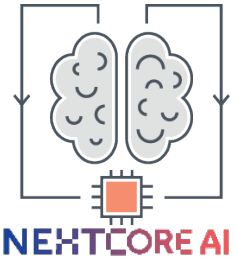
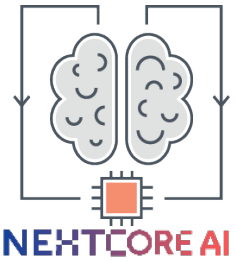


Angular Material schematics

- For example, you can create a side navigation layout by executing this:

```
$ ng generate @angular/material:material-nav --name=side-nav

CREATE src/app/side-nav/side-nav.component.css (110 bytes)
CREATE src/app/side-nav/side-nav.component.html (945 bytes)
CREATE src/app/side-nav/side-nav.component.spec.ts (619 bytes)
CREATE src/app/side-nav/side-nav.component.ts (489 bytes)
UPDATE src/app/app.module.ts (882 bytes)
```



Material Toolbar modifying page

1. Update app.component.ts to use an inline template
2. Remove app.component.html and app.component.cssCurrent date

src/app/app.component.ts

```
import { Component } from '@angular/core'

@Component({
  selector: 'app-root',
  template: `
    <div style="text-align:center">
      <h1>
      LocalCast Weather
      </h1>
      <div>Your city, your forecast, right now!</div>
      <h2>Current Weather</h2>
      <app-current-weather></app-current-weather>
    </div>
  `

})
export class AppComponent {}
```

src/app/app.component.ts

```
import { Component } from '@angular/core'

@Component({
  selector: 'app-root',
  template: `
    <div style="text-align:center">
      <h1>
      LocalCast Weather
      </h1>
      <div>Your city, your forecast, right now!</div>
      <h2>Current Weather</h2>
      <app-current-weather></app-current-weather>
    </div>
  `
})
export class AppComponent {}
```

Material Toolbar continued

3. Observe the h1 tag in app.component.ts :

src/app/app.component.ts

```
<h1>
  LocalCast Weather
</h1>
```

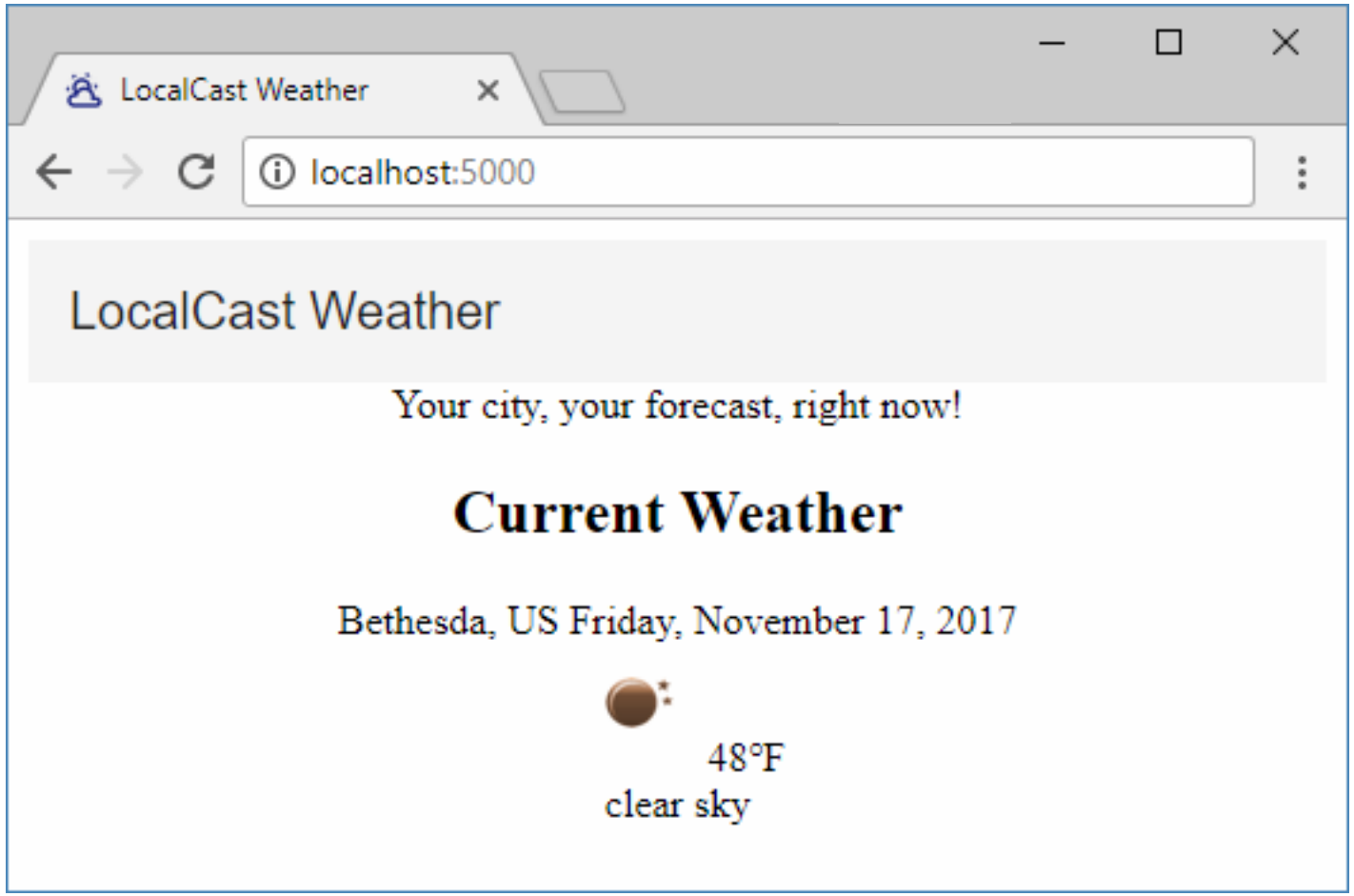
4. Update the h1 tag with mat-toolbar:

