**Web App Exercise Solutions**

List box screen width

Graphical user interface, text, application

Description automatically generated

Show all data on load

Graphical user interface, text, application, chat or text message

Description automatically generated

Paging

**Graphical user interface, text, application

Description automatically generated**

**Text

Description automatically generated**

*This has been updated later on to put the logic into the controller.*

Column sorting

Graphical user interface, text, application

Description automatically generated

*This has been updated later on but mostly still the same.*

Colour adding

I originally intended for a Color object to be used. Then when putting that into the UI, the Colorbox manages a string representation. That could be used further but I’ve opted for a simple string readable representation. In particular, this is arguably required for the sorting as we don’t want a sort to be on a text form containing rgb values. For simplicity, I’ve omitted the technical form of the colour but left the encapsulating object in so it could be put back in if needed and/or there is a hook to do so.

Conditional text colouring

Graphical user interface, text

Description automatically generated

Bug with detail still displayed

Handled with listening on selection events for the Listbox. When there is a selection, make sure the detail is visible; and where there isn’t, make sure it’s hidden. Searching also refreshes the selection.

I also decided to retain the selection such that if the user selects a car, then searches, the car remains selected if it is in the search results. Otherwise, it hides the detail.

Add new car

New page created. As above with colour, used simplified representation – could be improved if something more was needed.

From the outset, it seemed that the logic/behaviour for adding was going to be similar to that of editing, so I wrote the code with editing in mind.

Delete car

Controlled whether it is disabled or not in the controller based on selection change. Same logic for edit selection.

Persistence

Decided to persist the paging mode (using the check box) and the sorting. These both use cookies, the former being a simple true or false stored. The latter is trickier and uses a cookie value of column name and sort direction. Only one type of sorting can be applied at any given time (as this seemed to be the default behaviour). This could be improved to allow multiple sorting (by column, then by column, etc).

Testing

Developed along the way.

New Feature

The main additional feature is providing an edit function. I chose to do this because without it, it leaves the users with more work to do – delete the existing car and recreate it in order to change it. This could be a problem too if the id is used for anything in future (as the original would be lost). Technically, it’s functionally similar to the add so it’s somewhat of a quick win.