5.2C: Create a Form web page from requirements

Tasks

The **dKin Caps** business is taking off and you are looking at having an online website where your valued clients can purchase **dKin Caps** with various and colourful designs.
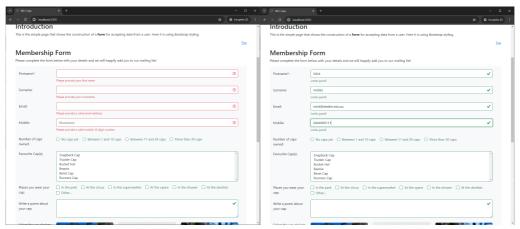
To gauge the interest of cap designs, you have been asked to improve and extend the *Membership Form* in two ways: adding improved form validation; and to show samples cap designs.

Please use your sample form created for Task 5.1P as a starting point. Extend to include the following:

- Field validation output, showing both correct (valid) and incorrect (invalid) user input

  - Invalid fields should show clear explanations on why

- Extended information requested from the user

  - Data of birth
  - Postal address
  - Cap sizing information

- Sample cap designs selections

  - various (at least 6) cap designs
  - option to select multiple cap designs
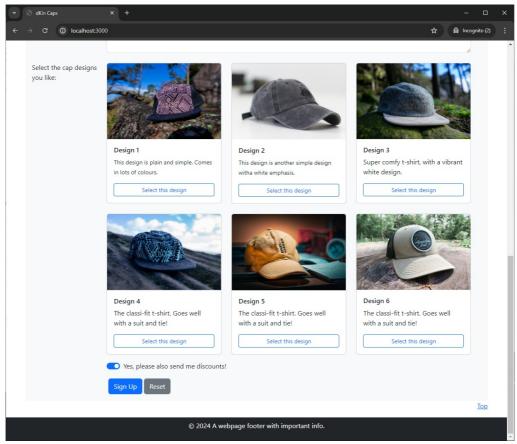 - presented in an organised (grid) format

The user input validation on the form can use the Bootstrap components, Boostrap Custom Validation Styles with an example of the possible out shown below (**NOTE:** You will need to also add the JavaScript code block shown in the Bootstrap example to the end of your `<body>` section of your page):
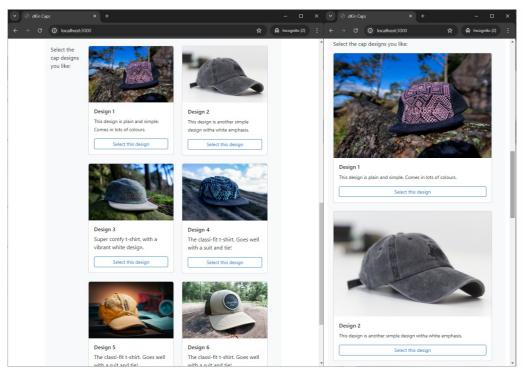
Task5.2.1 Feild Validation Examples

In presenting the customer with the various cap designs, consider using an interesting/user-friendly way of selecting cap designs. Below is one *possible* layout using Bootstrap Cards:

Task5.2.1 Sample Cap Design Selection

Making sure the web page is also usable on smaller sized screens, the cards should arrange themselves accordingly (here it is 3 columns on larger screens, 2 columns on mid-sized screens and a single column on small/mobile screens).

Task5.2.2 dKin Cap Design Selection (small sized screens)

Your goal is to present this form in a way that is clear and easy to use - regardless of the device on which it is being displayed (i.e., responsive). You can reorganise the fields or change what is being asked, as the above list is **just a starting point**.

Steps

1. You are expected to create an online form for the details requested above and that implements it in a web page. Create a new basic web document, then create the basic form elements, and add form control as appropriate, using Bootstrap and CSS to configure the form layout.

2. You should always use the appropriate form control for a particular question:

   - if the user ought to be able to select from a list (rather than typing in a response) they must be able to do so (use select list).
   - if the user ought to be able to choose only one value, they must be able to do so (use radio buttons).
   - if the user ought to be able to choose several options at once, they must be able to do so (use check boxes).
   - if there is a sensible default, it must be already selected, or entered into a text box, when the form is loaded.
   - if a field isn't required, then it should be treated as optional.

3. Think very carefully about how to use the right form controls for entering a birthday: it should never involve the user typing in the date! There are only 12 months, and no more than 31 days.

4. All form controls must have a unique, meaningful name attribute. Put in an id

also and give it the same value as name.

What will you submit?

You should submit:

- Screen-shot 1 - the web page showing the form with data entered on a normal/larger screen (i.e., monitor).
- Screen-shot 2 - the web page shown on a smaller screen (i.e., mobile device).
- Screen-shot 3 - the page when data is missing or in error - demonstrating error handling.
- The HTML file.