src/app/app.component.ts

```
<mat-toolbar>
  <span>LocalCast Weather</span>
</mat-toolbar>
```

toolbar

5. Observe the result; you should see a toolbar, as illustrated:



LocalCast Weather Toolbar

6. Update mat-toolbar with a more attention-grabbing color:

src/app/app.component.ts

```
<mat-toolbar color="primary">
```

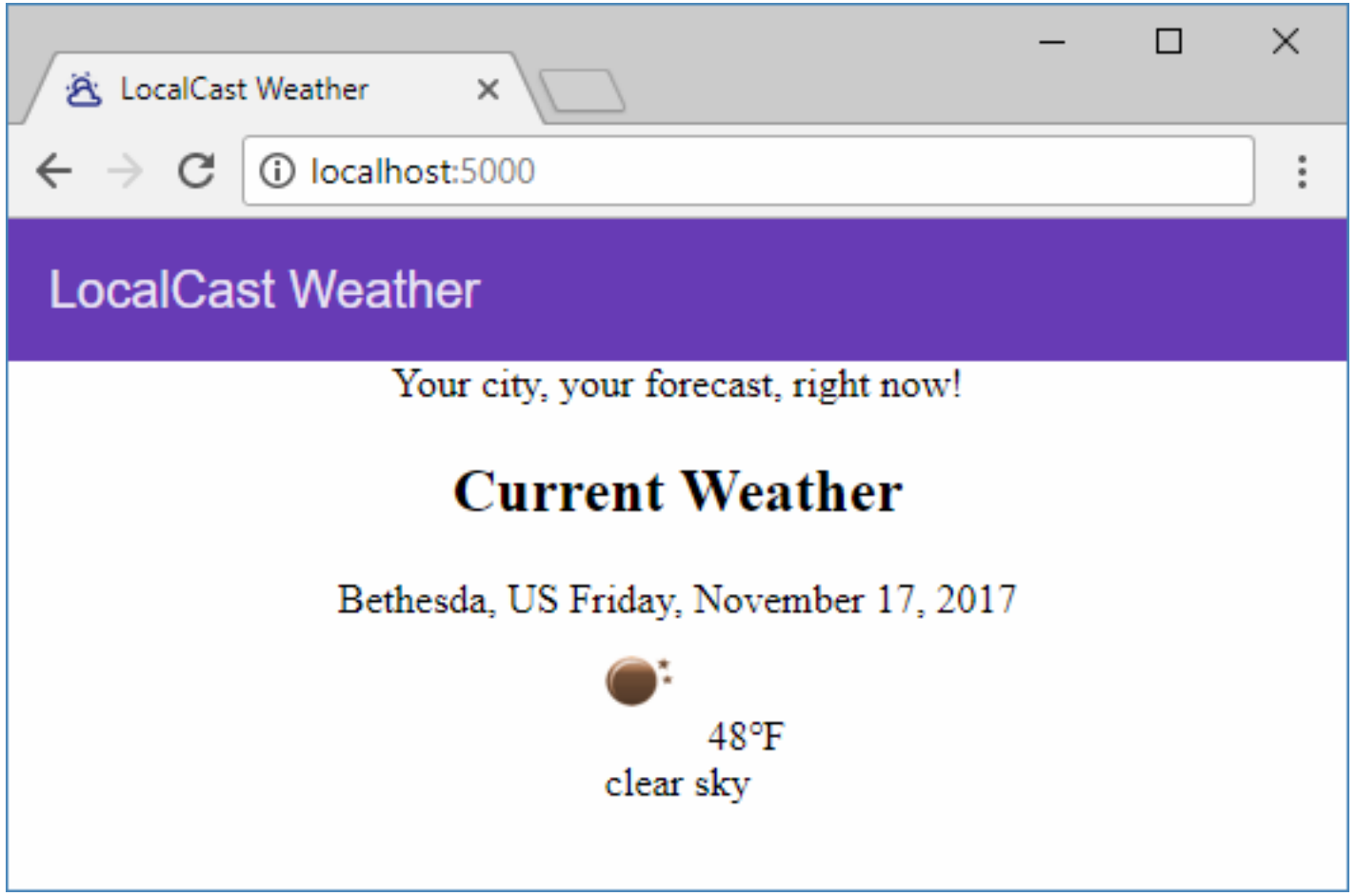
src/styles.css

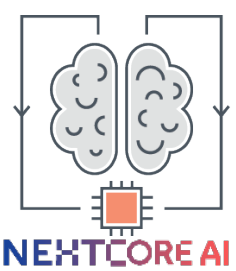
```
body {
  margin: 0;
}
```

src/styles.css

```
