**Assignment**

**Garage Database with JSF front-end**

Contents

[Introduction 3](#_Toc467092235)

[Marks 4](#_Toc467092236)

[Submission of the Project 4](#_Toc467092237)

[Overview of Web App 5](#_Toc467092238)

[Main Page 5](#_Toc467092239)

[Manage Manufacturer Page 6](#_Toc467092240)

[Manage Models Page 7](#_Toc467092241)

[Manage Vehicles Page 8](#_Toc467092242)

[Full Vehicle Details Page 9](#_Toc467092243)

[Find All Vehicles Page 10](#_Toc467092244)

[Add Manufacturer Page 13](#_Toc467092245)

[Add Model Page 14](#_Toc467092246)

[Add Vehicle Page 15](#_Toc467092247)

[Update Manufacturer Page 16](#_Toc467092248)

[Delete Manufacturer Link 17](#_Toc467092249)

[Demonstration of Application 17](#_Toc467092250)

### **Introduction**

Write a JSF/JDBC dynamic web application that queries and updates a MySQL database.

The database is called *garage* and can be downloaded from the *Project* section of Moodle for this module (*GarageDB.sql*).

The *garage* database consists of 3 tables:

|  |  |
| --- | --- |
| **Manufacturer** | |
| **Column Name** | **Details** |
| manu\_code | A code representing the car manufacturer e.g. TOY, FOR etc. |
| manu\_name | The manufacturer name e.g. Toyota, Ford etc. |
| manu\_details | Details about the manufacturer e.g. company history. |

|  |  |
| --- | --- |
| **Model** | |
| **Column Name** | **Details** |
| manu\_code | A code representing the car manufacturer e.g. TOY, FOR etc. |
| model\_code | A code representing the car model e.g. ESC, FIE, COR etc. |
| model\_name | The model name e.g. Escape, Fiesta, Corolla etc. |
| model\_desc | Details about the model e.g. type of vehicle, when it was introduced etc. |

|  |  |
| --- | --- |
| **Vehicle** | |
| **Column Name** | **Details** |
| Reg | The registration of the vehicle e.g. 2010-G-13345 |
| manu\_code | A code representing the car manufacturer e.g. TOY, FOR etc. |
| model\_code | A code representing the car model e.g. ESC, FIE, COR etc. |
| Mileage | The mileage on the car |
| Price | The price of the car |
| Colour | The colour of the car |
| Fuel | The fuel used by the car either *petrol* or *diesel* |

Full details of the Primary and Foreign Keys as well as datatypes can be found in the database itself.

### **Marks**

This assignment is worth 50% of the marks for the module.

### **Submission of the Project**

The zipped Eclipse project (named GXXXXXXXX.zip where GXXXXXXXX is your student number) should be uploaded to the *Project* section of Moodle no later than **Sunday December 11th 2016 at 9:00pm.**

### **Overview of Web App**

The web app should consist of the following pages:

#### Main Page

The *Main* page simply consists of 4 links:

* One to the *Manage Manufacturers* page
* One to the *Manage Models* page
* One to the *Manage Vehicles* page
* One to the *Find Cars* page

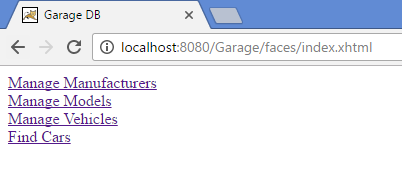


Figure 1 Main Page

#### Manage Manufacturer Page

The *Manage Manufacturer* page:

* Shows details of all manufacturers
* Has an *Add Manufacturer* button
* Has *Update* and *Delete* actions for each manufacturer
* Has a link back to the *Main* page

