

Navigation

Home

Trainer Form

Berries

Poké Balls

Favorite Pokémon

Falkner (Violet City)

Bugsy (Azalea Town)

Whitney (Goldenrod City)

Morty (Ecruteak City)

Chuck (Cianwood City)

Jasmine (Olivine City)

Pryce (Mahogany Town)

Clair (Blackthorn City)

### Pokemon Trainer Registration Form

Name:

Age:

Address:

Favorite Pokemon:

Submit

#### Pokemon Trainer List:

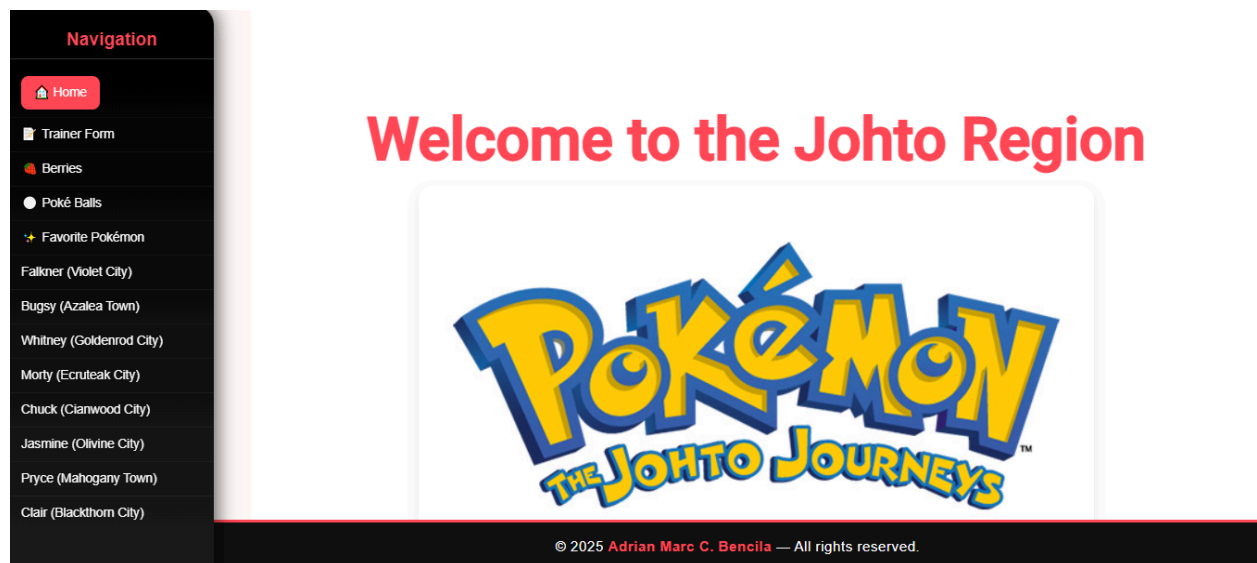
Name: Pikachu

Age: 4

Address: pokemon

Favorite Pokemon: pichu

© 2025 Adrian Marc C. Bencila — All rights reserved.



Navigation

Home

Trainer Form

Berries

Poké Balls

Favorite Pokémon

Falkner (Violet City)

Bugsy (Azalea Town)

Whitney (Goldenrod City)

Morty (Ecruteak City)






Chuck (Cianwood City)

Jasmine (Olivine City)

Pryce (Mahogany Town)

Clair (Blackthorn City)

🍓 JOHTO REGION BERRIES

Berry	Name	Effect
	Oran Berry	Restores 10 HP
	Sitrus Berry	Restores 25% HP
	Lum Berry	Cures any status condition
	Cheri Berry	Cures paralysis
	Chesto Berry	Wakes up a sleeping Pokémon

© 2025 Adrian Marc C. Bencila — All rights reserved.

