Angular
Level 3 Certification
mini-project

angulartraining.com

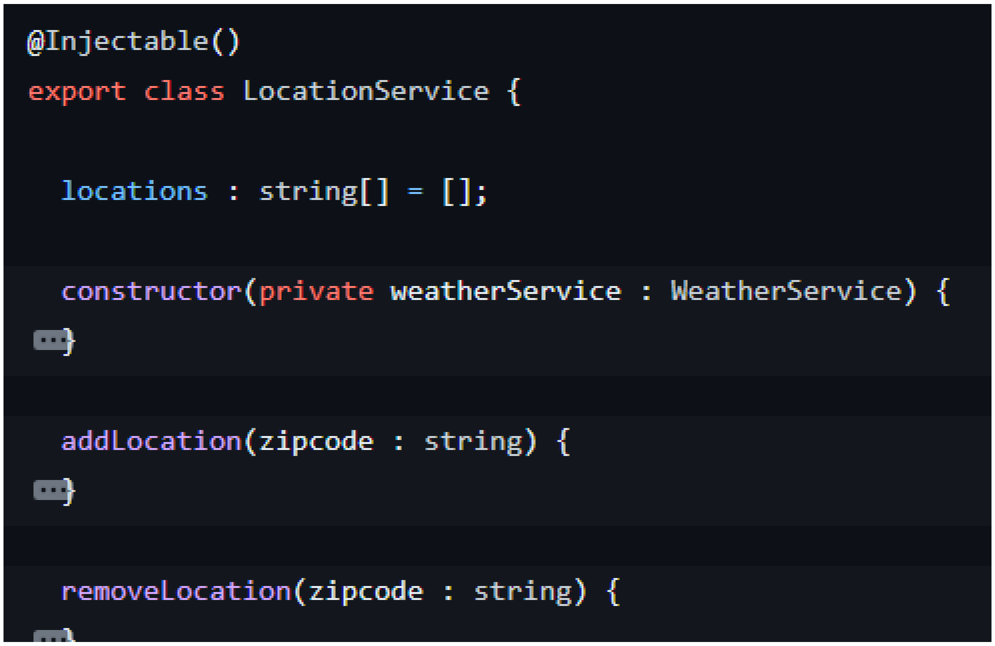
This document is online at
https://bit.ly/at-cert-lv3-weather

GOAL: Improve a small weather application that displays current weather conditions and 5-day forecast info for a list of locations

This project will be based on an existing application. [Click here to check out a version of the code you'll need](#).

In order to submit your work, you'll need to provide either the link to your StackBlitz project or a link to a Git repository that contains your code with the solution for the following exercises. A public URL of your hosted application is needed, too. More information on how to submit your project can be found at the end of this document.

STEP #1 - Implement a reactive notification system in LocationService



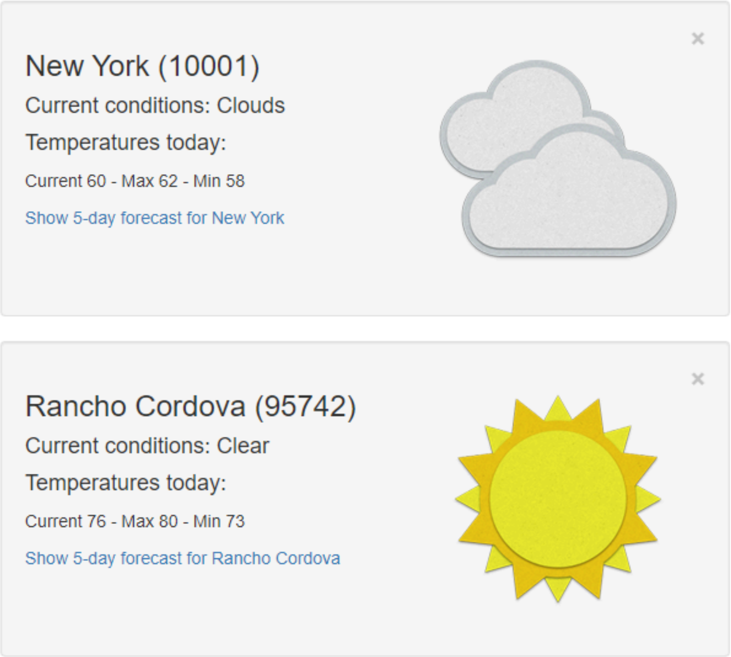
```
@Injectable()
export class LocationService {

  locations : string[] = [];

  constructor(private weatherService : WeatherService) {
⋯}

  addLocation(zipcode : string) {
⋯}

  removeLocation(zipcode : string) {
⋯}
```

The current implementation of [LocationService](#) stores all locations (zip codes / postal codes) and calls WeatherService when there's a change. Your goal is to break that dependency so that any number of components/services can receive updates when the list of locations changes. As a result, WeatherService should not be referenced in LocationService anymore.

