



GYMNASIUM

JAVASCRIPT FOUNDATIONS

Lesson 6 Assignment

Advanced Concepts

CORE CONCEPTS

1. Variables can have two possible scopes: local and global. Local variables exist only within a function. Global variables exist on the window object and are accessible by any script on that page.
2. Global variables can be useful for creating libraries or code that you want to share across scripts. But, they can be dangerous when you get multiple developers working on the same site. Limit your global variables to only what is absolutely necessary, and give them unique names.
3. Software design patterns define common, reusable solutions to problems repeatedly encountered in programming. They are guidelines to help you come up with a workable solution to a specific situation.
4. The Model-View-Controller pattern is an overall pattern for architecting an application. The model holds the application's data and is often an observable subject. The view observes changes on the model and renders that model's data. It also picks up user input and forwards it to the controller. The controller deals with the user input by updating the model in some way. Many variations and interpretations of MVC exist.
5. There are many application frameworks available for JavaScript. We looked briefly at Ember JS and Angular JS, and more in depth at Backbone JS. While these all accomplish the same basic purpose of helping you to better organize complex applications, there are many differences between them.
6. Backbone JS emphasizes views, which are tied to HTML elements, and models, which usually load their data from a web service. It's up to you to create a controller or otherwise decide how your models and views will be connected.

ASSIGNMENT

1. Quiz
2. Load a few sites of your choice. Open the console and type "window". Expand the results to see all the properties on the window object. Scroll down and see if you can identify any global variables that developers may have put there. You can double click on a property's value in the console and edit it. Try changing the value of a global variable and see if it has any adverse reactions on the site.
3. Research more about design patterns. I've included a list of resources below. Find a simple pattern that interests you, and create an implementation of it in JavaScript.
4. In the MVC application we created in the third video of this lesson, I mentioned that there were various ways you could have coded the app a bit differently; such as having the view get the value of the text input, and pass that directly to the controller. Experiment with changing the code in some of these ways, or in any other way that you think would improve the application.
5. In the movie finder application done in Backbone, the views are communicating directly with the models. Try making a controller object, or multiple controllers. The views would pass any input to the controller, and the controller would take this input and use it to fetch the data from the model.
6. If you are really ambitious, study up on Ember or Angular, or even some other MV* framework and see if you can re-implement the movie finder using one of those.

RESOURCES

- Books on software design patterns:

Free:

<http://addyosmani.com/resources/essentialjsdesignpatterns/book/>

Other:

[JavaScript Patterns](#) by Stoyan Stefanov

[Pro JavaScript Design Patterns: The Essentials of Object-Oriented JavaScript Programming](#) by Dustin Diaz and Ross Harmes

[Head First Design Patterns](#) by Eric Freeman, Bert Bates, Kathy Sierra and Elisabeth Robson

[Design Patterns: Elements of Reusable Object-Oriented Software](#) by Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides

Wikipedia:

http://en.wikipedia.org/wiki/Software_design_pattern

Ember JS:

<http://emberjs.com/>

<http://emberjs.com/ember-users/>

Angular JS:

<https://angularjs.org/>

<https://builtwith.angularjs.org/>

Backbone JS:

<http://backbonejs.org/>

<https://github.com/jashkenas/backbone/wiki/Projects-and-Companies-using-Backbone>