.content-margin {
margin-left: 8px;
margin-right: 8px;
}
```

LocalCast Weather





Representing weather in Material Card

1. Import MatCardModule in material.module:

src/app/material.module.ts

```
import { ..., MatCardModule} from '@angular/material'
...
@NgModule({
  imports: [..., MatCardModule],
  exports: [..., MatCardModule],
})
```

Represent weather in Material Card

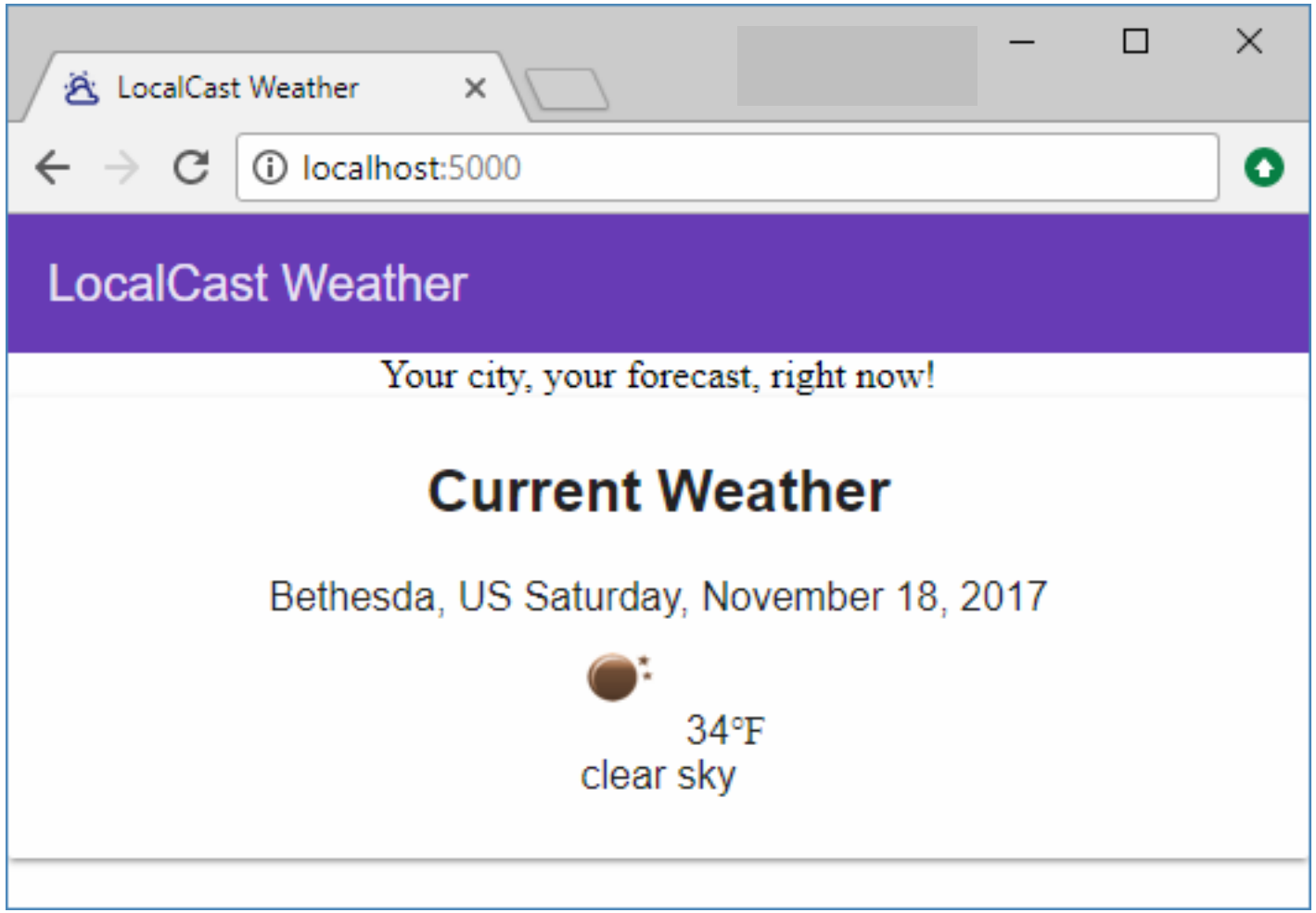
2. In app.component, surround <app-current-weather> with <mat-card> :

