

# Title: Angular Custom Pipes Lab Exercise

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

## Requirements:

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

**Scenario:** You are developing a task management application for a project team. The application should display task details and provide various formatting options for the task descriptions. You are required to implement custom pipes to format and display task details in a user-friendly manner.

## Tasks:

### 1. Create Angular Project:

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

### 2. Create Custom Pipes:

- Create a custom pipe named `ShortenDescriptionPipe`. This pipe should truncate long task descriptions to a specified length and append ellipsis (...) at the end.
- Create another custom pipe named `FormatDatePipe`. This pipe should format task due dates into a human-readable format (e.g., "Today", "Tomorrow", "Next week").

### 3. Implement Custom Pipes:

- Implement the `ShortenDescriptionPipe` to truncate the task descriptions to a specified length (e.g., 50 characters) and append ellipsis (...) at the end.
- Implement the `FormatDatePipe` to format the task due dates into a human-readable format based on their proximity to the current date (e.g., "Today", "Tomorrow", "Next week").

### 4. Integrate Custom Pipes:

- Integrate the custom pipes into the task management application.
- Apply the `ShortenDescriptionPipe` to the task descriptions displayed in the task list.
- Apply the `FormatDatePipe` to the due dates of the tasks.

### 5. Test and Validate:

- Test the application to ensure that the custom pipes are functioning as expected.
- Verify that the task descriptions are truncated and appended with ellipsis.
- Check that the due dates are formatted into human-readable formats based on their proximity to the current date.

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

- Add additional features or improvements to the custom pipes (e.g., customizable truncation length, additional date formatting options).

#### **Submission:**

- Submit the source code of your Angular project along with a brief report documenting the implementation of the custom pipes 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 task management application and the custom pipes.