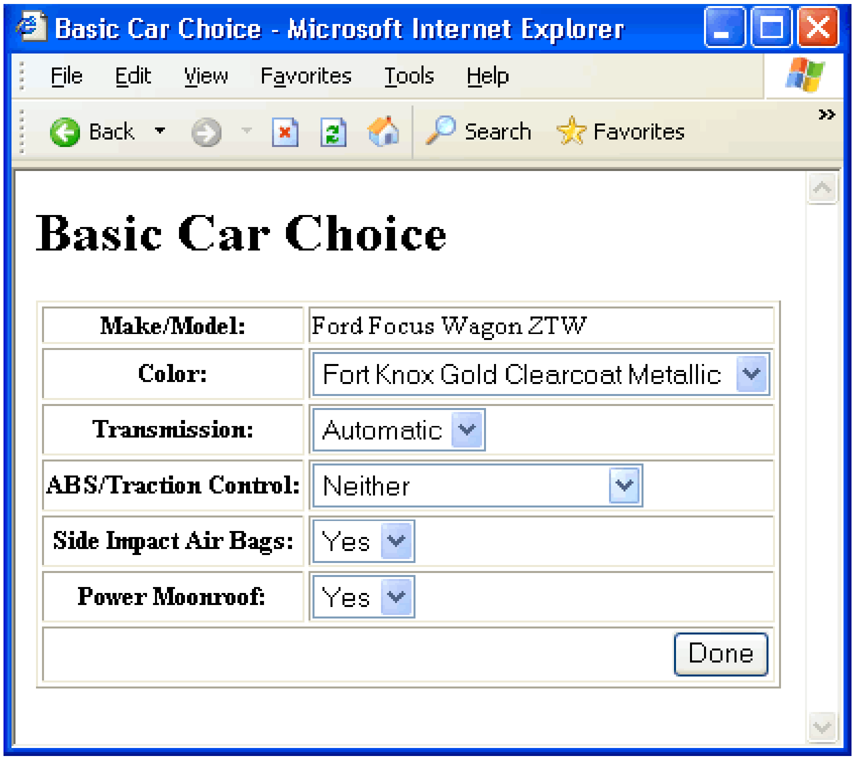
**Project – Unit 5**

We now have a client-server system where the end users, mostly non-programmers, can easily add new car models to the system with over writing each other's data entry.

We now need to extend this system so it can be used over the web. You will need to extend the functionality of Client (created in Unit 4) with the following features:

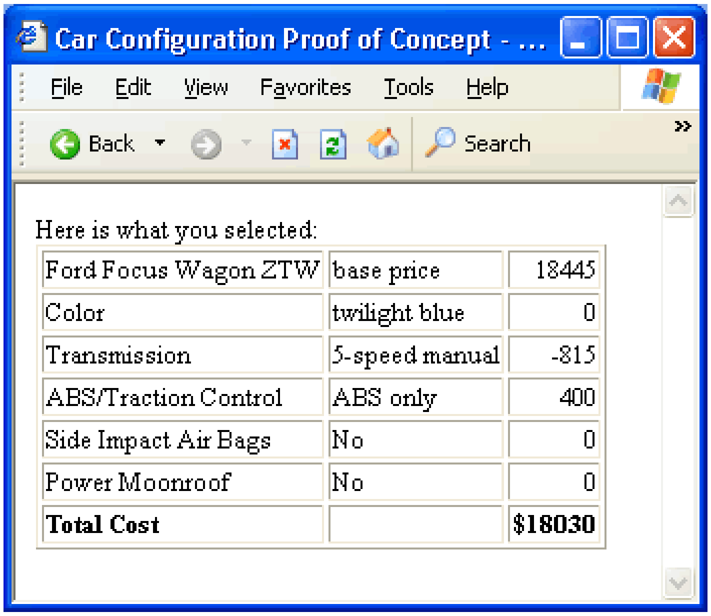
1. Show list of available models in a drop down box.
2. Upon selection of a Make/Model, the OptionSet and all related values load as shown below:



# Basic Car Choice

| | |
|---|---|
| **Make/Model:** | Ford Focus Wagon ZTW |
| **Color:** | Fort Knox Gold Clearcoat Metallic |
| **Transmission:** | Automatic |
| **ABS/Traction Control:** | Neither |
| **Side Impact Air Bags:** | Yes |
| **Power Moonroof:** | Yes |

3. User selection option and click on Done.
4. User is redirected to a page that displays selected choices and total price for a vehicle.



Here is what you selected:

| | | |
|---|---|---|
| Ford Focus Wagon ZTW | base price | 18445 |
| Color | twilight blue | 0 |
| Transmission | 5-speed manual | -815 |
| ABS/Traction Control | ABS only | 400 |
| Side Impact Air Bags | No | 0 |
| Power Moonroof | No | 0 |
| **Total Cost** | | **$18030** |

Use these screen shots as a design reference on which to model the interface of your own page.

Our first goal here is to develop the layout of the forms page, as well as to set up choices for users to select. Then we need to update the page to reflect choices made on the forms page.

Also, as you begin development, be sure to use the same configuration options that you used in previous unit.  Specifically, I'm referring to the choices regarding the moonroof, color, transmission, brakes, and airbags.

Technical Concepts Applied:

You have to design this unit with following technical requirements in mind:

1. Create a Servlet that interacts with Client created in Unit 4 to get the list of available models.
2. Create another Servlets that can interact with Client created in Unit 4 to get the data for the list of available OptionSets
3. Create a JSP that show the OptionSets and prints the selected choices with total vehicle cost.

Please keep in mind that data displayed is dynamic (i.e. read from LinkedHashMap from the Server created in Unit 4).

Grading your Submission

1. Program Specification/Correctness (25 points)
    a. No errors, program always works correctly and meets the specification(s).
    b. The code could be reused as a whole or each routine could be reused.
    c. Design is reusable and extensible.
    d. Interfaces and/or abstract classes are applied between Servlets and Client Server interaction.
    e. Code is adequately tested and test runs are shown for both Unit 5.
2. Readibility(5 points)
    a. No errors, code is clean, understandable, and well-organized.
    b. Code has been packaged and authored based on Java Coding Standards.
3. Documentation(5 points)
    a. The documentation is well written and clearly explains what the code is accomplishing and how.
    b. Detailed class diagram is provided.
4. Code Efficiency(5 points)
    a. No errors, code uses the best approach in every case. The code is extremely efficient without sacrificing readability and understanding.