src/app/app.component.ts

```
<div style="text-align:center">
  <mat-toolbar color="primary">
    <span>LocalCast Weather</span>
  </mat-toolbar>
  <div>Your city, your forecast, right now!</div>
  <mat-card>
    <h2>Current Weather</h2>
    <app-current-weather></app-current-weather>
  </mat-card>
</div>
```

Represent weather in Material Card

3. Observe the barely distinguishable card element, as shown:



weather in Material Card

4. Remove style="text-align:center" from the surrounding :

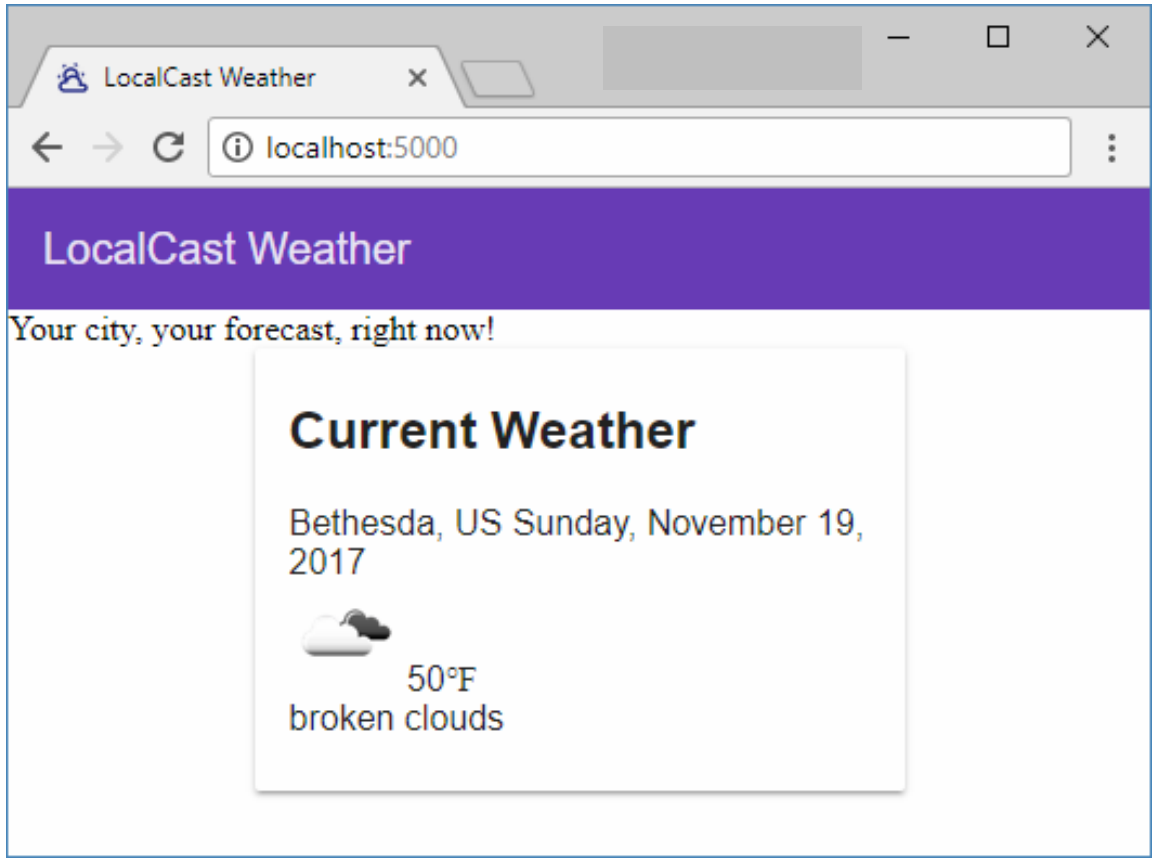
src/app/app.component.ts

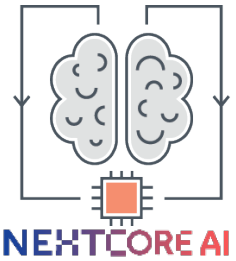
```
<div fxLayout="row">
  <div fxFlex></div>
  <div fxFlex="300px">
    ...
  </div>
  <div fxFlex></div>