STEP #2 - Turn the current card layout into tabs

The goal of this step is to replace the current card display and create a generic, reusable tab mechanism that supports any number of tabs. **No external libraries such as Material are allowed**, and your tab system should not depend on weather data, it should be reusable in a different context.

BEFORE:



New York (10001)

Current conditions: Clouds

Temperatures today:

Current 60 - Max 62 - Min 58

Show 5-day forecast for New York

Rancho Cordova (95742)

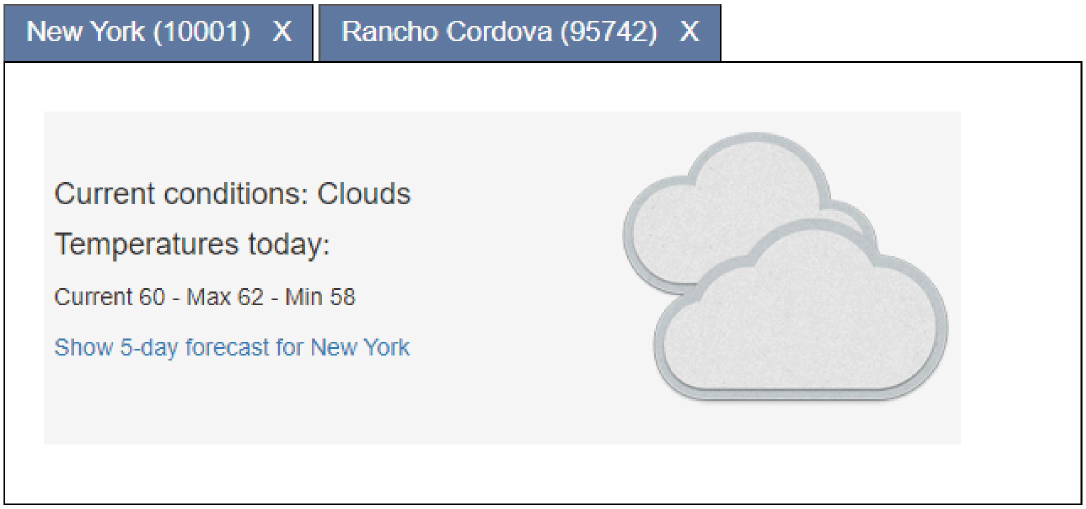
Current conditions: Clear

Temperatures today:

Current 76 - Max 80 - Min 73

Show 5-day forecast for Rancho Cordova

AFTER:



Current conditions: Clouds

Temperatures today:

Current 60 - Max 62 - Min 58

Show 5-day forecast for New York

Your solution should allow the rendered tabs to be fully reusable and customizable, including the **HTML contents of the tab** that can be **entirely different from one tab to the other**.

Tabs can be closed by clicking the “X” button to remove a location, and of course multiple tabs can be added as the user adds locations using the zip code entry form.

The logic to close a tab and to decide which tab is displayed **must be within your tabs component**. The parent component **doesn’t have to know how to remove a tab or how to select a tab**. Your component must handle that itself.

STEP #3 - Implement a generic caching system for the application's data

Implement a generic caching system using the browser's storage so that both 5-day forecasts and current conditions get cached for 2 hours for each location, meaning that the app won't make HTTP requests for any single location more than once every 2 hours.

A full-page refresh of the app or opening the app in another tab must use that cached data instead of making a new request to the server.

That duration should be customizable as well so we can test the cache by using a few seconds instead of 2 hours. Also, any data should be storable in that cache and not limited to weather data. We want a truly generic implementation that works with anything.

Extra instructions:

Your work will be evaluated based on the quality of your solution:

- Is the code truly reusable?
- Is the code easy to understand?
- Is the code concise?
- Feel free to add comments to explain your code wherever needed

Once your application is working with the 3 steps implemented, submit your work by going to https://angulartraining.com/certification/level3-step2.html