

Requirements:

To develop a Mobile Web Site using Angular framework.

Create a new Angular project called '<yourLoginName>A4' with four components: "header", "footer", "list", and "form", along with a service named 'getData'.

- **External Data**
 - Define a Class using <your login> as name and save it in a file (choose a file name) within the 'app' folder. This Class should include the following properties (choose appropriate property names and data types):
 - **Name, Student number, Login name, Sheridan email, Campus, and Assignment title**
 - Create an external data file (choose a name) in the 'assets/data' folder. This file should contain a const with information utilizing the login class as type.
 - Define a Class named "FieldData" and save it in a file (choose a file name) within the 'app' folder. This Class should consist of the following properties (choose appropriate property names and data types):
 - **First name, Last name, Address, Postal code, Email, and Phone #**
 - Create an external JSON file (choose a name) in the 'assets/data' folder. This file should contain at least three JSON objects utilizing the 'FieldData' class as type.
 - Import the Classes, the const, and use HTTP to retrieve the field data from the JSON file into the service.
- **Header and Footer Components**
 - Import the login class and the login data into both the header and footer components.
 - In the header component's template, utilize string interpolation to display an H2 containing **assignment title / your student number / your name**.
 - In the footer component's template, utilize string interpolation to display an H3 containing **your login / your Sheridan email / your campus**.
- **Root Component**
 - Integrate an Angular Material Tab Group with 2 tabs, including the "list" component in the first tab and the "form" component in the second tab.
- **List Component**
 - Retrieve the field data from the service.
 - Display the property names and data using Angular Material List.
- **Form Component**
 - Create an HTML form using the Reactive approach.
 - Retrieve the field data from the service to populate the form fields. Use the **setValue()** method to update the form control (field) values.
 - Implement necessary field validations as required (refer to 'Other Requirements' for validation rules). Validation can be done in either the template or the .ts file.
 - Use <ng-template> to display validation messages.
 - Upon submission, log the field name and data to the console; load the next set of form data until all data is provided, then disable the submit button.

Other Requirements:

- Field validation Rules (use Validators):
 - First Name: Required, maximum length of 30 characters.
 - Last Name: Required, maximum length of 30 characters.
 - Address (textarea): Required, maximum length of 200 characters.
 - Postal Code: Required, Canadian postal code format.
 - Email: Required, valid email format.
 - Phone number: maximum length of 15 characters.
- Implement using TypeScript.
- Implement techniques covered in class for form creation, data validation, and message display.
- Pay attention to formatting, layout, colors, background, and labeling for evaluation.
- During grading, Chrome's Toggle Device Bar will be utilized, set to iPhone 12 in portrait mode.
- Keep in mind that the emphasis is on mobile optimization, so ensure that formatting and layout are optimized accordingly.

Grading Rubric

Items	Marks
Classes and Data	1
Form Creation	4
Data Validation and Messages	4
Formatting and Presentation	1
Site NOT published to cPanel	-5

Submission:

- Deploy your project to **cPanel**. Deployment can be delayed up to 3 days without penalty.
- Submit to "Assignment 4" Dropbox on SLATE (e-mail submissions will NOT be accepted):
 - A zipped copy of the "**src**" folder of your Angular Project Folder.
 - Include the **cPanel** URL to your published site in the comment section of the Dropbox.
 - No need to password-protect your cPanel site.
- For assistance with publishing/deploying, refer to the "**Angular-Deployment**" in the References section.
- Assignment can be submitted up to 3 days after the due date with a penalty of 10% per day. After 3 days, the assignment will no longer be accepted.
- Only the final submission will be accepted and graded.
- **Assignment must be completed individually.**