</div>
```

weather in Material Card

5. Surround `<mat-card>` with the preceding HTML

6. Observe that the card element is properly centered, as follows:





Card header and content

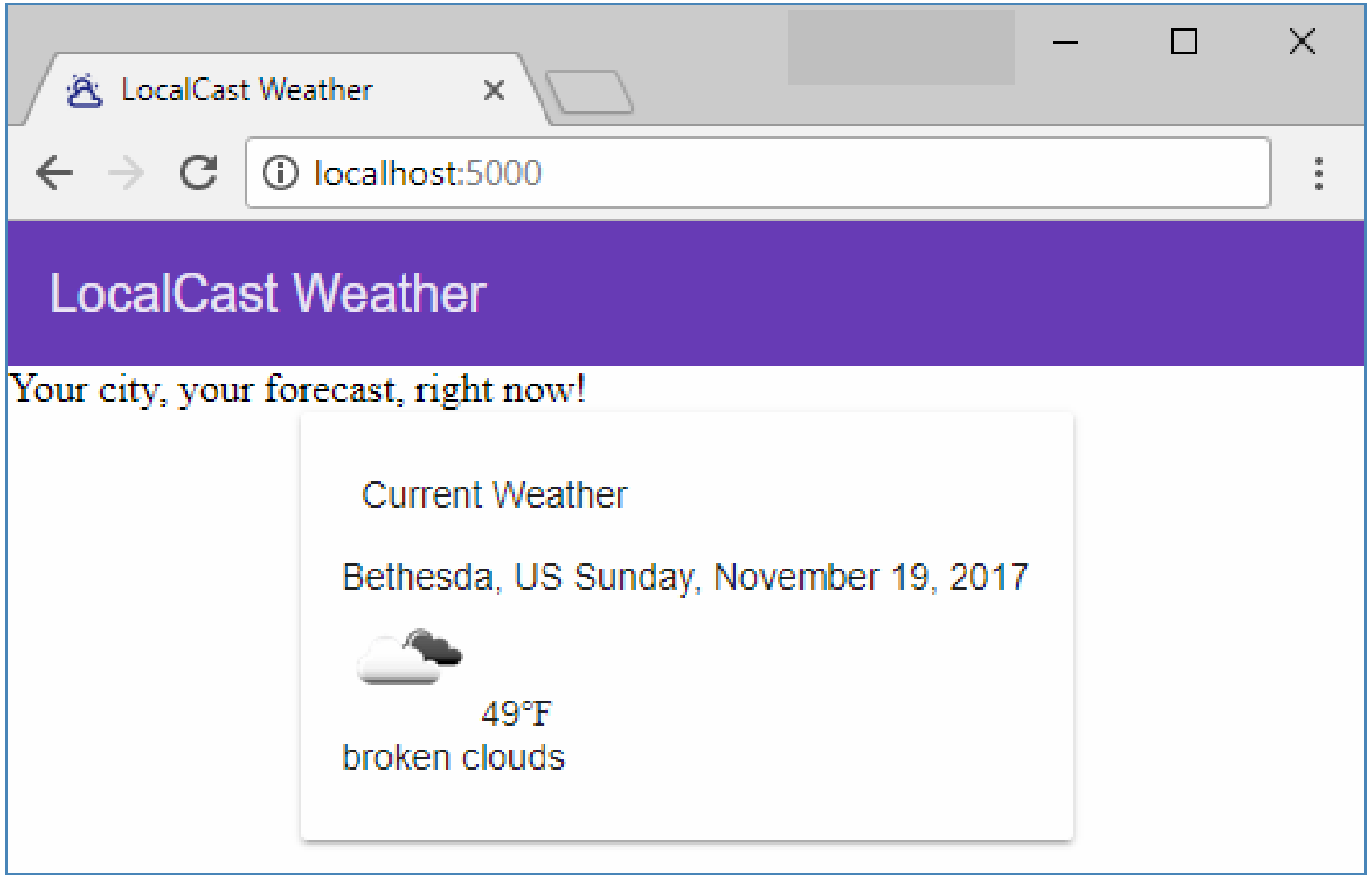
src/app/app.component.ts

```
<mat-toolbar color="primary">
  <span>LocalCast Weather</span>
</mat-toolbar>
<div>Your city, your forecast, right now!</div>
<div fxLayout="row">
  <div fxFlex></div>
  <mat-card fxFlex="300px">
    <mat-card-header>
      <mat-card-title>Current Weather</mat-card-title>
    </mat-card-header>
    <mat-card-content>
      <app-current-weather></app-current-weather>
    </mat-card-content>
  </mat-card>
  <div fxFlex></div>
</div>
```

