

# Lab 6 Instructions, Objects, Group C

## CS 133JS, Beginning Programming: JavaScript

### Overview

The objective of this lab is to give you practice using:

- Creating objects
- Working with object properties
- Working with object methods
- Using object constructors
- Working with complex object

### Part 1: Object Exercises

A web page, *ObjectExercises.html*, has been written for you that contains code to call functions and use code that you will write in a file named *ObjectExercises.js*. The instructions for writing your functions and the code to test your functions are in *ObjectExercises.html*, but all the code you write will go in *ObjectExercises.js*.

### Part 2: Web App

#### Web App for Group C – ToDo List

This web app displays a list of tasks and allows a user to add tasks, set priorities and mark tasks as finished.

This is a new version of the ToDo List app you made previously using arrays. This one will use objects.

Implementation:

1. Declare a global one-dimensional array named *tasks*.
2. Define an object constructor named *Task* with the following properties:
  - a. *description* – the task description
  - b. *priority* – a number
  - c. *complete* – Boolean, *true* if the task is complete
3. Write three functions:
  - a. *addTask*  
This function will add an element containing a *Task* object to the *tasks* array.
    - Has two parameters: a string describing a task, and a priority number.
    - Returns nothing.

## Lab 6 Instructions, Objects, Group C

CS 133JS, Beginning Programming: JavaScript

b. *removeTask*

This function removes a task from the Task List by removing the specified element from the *tasks* array. The function:

- Has one parameter: an array index for the *tasks* array.
- Returns *true* if the index is valid.  
(Valid means the index is greater than zero and less than the length of the array.)
- Hint: use the *splice* method to remove the element for this task from the array.

c. *changeDone*

The purpose of function is to cause the word “Done” to appear or disappear next to a task’s check box depending on whether the box is checked or un-checked.

The function does this by setting the *complete* property of the specified *Task* object to either true or false. The function:

- Has two parameters: an array index number and a Boolean completion value.
- Returns *true* if the index was valid.

This is a screenshot of a working Grade Book web app after two additional tasks were added to the list and two check-boxes were checked.

## ToDo List

Task			Priority	Done
1	Brush the unicorns	Delete	3	<input type="checkbox"/>
2	Feed the dragons	Delete	1	<input checked="" type="checkbox"/> Done
3	Bathe the lions	Delete	2	<input type="checkbox"/>
4	Walk the llamas	Delete	4	<input checked="" type="checkbox"/> Done
5	Clean the griffon's pen	Delete	5	<input type="checkbox"/>
Clean the griffon's pen		5	Add Task	

## Lab 6 Instructions, Objects, Group C

CS 133JS, Beginning Programming: JavaScript

### Submitting your lab work on Moodle

#### Beta Version

Post the following in the *Lab Beta forum*:

1. The web pages you created for part 2.  
(Zip the files for your web pages and attach them to the post.)
2. A code review of your lab partner's web page for part 2.  
(Review the part 2 web apps for one of your lab partners using the Code Review Form provided.)

#### Code Review

1. Submit a copy of the code review above to the *Lab Code Review assignment*.

#### Production Version

You may revise your beta version before submitting the production version. On the code review form you received from your lab partner, complete the "Production" column to show what you did or did not revise.

Upload the following 5 files to the *Lab Production Version* assignment:

1. Two for files (.html and .js) for part 1.
2. Two files (.html and .js) for part 2.
3. The code review from your lab partner with the "Prod" column filled in by you.