Navigation

Home

Trainer Form

Berries

Poké Balls

Favorite Pokémon

Falkner (Violet City)

Bugsy (Azalea Town)

Whitney (Goldenrod City)

Morty (Ecruteak City)






Chuck (Cianwood City)

Jasmine (Olivine City)

Pryce (Mahogany Town)

Clair (Blackthorn City)

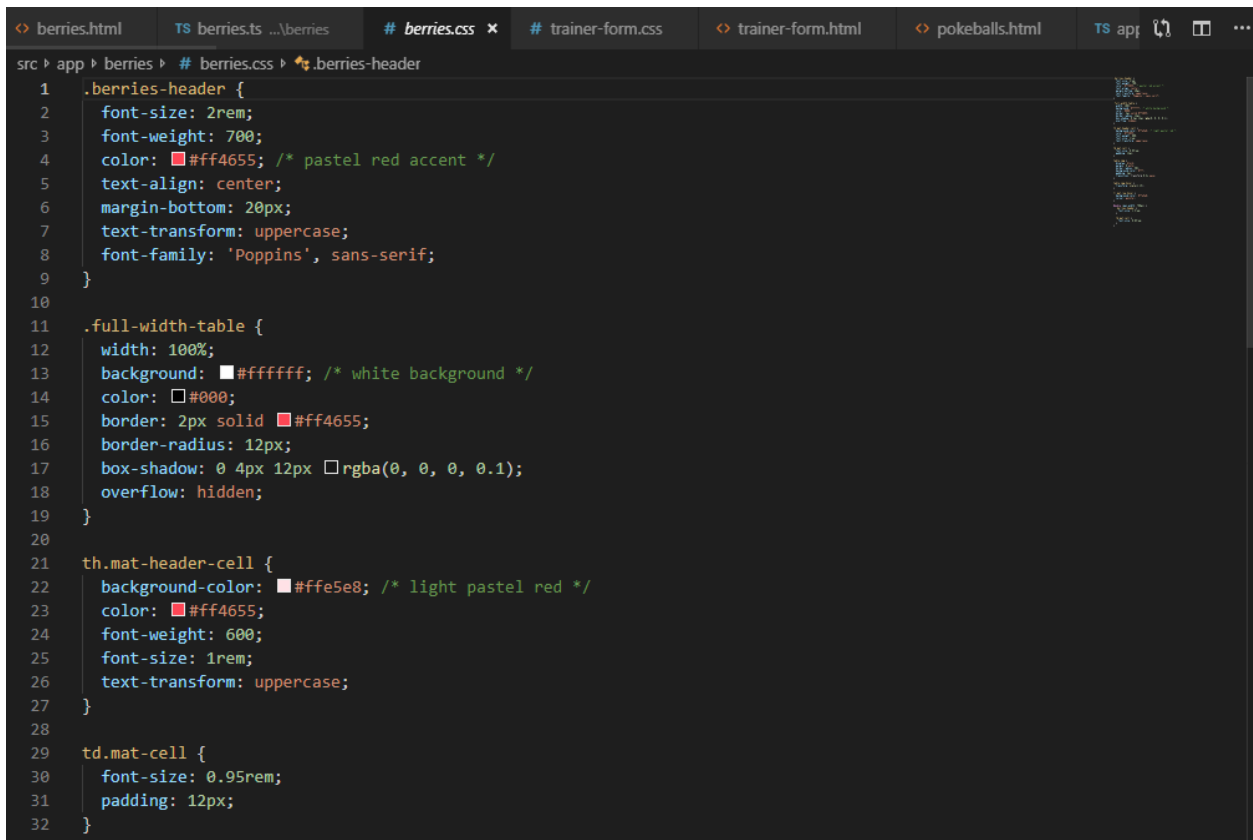
🏠 POKÉ BALLS COLLECTION

Poké Ball	Name	Type
	Poké Ball	Standard
	Great Ball	Improved
	Ultra Ball	High Performance
	Master Ball	Legendary
	Premier Ball	Special

