1. D10
2. D11
3. D12
   1. MVC
      1. Controller
         1. Practice using the different methods in the Controller class
         2. Practice using the different methods in the Text.Encodings class. How do these help Controller class functionality?
         3. Methods in a Controller require no parentheses. Why is this?
         4. Investigate the Intellisense for Controllers (and ASP.Net MVC) in general.
         5. Every **public** method in a Controller can be called as an **HTTP endpoint**. What is an HTTP endpoint, and what why is this useful for Controllers?
            1. An HTTP endpoint is a targetable URL *in the application*, meaning that the methods can be “invoked” in the URL bar.
            2. Endpoint = “simply returns the data to be written directly to the body of the response”/”the end of a communication channel”
            3. “Similar to doing @ResponseBody in Controller”
            4. Actuator endpoints =

Str

Endpoints are meant to perform the highly specific task of printing your Object(Json) on HTTP which is exactly what you want to do here.

To separate monitor-n-manage code from your application-specific code.

To keep things cleaner and cohesive

* + 1. Authentication type
    2. Logging
    3. Action methods
    4. Complete MVC Tutorial
    5. Application styling in MVC
       1. Bootstrap
    6. Translate the Program.cs file for a friend
    7. Translate the Startup.cs file for a friend
    8. Translate the \_Layoutcs.html for a friend
       1. How do links work within ASP.NET?
    9. Rebuilding
    10. How does localhost server work?
        1. Can be accessed without internet – basically the computer talking to itself
        2. https://www.geeksforgeeks.org/what-is-local-host/
    11. Html helpers vs asp helpers
    12. Code first vs Database first vs Model first
    13. What is does a pure MVC structure look like functionally?
    14. Razor syntax vs Razor pages (view)

1. D13
2. D14