111
Project 3

Michael Hennessy, Computer & Information Science, UO
Project 3 requires you to write web programs that use concepts from ch. 2, 4, 5 of our textbook.

Since our midterm is also in Week 6, this project may be turned in by 1700-Mon-Feb-16 for full credit.

Students meeting the 1700-Fri-Feb-13 deadline will receive +5 pts XC for Speedy Deliveries.
Due Date: F 1700 2/13.

PROJECT LEARNING OUTCOMES

After completing this project, you will

• Understand the four steps of the WebDev Workflow as explained in the glossary.

• Understand Command-Line JavaScript and Client-Side JavaScript.

• Be able to use selection statements to solve programming problems

• Be able to use loops to solve programming problems

• Be able to write value-returning functions

PROJECT REQUIREMENTS

For best results, solve the following problems in order.

This project will be discussed in detail in class; that is the best place to ask questions about the project.


All of the following Command-Line JavaScript exercises. None of these functions will use prompt, and none of these functions will use console.log or alert.

You do not need to create a web page for this part of the project. Store the code for each exercise in functions.js in your p3 folder.

Download if.js to your 111/js/folder, and study it.

A) Write a write a value-returning function named tLightAdvisor that accepts a character as an argument and works like this:

```
tLightAdvisor('G') => 'Go'
```
tLightAdvisor('r') => 'Stop'
tLightAdvisor('Y') => 'Proceed with caution'
tLightAdvisor('O') => 'Signal broken!

B) write a function named maxOf2 that works like this:

maxOfTwo(2, 3) => 3
maxOfTwo(7, 4) => 7

C) write a function named isDigit that works like this:

isDigit('0') => true
isDigit('a') => false

D) write a function named IPA that verbalizes the first three letters of the alphabet using the International phonetic alphabet, and works like this:

IPA('A') => 'Alpha'
IPA('b') => 'Bravo'
IPA('e') => 'illegal input: e'


A) Command-Line JavaScript. Write a function named ordForm that accepts a positive integer and returns a string representing the ordinal form, like this:

ordForm(20) => '20th'
ordForm(1) => '1st'
ordForm(22) => '22nd'
ordForm(103) => '103rd'
ordForm(12) => '12th'

B) Create a web page named ordForm.html in your p3 folder, that allows the user to enter a positive integer. When the user clicks a button, the ordinal form of the number is displayed on a web page.

Since the web page is using a value-returning function to solve this problem, Follow the same three steps shown in pattern-2.js.

Client-Side Debugging Tip: If nothing happens when you click the button, always remember to open the DevTools console and check for error messages.

When your program is not working correctly: Your first stop is always the DevTools console.
How to Turn In your Project

All you Have to Do is Make Sure your web pages are uploaded to the server and tested on the server by the Due-Date.

When your web pages are on the server, they can be graded.

You do not have to submit this project in Blackboard, nor do you have to notify your instructor in any way (not even by Owl post).

Just make sure you complete the project by the Due-Date, and do not upload or edit the files after the due-date. If you change the web page files in any way after the due-date, this will change the time-stamp of the files on the server, and your project will be late (zero points).

Questions About this Project? Think, Piazza!

Do not send email to your instructors to ask questions about projects. Post your questions on Piazza, so all students in class can see the answer.

111 Help

See Office Hours and 111 Help Hours in Blackboard for times assistance is available in 013 Klamath.
HOW YOUR PROJECTS WILL BE GRADED

These checkpoints will help you get full credit on your projects.

- The files you upload to the server by the due-date are what will be graded, so be sure to test your web pages on the server to make sure they are correct.
- Your job: make sure the files are on the server on time, and that you have tested them to make sure they are correct.
- There are no second chances. We do not have the time or the resources to grade your work twice. Therefore make sure that what you upload to the server is correct. Test your web pages on the server after uploading them.
- Time-Stamps are Crucial. When you upload a file to the server, it is stamped with the exact time of the upload. This time-stamp must be no later than the project due-date. Your project is on-time only if the time-stamps show that it was uploaded to the server on time.
- Do not re-upload any of your project files after the due-date. If you do, this will change the time-stamp and your project will be late (0 pts).

- Do not use Sublime's Sync feature, as this will change the time-stamp on all your files on the server.
- Your 111 folder on the server must be .htaccess password-protected. If it is not, your project score will be zero (0). See your instructor or GTF for assistance if necessary.
- Know the 111 Late Policy.
How to Handle the 17:00 Friday Deadline

- Start working on your project early. Do not delay.
- Friday Office and Help hours are jammed, and may end before you get assistance. Plan on completing all your projects before the Friday deadline.
- Turn in what you have by the deadline-- partial credit is better than none.
- Piazza is good for answering questions. For debugging code you need F2F help, which is the gold standard.

- For Gold Standard Help:
  -- See Contacts in Blackboard.
  -- See 111 Help Hours in Blackboard.

[Instructors] are a Superstitious Sect, Great Keepers of Set Times and Places.

-- from Poor Richard's Almanac
The 111 WebDev Workflow

Here is the CIS 111 Web Development Workflow. Memorize these 4 steps.

1. **Edit.** Use the Sublime Text editor to create a web page (.html and .js files) on your computer.

2. **Preview.** Open the web page on your computer using Chrome. When it is perfect, and not before, go to the next step.

3. **Upload.** Move all project files (.html, .js, .png, and so on) to the server using an SFTP client (CyberDuck, Sublime, Aptana, ...). This is also known as Publishing the web page.

4. **Test.** Use Chrome to open your web page that is on the server. Do not use CyberDuck, do not use Aptana-- use Chrome. Make sure that this web page is correct, because that is what will be graded.

**Related Glossary Terms**

Drag related terms here