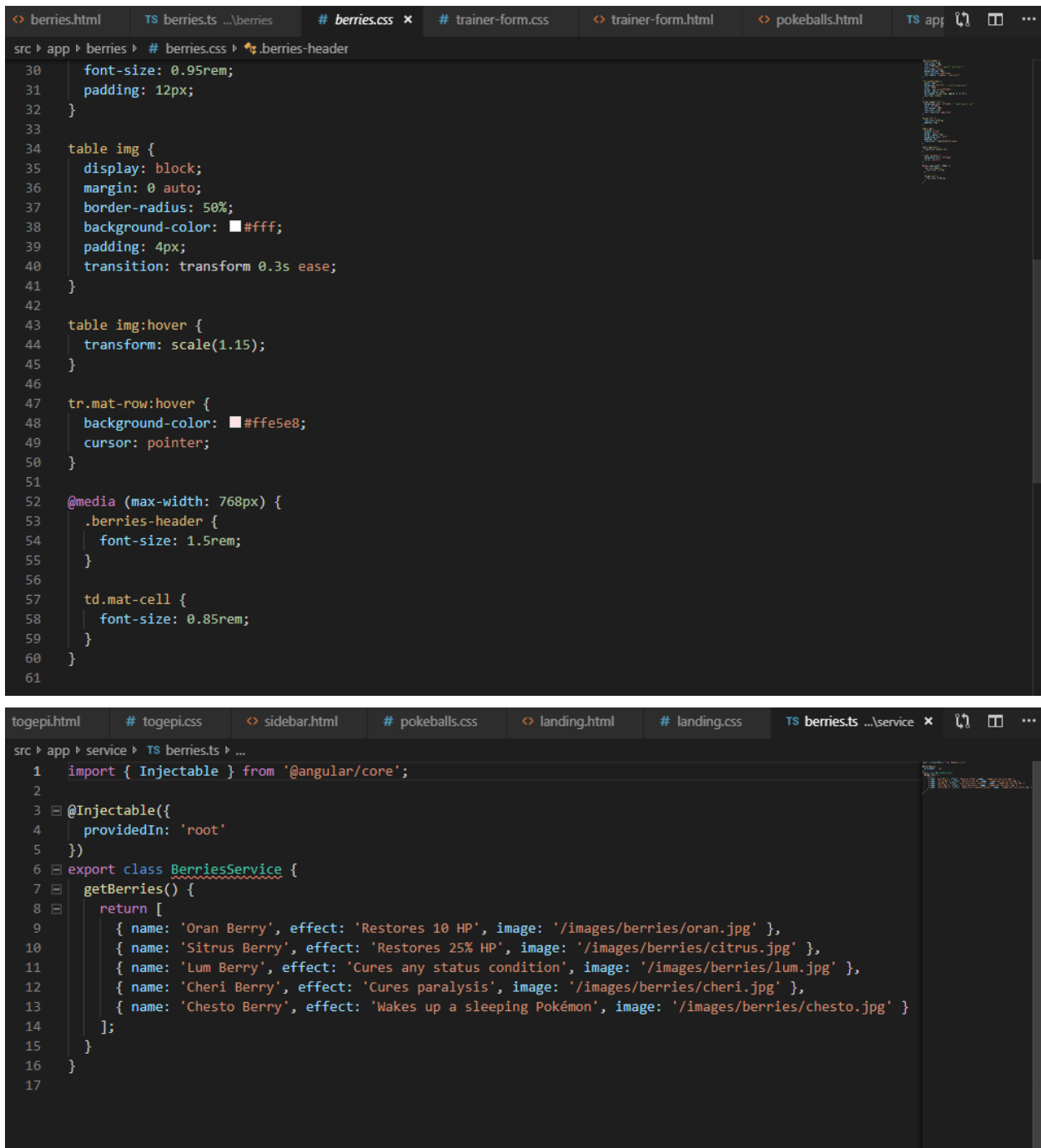
© 2025 Adrian Marc C. Bencila — All rights reserved.

```
berries.html x TS berries.ts ...berries # trainer-form.css < trainer-form.html < pokeballs.html TS app-routing-module.ts
src > app > berries > berries.html > h1.berries-header
1 <h1 class="berries-header"> Johto Region Berries</h1>
2
3 <table mat-table [dataSource]="dataSource" class="mat-elevation-z8 full-width-table">
4
5   <ng-container matColumnDef="image">
6     <th mat-header-cell *matHeaderCellDef> Berry </th>
7     <td mat-cell *matCellDef="let berry">
8       <img [src]="berry.image" alt="{{ berry.name }}" width="40" height="40">
9     </td>
10  </ng-container>
11
12  <ng-container matColumnDef="name">
13    <th mat-header-cell *matHeaderCellDef> Name </th>
14    <td mat-cell *matCellDef="let berry">{{ berry.name }}</td>
15  </ng-container>
16
17  <ng-container matColumnDef="effect">
18    <th mat-header-cell *matHeaderCellDef> Effect </th>
19    <td mat-cell *matCellDef="let berry">{{ berry.effect }}</td>
20  </ng-container>
21
22  <tr mat-header-row *matHeaderRowDef="displayedColumns"></tr>
23  <tr mat-row *matRowDef="let row; columns: displayedColumns"></tr>
24 </table>
25
```

```
berries.html x TS berries.ts ...berries x # trainer-form.css < trainer-form.html < pokeballs.html TS app-routing-module.ts
src > app > berries > TS berries.ts > ...
1 import { Component } from '@angular/core';
2 import { MatTableModule } from '@angular/material/table';
3 import { BerriesService } from '../service/berries';
4
5 @Component({
6   selector: 'app-berries',
7   standalone: true,
8   imports: [MatTableModule],
9   templateUrl: './berries.html',
10  styleUrls: ['./berries.css']
11 })
12 export class Berries {
13   dataSource: { name: string; effect: string; image: string }[] = [];
14   displayedColumns: string[] = ['image', 'name', 'effect'];
15
16   constructor(private b: BerriesService) {}
17
18   ngOnInit(): void {
19     this.dataSource = this.b.getBerries();
20   }
21 }
22
```



```
src > app > berries > # berries.css > .berries-header
1  .berries-header {
2      font-size: 2rem;
3      font-weight: 700;
4      color: #ff4655; /* pastel red accent */
5      text-align: center;
6      margin-bottom: 20px;
7      text-transform: uppercase;
8      font-family: 'Poppins', sans-serif;
9  }
10
11  .full-width-table {
12      width: 100%;
13      background: #ffffff; /* white background */
14      color: #000;
15      border: 2px solid #ff4655;
16      border-radius: 12px;
17      box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
18      overflow: hidden;
19  }
20
21  th.mat-header-cell {
22      background-color: #ffe5e8; /* light pastel red */
23      color: #ff4655;
24      font-weight: 600;
25      font-size: 1rem;
26      text-transform: uppercase;
27  }
28
29  td.mat-cell {
30      font-size: 0.95rem;
31      padding: 12px;
32  }
```