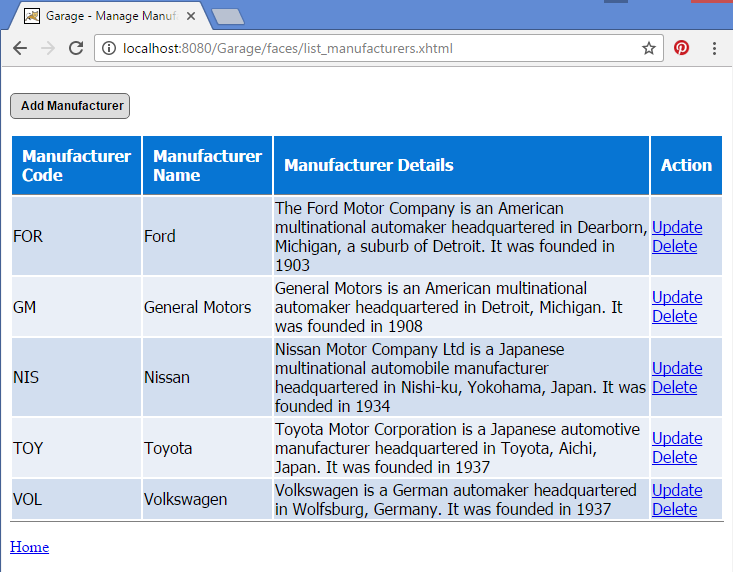
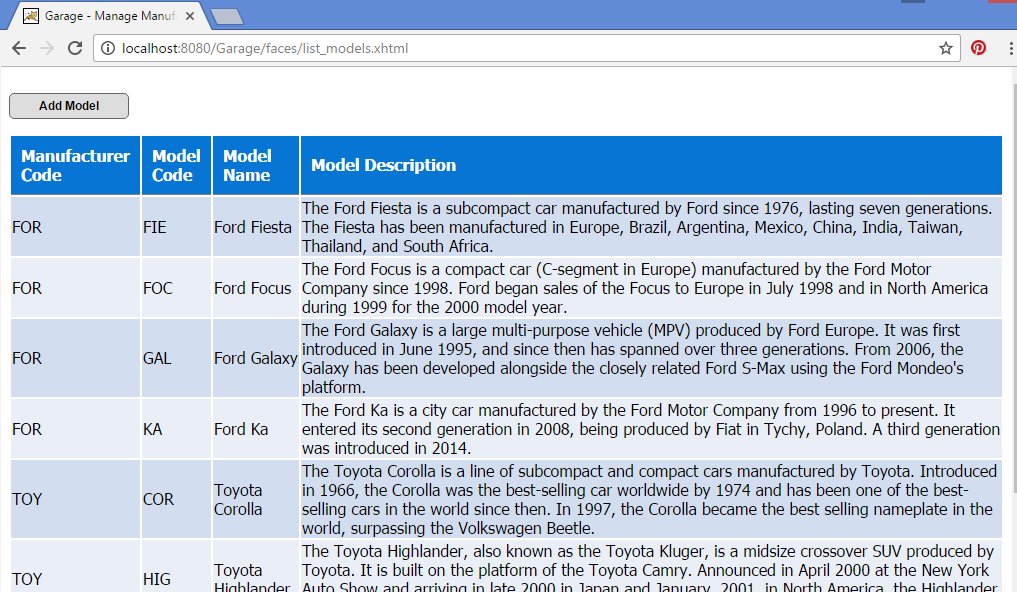


Figure 2 Manage Manufacturers Page

#### Manage Models Page

The *Manage Models* page:

* Shows details of all models
* Has an *Add Model* button
* Has a link back to the *Main* page

 Figure 3 Manage Models Page

#### Manage Vehicles Page

The *Manage Vehicles* page:

* Shows details of all vehicles
* Has an *Add Vehicle* button
* Has an *All Details* action for each vehicle
* Has a link back to the *Main* page

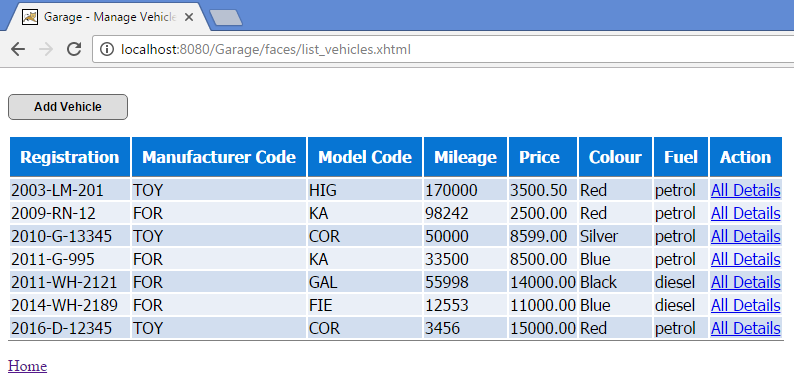


Figure 4 Manage Vehicles Page

#### Full Vehicle Details Page

The Full Vehicle Details page:

* Shows all details of the vehicle plus the manufacturer and model details as shown.
* Has a link back to the *Main* page
* Has a link back to the *Vehicles* page



Figure 5 Full Vehicle Details Page

#### Find All Vehicles Page

The *Find All Vehicles Page* page:

* Has a link back to the *Main* page
* Allows the user to search for cars on the following criteria:
  + Price
  + Colour
  + Fuel

*Fuel* will always be either *Petrol* or *Diesel*.

*Price* could be *Less Than, Greater Than* or *Equal To* a given value.

