"THE GREAT LEGACY CODE CRISIS OF 2016"

**LEGACY CODE
IS INEVITABLE**

Agenda

- Last Week
- RESTful Apps
- Assignment #2
- Selecting Groups
- Inheriting Code
- In-Class Exercise
- Reading Week!

What did we cover last week?

What was the significance?

Our List of Pups

Name: Franky

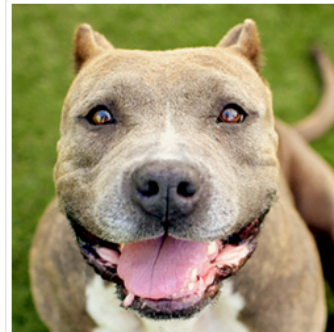Owner: The Hoff

Details

Name: Milo

Owner: Rob

Details

Name: Isis

Owner: Pam Anderson

Details

Name: Billie

Owner: Annie Jones

Details

Completion of Your To Do List App

You can now make a RESTful app using Node.js!

RESTful App

REST systems aim for fast performance, reliability, and the ability to grow, by re-using components that can be managed and updated without affecting the system as a whole, even while it is running.

The four HTTP methods common to a REST based architecture:

- GET - This is used to provide a read only access to a resource.

- POST – This is used to update or create a new resource.

- PUT – This is used to create a new resource.

- DELETE – This is used to remove a resource.

Assignment #2

As discussed in class, JavaScript is an evolving language. As you start looking for a job, many employers will list libraries or processes that you may be unfamiliar with. The objective of assignment #2 is to create a resource of information about various JS libraries. Working in your assigned groups of three, you will create at 30-40 minute workshop for your classmates (including me) that will help them understand what your selected topic is, how it is useful, and provide us with code & documentation so we can get up and running with the selected framework/process.

See BlackBoard for more details.
DUE: MARCH 17, 2017 or MARCH 24, 2017

Selecting Groups for Assignment #2

We will do a random draw…

Inheriting Code – A Learning Experience?

It is inevitable that you will inherit code. This can be challenging, but a very good way to learn.

In-Class Challenge (Work in Your Groups)

Working in your groups, each member should open the sample code. You can debug together, but I want each one of you to have a copy of the fixed code.

1. Run MongoDB in a terminal window. In another terminal window run "wedding_sample_data.js" in node to load the content into your database.

2. Run the app. There numerous bugs that are preventing the app from working properly. Squash those bugs to get the app working.

3. Once the app is "working", you need to adjust the code so it functions properly. A list (in order of importance) is on the next page. Work through each of these problems to get the site working to the best of your ability.

This challenge forces you to use what we have learned in class and debug some really ugly looking code.

In-Class Challenge (Work in Your Groups)

Here is the functionality requested. Ordered from most important to least important.

1. Make sure a user can add and delete and event from the DB.

2. When a user clicks to delete an event, prompt them with a confirm message.

3. Sort the list of events by days. Events that are closest should be listed first.

4. Add the ability to edit an event's details.

5. The input date only works in Chrome…fix this!

6. Remove the need to a user to perform a search after they add, update or delete data.

Submit what you have completed to BlackBoard at 5:00. Anything submitted after, will not be accepted!