The image shows two screenshots of a code editor (VS Code) displaying code for a Pokémon application. The top screenshot shows a CSS file named `berries.css` with styles for a berries header, a table of berries, and a media query for mobile devices. The bottom screenshot shows a TypeScript file named `berries.ts` defining an `Injectable` service that returns an array of berry objects.

```
src > app > berries > # berries.css > *.berries-header
30   font-size: 0.95rem;
31   padding: 12px;
32 }
33
34 table img {
35   display: block;
36   margin: 0 auto;
37   border-radius: 50%;
38   background-color: #fff;
39   padding: 4px;
40   transition: transform 0.3s ease;
41 }
42
43 table img:hover {
44   transform: scale(1.15);
45 }
46
47 tr.mat-row:hover {
48   background-color: #ffe5e8;
49   cursor: pointer;
50 }
51
52 @media (max-width: 768px) {
53   .berries-header {
54     font-size: 1.5rem;
55   }
56
57   td.mat-cell {
58     font-size: 0.85rem;
59   }
60 }
61
```

```
togepi.html # togepi.css < sidebar.html # pokeballs.css < landing.html # landing.css TS berries.ts ...service x
src > app > service > TS berries.ts > ...
1  import { Injectable } from '@angular/core';
2
3  @Injectable({
4    providedIn: 'root'
5  })
6  export class BerriesService {
7    getBerries() {
8      return [
9        { name: 'Oran Berry', effect: 'Restores 10 HP', image: '/images/berries/oran.jpg' },
10       { name: 'Sitrus Berry', effect: 'Restores 25% HP', image: '/images/berries/citrus.jpg' },
11       { name: 'Lum Berry', effect: 'Cures any status condition', image: '/images/berries/lum.jpg' },
12       { name: 'Cheri Berry', effect: 'Cures paralysis', image: '/images/berries/cheri.jpg' },
13       { name: 'Chesto Berry', effect: 'Wakes up a sleeping Pokémon', image: '/images/berries/chesto.jpg' }
14     ];
15   }
16 }
17
```