*Colour* could be a given colour.

If there are no values entered for *Price* or *Colour*, these parameters are not searched on.

Figure 6 and Figure 7 show the results when only the *Fuel* parameter is specified.

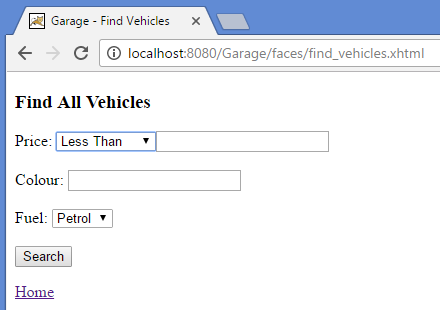


Figure 6 Find All Vehicles Page - Search only on Fuel

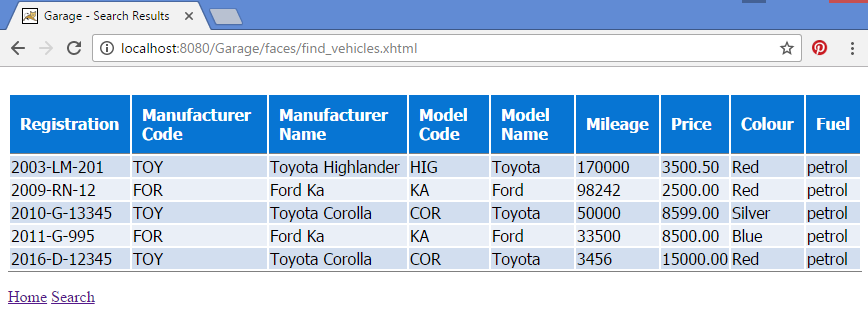


Figure 7 Find All Vehicles Results Page - Search only on Fuel

Figure 8 and Figure 9 show the results when only the *Price (Less Than)* and the *Fuel* parameters are specified.

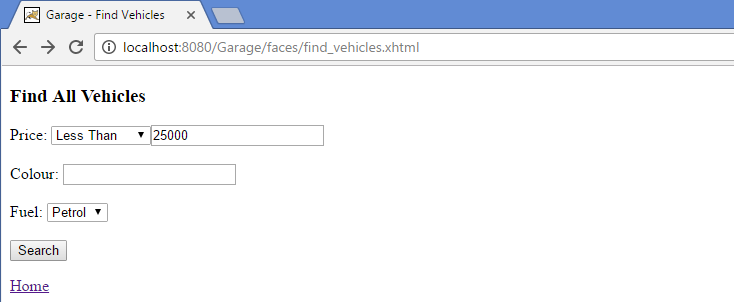


Figure 8 Find All Vehicles Page - Search on Price and Fuel

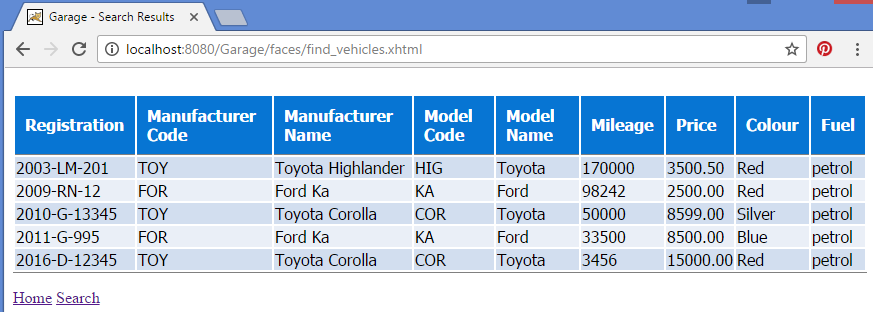


Figure 9 Find All Vehicles Results Page - Search on Price and Fuel

Figure 10 and Figure 11 show the results when the *Price (Less Than)*, *Colour* and *Fuel* parameters are specified.

