Assignment instructions

20 minutes to complete

30 student solutions

In this assignment exercise, you will redevelop the previous assignment (Weather App) with layout views.

Requirement:

Imagine a weather application that shows weather details of the selected city. Create an Asp.Net Core Web Application that fulfils this requirement.

Consider the following hard-coded weather data of 3 cities.
```
CityUniqueCode = "LDN", CityName = "London",
        DateAndTime = "2030-01-01 8:00",
        TemperatureFahrenheit = 33

CityUniqueCode = "NYC", CityName = "London",
        DateAndTime = "2030-01-01 3:00",
        TemperatureFahrenheit = 60

CityUniqueCode = "PAR", CityName = "Paris", DateAndTime
        = "2030-01-01 9:00",  TemperatureFahrenheit = 82
```

Consider a model class called '**CityWeather**' with following properties:
```
string CityUniqueCode
```

```
string CityName
DateTime DateAndTime
int TemperatureFahrenheit
```

Example #1:

If you receive a HTTP GET request at path "/", it has to generate a view with weather details of all cities with HTTP status code 200.
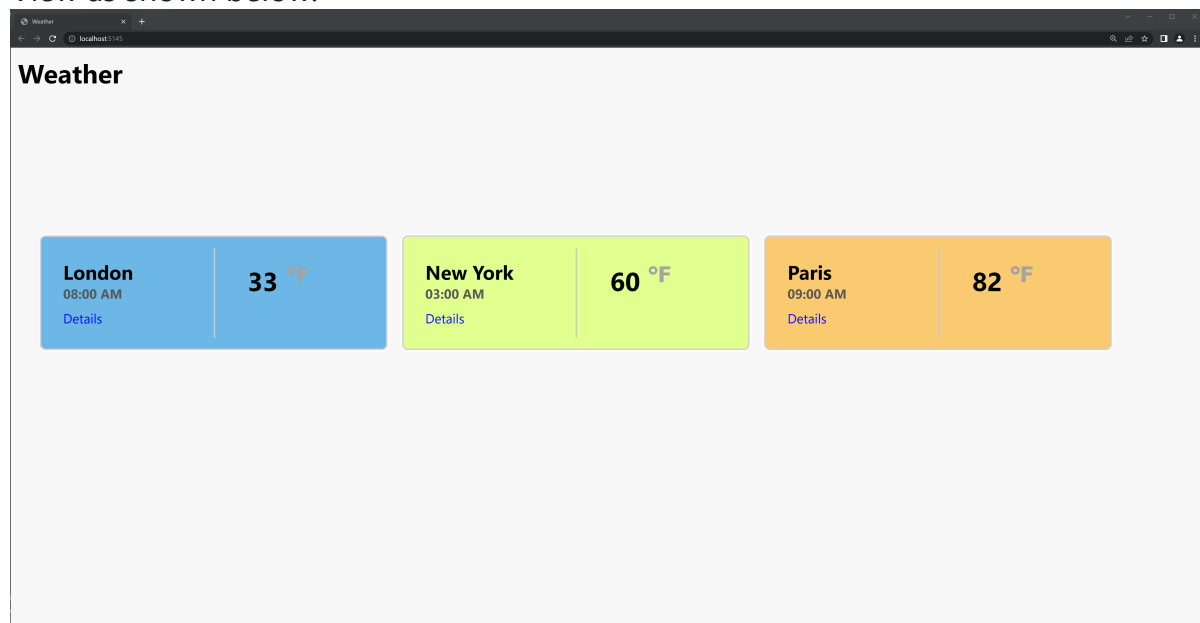
Request Url: /

Request Method: GET

Response Status Code: 200

Response body (output):

View as shown below.

Example #2:

If you receive a HTTP GET request at path "/weather/{cityCode}", it has to generate a view with weather details of the selected city s with HTTP status code 200.
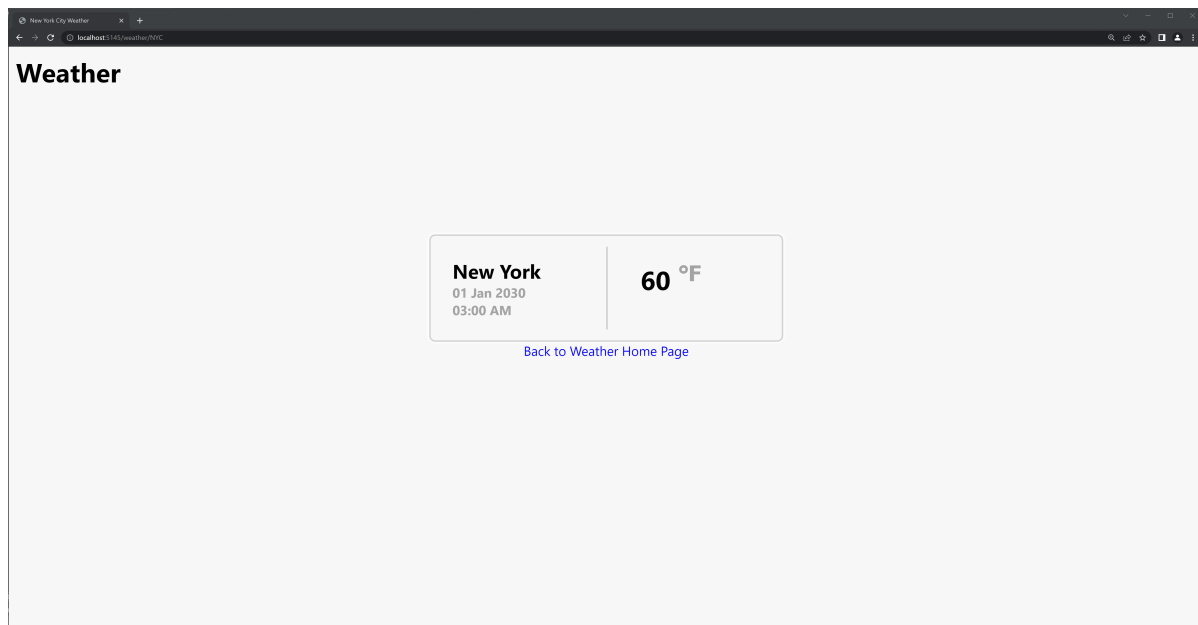
Request Url: /

Request Method: GET

Response Status Code: 200

Response body (output):

View as shown below.

Instructions:

- Create controller(s) with attribute routing.

- Initialize the hard-coded data as collection of model objects in the controller.

- Use strongly-typed views. Send model object(s) to view.

- If you supply an invalid city code as route parameter, it should show a page with proper error message, instead of throwing an exception.

- Use CSS styles, layout views, _ViewImports, _ViewStart, Razor view engine as per the necessity.

- The UI should be consistent and modern. It should minimum look like as shown in the screenshot. Optionally, you can try enhancing it based on your thoughts.

- Apply background color for each box, based on the following categories of temperature value. Write local function in view, to determine the apppriate css class to apply background color.

`Fahrenheit` **`is`** `less than 44 = blue background color`

`Fahrenheit` **`is`** `between 44` **`and`** `74 = blue background`
`        color`

`Fahrenheit` **`is`** `greater than 74 = blue background color`
- The CSS file is provided as downloadable resource for applying essential styles. You can download and use it.

Questions for this assignment

Check your source code with Instructor's source code.