

# Title: Angular Custom Directives Lab

## Exercise

**Objective:** In this lab exercise, you will practice creating custom directives in Angular to handle real-world scenarios. You will create custom directives to enhance the functionality and user experience of a sample application.

### Requirements:

- Basic understanding of Angular framework.
- Angular CLI installed.
- Code editor (e.g., Visual Studio Code).

**Scenario:** You are tasked with developing a user management system for a company. The system should allow users to input their personal details and display them in a visually appealing manner. You are required to implement custom directives to enhance the input fields and improve the user experience.

### Tasks:

#### 1. Create Angular Project:

- Generate a new Angular project using Angular CLI.
- Navigate into the project directory.

#### 2. Create Custom Directives:

- Create a custom directive named `InputFormatDirective`. This directive should format the input field based on specified criteria (e.g., uppercase, lowercase, title case).
- Create another custom directive named `PasswordStrengthDirective`. This directive should provide real-time feedback on the strength of the password entered by the user.

#### 3. Implement Custom Directives:

- Implement the `InputFormatDirective` to format the input fields for first name, last name, and email address. Allow users to choose the format (e.g., uppercase, lowercase).
- Implement the `PasswordStrengthDirective` to display a strength meter below the password input field. Provide visual feedback based on the strength of the password (e.g., weak, medium, strong).

#### 4. Integrate Custom Directives:

- Integrate the custom directives into the user management form.
- Apply the `InputFormatDirective` to the appropriate input fields.

- Apply the `PasswordStrengthDirective` to the password input field.

#### 5. **Test and Validate:**

- Test the application to ensure that the custom directives are functioning as expected.
- Verify that the input fields are formatted according to the specified criteria.
- Test the password strength meter to ensure it provides accurate feedback.

#### 6. **Enhancements (Optional):**

- Add additional features or improvements to the custom directives (e.g., customizable password strength criteria, animations).

#### **Submission:**

- Submit the source code of your Angular project along with a brief report documenting the implementation of the custom directives and any additional features or enhancements.

**Note:** Ensure that you follow best practices for Angular development, including proper folder structure, component separation, and code organization. Use HTML and CSS to enhance the visual presentation of the user management form and the custom directives.