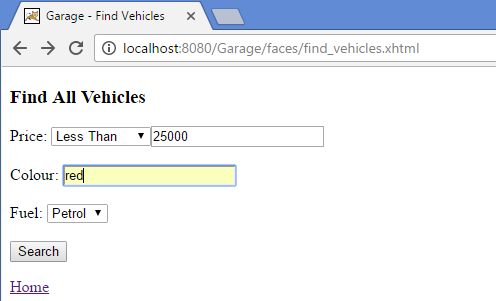


Figure 10 Find All Vehicles Page - Search Price, Colour and Fuel

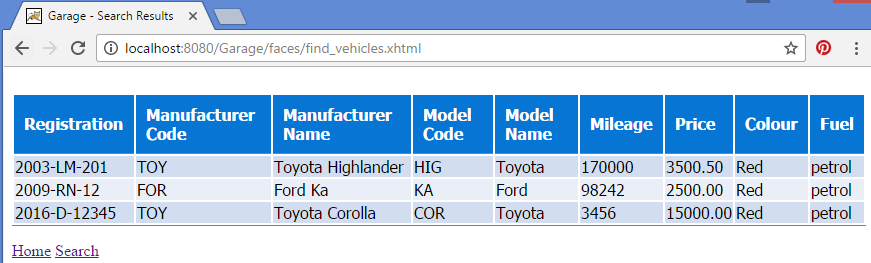


Figure 11 Find All Vehicles Results Page - Search on Price, Colour and Fuel

#### Add Manufacturer Page

The *Add Manufacturer Page* allows the user to enter a Manufacturer Code, Name and Details.

If an existing manufacturer code is entered an error should be displayed.

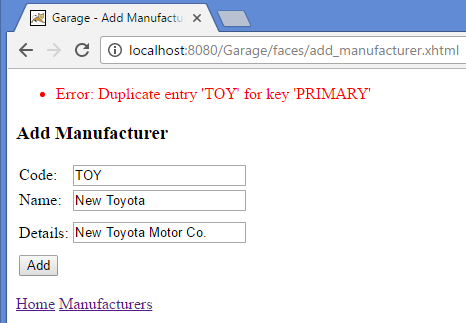


Figure 12 Manufacturer Code already exists

If a manufacturer code or name is not specified an error is also shown:

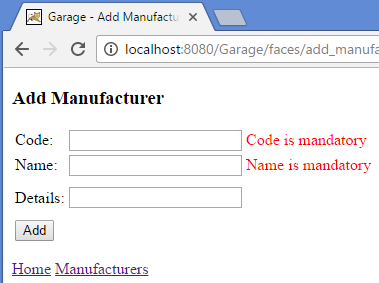


Figure 13 No Manufacturer Code or Name

When a manufacturer has been added successfully, the user is returned to the *Manage Manufacturers Page.*

#### Add Model Page

The *Add Model Page* allows the user to enter a Manufacturer Code, Model Code, Name and Description.

If a Manufacturer Code, Model Code or Model Name are not specified error messages are shown:

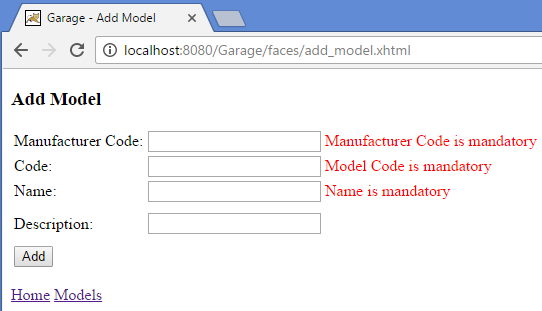


Figure 14 Manufacturer Code, Model Code, Model Name missing

If the user specified a non-existing Manufacturer Code an error message is shown.

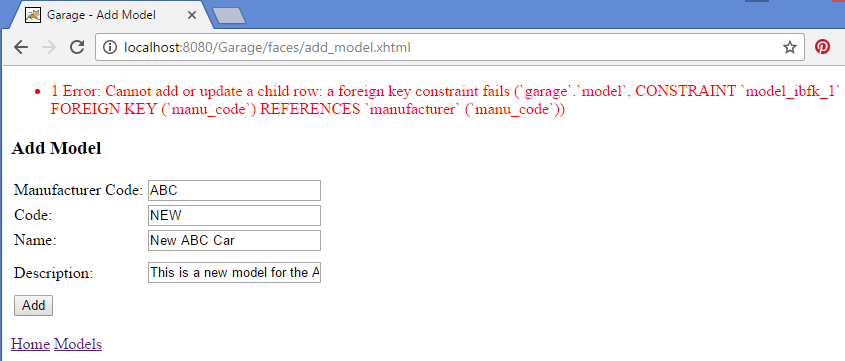


Figure 15 Manufacturer Code does not exist

When a model has been added successfully, the user is returned to the *Manage Models Page.*

#### Add Vehicle Page

The *Add Vehicle Page* allows the user to enter a Car Registration, Manufacturer Code, Model Code, Mileage, Price, Colour and Fuel for a car.

