Module 3 Day 10

MVC Review

What makes an application?

- Program Data
  - ✓ Variables & .NET Data Types
  - ✓ Arrays
  - ✓ More Collections (list, dictionary, stack, queue)
  - ✓ Classes and objects (OOP)

- Program Logic
  - ✓ Statements and expressions
  - ✓ Conditional logic (if)
  - ✓ Repeating logic (for, foreach, do, while)
  - ✓ Methods (functions / procedures)
  - ✓ Classes and objects (OOP)
  - ❖ Frameworks (MVC)

- Input / Output
  - User
    - ✓ Console read / write
    - ✓ HTML / CSS
    - ❑ Front-end frameworks (HTML / CSS / JavaScript)
  - Storage
    - ✓ File I/O
    - ✓ Relational database
    - ❑ APIs

Next Tuesday – Another Review Day

- Class Lightning talks
- ~15 minutes, hard stop at 30
- Share something
  - Something you worked on and created
  - Something you are planning or designing
  - Some technology you investigated and want to try
    - Preferably, MVC-related, but whatever…
- It does not have to be perfect!
  - After all, look at what I share with you every day ☺

TempData

- Similar to Session data, but stores data for this and next request
- Access it like you access ViewData
- Store:

    TempData["myKey"] = "myValue";

- Retrieve:
    string theData = (string)TempData["myKey"];
- Often used in conjunction with Redirects to "pass" data
- Example: Add City

Class Exercise

Question	Who
Describe how the recipe is saved in the DAO's. Can a partial save occur?	
Describe the flow of Add Recipe. How is model data saved and stored?	
Describe how and why the model is partially validated for each Add Recipe page	
Show where a partial view is used	
Describe how and where Session data is used	
Describe how DAOs are used to read data. Note reading ONE recipe vs MULTIPLE	
Describe where the Post-Redirect-Get pattern is used	
Describe how the Add Ingredient step stays on same page or moves to next	
Describe how I used inheritance in the controllers	
Show how TempData is used and why ViewData would not work	
Show where and how Select Lists are used for html dropdowns	