFj8wN1eLvXFH9Q/edit?usp=sharing> ) Classes Presentation
* ( <https://docs.google.com/document/d/1xxgPCmu2-wl8lTNIfi8CmstQ4PlhN0KupabmhrpZYDM/edit?usp=sharing> ) WEB1100 Test 1 Instructions

# Day 5

## Summary

Continue to go over classes and how to properly structure classes and how to create functions and to use properly give permission to different properties of the class. All this will be needed to complete the test. We do the Number Boxes project that will show exactly how to structure your classes for the test. Then help set up the class for the test.

## Objectives

* Review the concepts of class, inheritance, and creating methods especially the constructor method.
* Begin and finish the Number/ Boxes Project. Create a folder and include this project and the classes exercises that you made the day before. Turn it in on Canvas under "Classes"

## Deliverables

Students should complete one to three problems from exercises file as well as the homework proposed on the homework file. Also, today the first project will be announced.

# Day 6

## Summary

The first half of the class is discussion about servers, and an implementation of a simple hello world server using Express. The second half is a talk about sessions and scaling. See [day7Lecture.md](https://github.com/FVITech/Web1100--old-web-2010/blob/master/day7Lecture.md) for the full lesson plan.

## Objectives

* Create a basic Express App using Node JS. Showing how to use functions that are in the express app and how to run a node server in your local environment.
* Talk about how to set up a session in express.
* Begin the Star Wars app project. Most likely won't finish but will go

## Deliverables

* ( <https://nodejs.org/en/download> ) Node JS download page
* ( <https://youtu.be/NALxjuyRXaE> ) Beginner Express JS Tutorial Part 1
* ( <https://youtu.be/TVrfhONc8Jw> ) Beginner Express JS Tutorial Part 2
* ( <https://github.com/FVITech/Web1100--old-web-2010/blob/master/day7Lecture.md>) Day7Lecture.md file GitHub Location

# Day 7

## Summary

Test 1

# Day 8

## Summary

Give a lecture on the [Introduction to Functional Programming](https://docs.google.com/presentation/d/1NTxza91hMToWKqRyKll48AuxwamEik_nj4W1Cj6d8hs/edit?usp=sharing). After the functional programming presentation is over, we will talk about the concepts covered in [chapter 5 of eloquent javascript: Abstractions](https://docs.google.com/presentation/d/1JxBhgt37xZgazl9gXMCQB9acQu7B-qvLdeO_SZlvyu4/edit#slide=id.p). After, do the functional programming practice for the upcoming quiz. Introduce Bootstrap library and show a few ways to implement them, then introduce the Bootstrap portfolio project.

## Objectives

* Functional Programming lecture on how to create abstraction within your own code and show some useful higher order functions that help keep your code clean and easier to use.
* Do the functional programming practice which focus on the map, filter, and reduce function. The practice will look very similar to the upcoming quiz.
* With extra time go over what the bootstrap library is and why professionals use it frequently. Then let the students know that they will be creating their own portfolios using a bootstrap template.

## Deliverables

* ( <https://docs.google.com/presentation/d/1NTxza91hMToWKqRyKll48AuxwamEik_nj4W1Cj6d8hs/edit> ) Functional Programming Presentation
* ( <https://docs.google.com/presentation/d/1JxBhgt37xZgazl9gXMCQB9acQu7B-qvLdeO_SZlvyu4/edit#slide=id.p> ) Higher Order Functions Presentation
* ( <https://docs.google.com/document/d/1yHEW0YmDnfZdKKxi0_smfhYd9YY7-KEy2WynDZNU78s/edit?usp=sharing> ) Map, Filter, Reduce Practice

# Day 9

## Summary

Students will create a portfolio website based on any of the [portfolio bootstrap themes](https://startbootstrap.com/template-categories/portfolios/), while using the [Bootstrap Documentation](http://getbootstrap.com/components/) to edit their pages. This is to teach them how to use Bootstrap and to ensure they don't forget the skills they learned in WEB1010.

## Objectives

* Have students pick a bootstrap theme if they haven't chosen one already.
* Students will work independently or in groups to develop their portfolio landing pages.

# Day 10

## Summary

Then we begin creating a Wikipedia visualizer using the Wikipedia API. Students should understand that for the wiki visualizer, they can build their api requests through a gui tool Wikipedia has. It's called the Wikipedia api sandbox. You also need to make sure that they remember to put callback=? As part of the query string, and when they build the api request, they need to use the generator called search, and up top they need to have the property set to extracts so they can get extracts of the page.

## Objectives

* Begin the Wikipedia App. This will be another use of an API and use of the ajax function with jQuery. More advanced students should be able to do this once they understand how the URL is implemented. Other students might need a walk through. Once finished together students should make at least 10 CSS changes to it, but to make the project their own. Once completed turn in project in Canvas under Wikipedia App Visualizer.

## Deliverables

* ( <https://github.com/FVITech/Web1100--old-web-2010/tree/master/classwork/wikipedia-app-starter> ) GitHub location for the Wikipedia App starter
* ( <https://en.wikipedia.org/wiki/Special:ApiSandbox> ) Wikipedia API Sandbox

# Day 11

## Summary

Do the Functional Programming quiz. Spend the rest of the day doing make up work and finish up the bootstrap portfolio.

## Objectives

* Functional Programming Quiz
* Continue with the bootstrap portfolio. Students should spend the rest of the day focusing on finishing their projects if they haven't and adding a link to the project in their portfolio site.

## Deliverables

* ( <https://docs.google.com/document/d/1lPwqPEJUuA03dOlOHiPvcArdgpB8MSABTIoNV46Mnrc/edit?usp=sharing> ) Functional Programming Quiz

# Day 2

## Summary

Final Exam