If the Registration, Manufacturer Code, Model Code or Price are not specified error messages are shown:

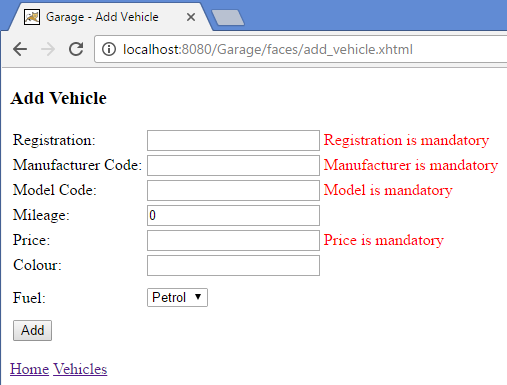


Figure 16 Registration, Manufacturer Code, Model Code, Price missing

If a correct Manufacturer Code, Model Code combination is not specified, or an invalid Manufacturer Code and/or Model Code is specified, an error message is shown:

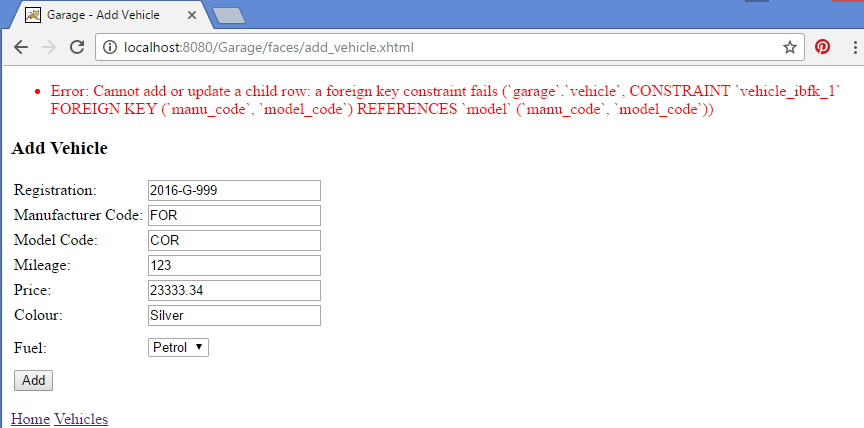


Figure 17 Invalid Manufacturer Code, Model Code combination

#### Update Manufacturer Page

The Manufacturer code cannot be updated, only the Name and Details:

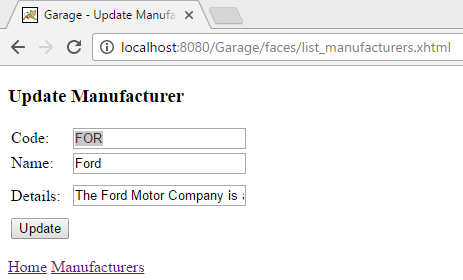


Figure 18 Update Manufacturer

#### Delete Manufacturer Link

When the user clicks on the *Delete* Manufacturer link he/she is presented with a confirmation dialog box:

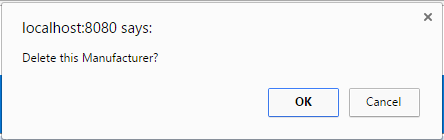


Figure 19 Delete Manufacturer Dialog

If the user clicks OK, the manufacturer is deleted – unless it is referenced by a Foreign Key in another table – in which case an error is shown:

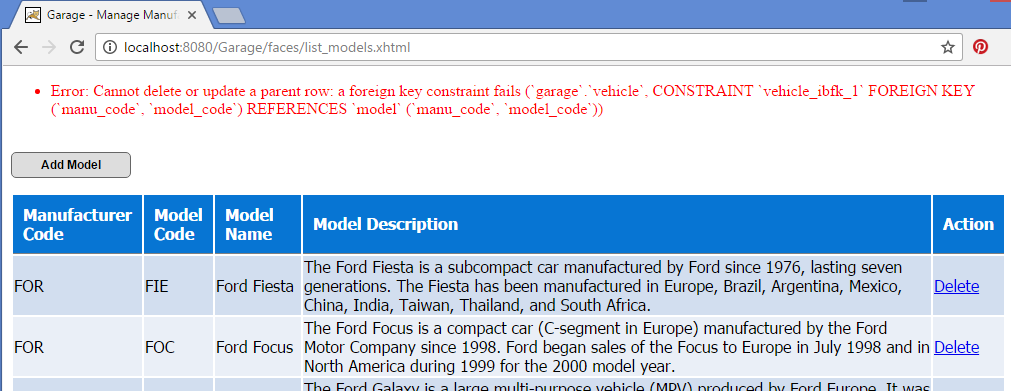


Figure 20 Cannot Delete Manufacturer

### **Demonstration of Application**

Download *Data Centric RAD Project Demo.webm* from the *Project* section of Moodle and open it in a web browser to see a working version of how the finished assignment should look.