src/app/app.component.ts

```
<mat-toolbar color="primary">
  <span>LocalCast Weather</span>
</mat-toolbar>
<div>Your city, your forecast, right now!</div>
<div fxLayout="row">
  <div fxFlex></div>
  <mat-card fxFlex="300px">
    <mat-card-header>
      <mat-card-title>Current Weather</mat-card-title>
    </mat-card-header>
    <mat-card-content>
      <app-current-weather></app-current-weather>
    </mat-card-content>
  </mat-card>
  <div fxFlex></div>
</div>
```

mat-card-header

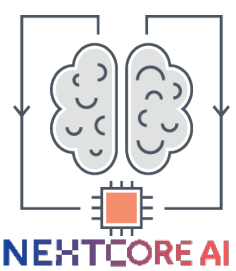


Material typography

- https://material.angular.io/guide/typography

Typography is a way of arranging type to make text legible, readable, and appealing when displayed.

Class Name	Usage
`display-4` , `display-3` , `display-2` and `display-1`	Large, one-off headers, usually at the top of the page (for example, a hero header)
`headline`	Section heading corresponding to the `<h1>` tag
`title`	Section heading corresponding to the `<h2>` tag
`subheading-2`	Section heading corresponding to the `<h3>` tag
`subheading-1`	Section heading corresponding to the `<h4>` tag
`body-1`	Base body text
`body-2`	Bolder body text
`caption`	Smaller body and hint text
`button`	Buttons and anchors



Applying typography

There are multiple ways to apply typography. One way is to leverage the mat-typography class and use the corresponding HTML tag like <h2>:

src/app/app.component.ts

```
<mat-card-header class="mat-typography">
  <mat-card-title><h2>Current Weather</h2></mat-card-title>
</mat-card-header>
```

src/app/app.component.ts

```
<mat-card-header class="mat-typography">
  <mat-card-title><h2>Current Weather</h2></mat-card-title>
</mat-card-header>
```

Typography on an element

- Another way is to apply the specific typography directly on an element, like class="mat-title“

src/app/app.component.ts

```
<mat-card-title><div class="mat-title">Current Weather</div></mat-card-title>
```

src/app/app.component.ts

```
<mat-card-title><div class="mat-title">Current Weather</div></mat-card-title>
```