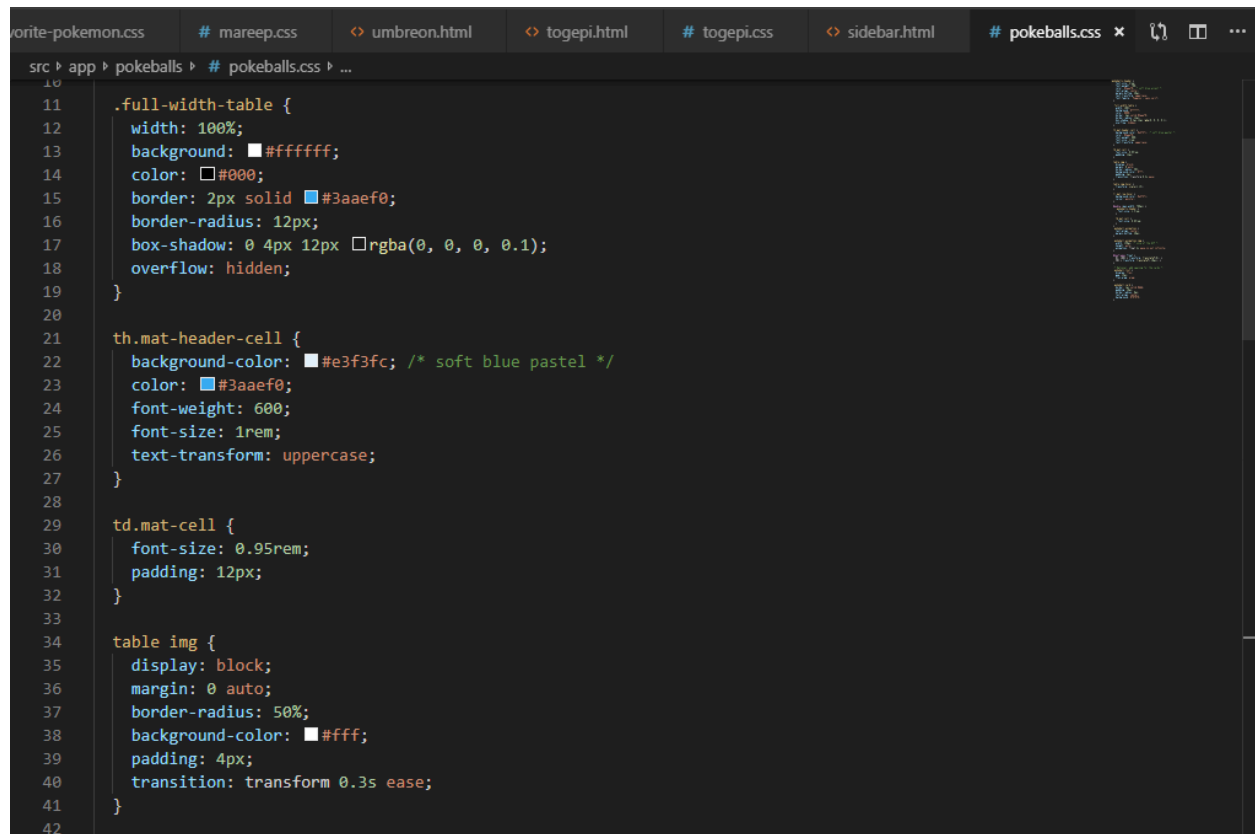
```
# togepi.css x < sidebar.html # pokeballs.css < landing.html # landing.css TS berries.ts ...service TS pokeballs.ts x [icon] [icon] ...

src > app > service > TS pokeballs.ts > ...
1  import { Injectable } from '@angular/core';
2
3  @Injectable({
4    providedIn: 'root'
5  })
6  export class PokeballsService {
7    getPokeballs() {
8      return [
9        { name: 'Poké Ball', type: 'Standard', image: '/images/pokeballs/standard.jpg' },
10       { name: 'Great Ball', type: 'Improved', image: '/images/pokeballs/great.jpg' },
11       { name: 'Ultra Ball', type: 'High Performance', image: '/images/pokeballs/ultra.jpg' },
12       { name: 'Master Ball', type: 'Legendary', image: '/images/pokeballs/master.jpg' },
13       { name: 'Premier Ball', type: 'Special', image: '/images/pokeballs/premier.jpg' }
14     ];
15   }
16 }
17
```

```
togepi.css x < sidebar.html # pokeballs.css < landing.html # landing.css TS berries.ts ...service TS pokeballs.ts x [icon] [icon] ...

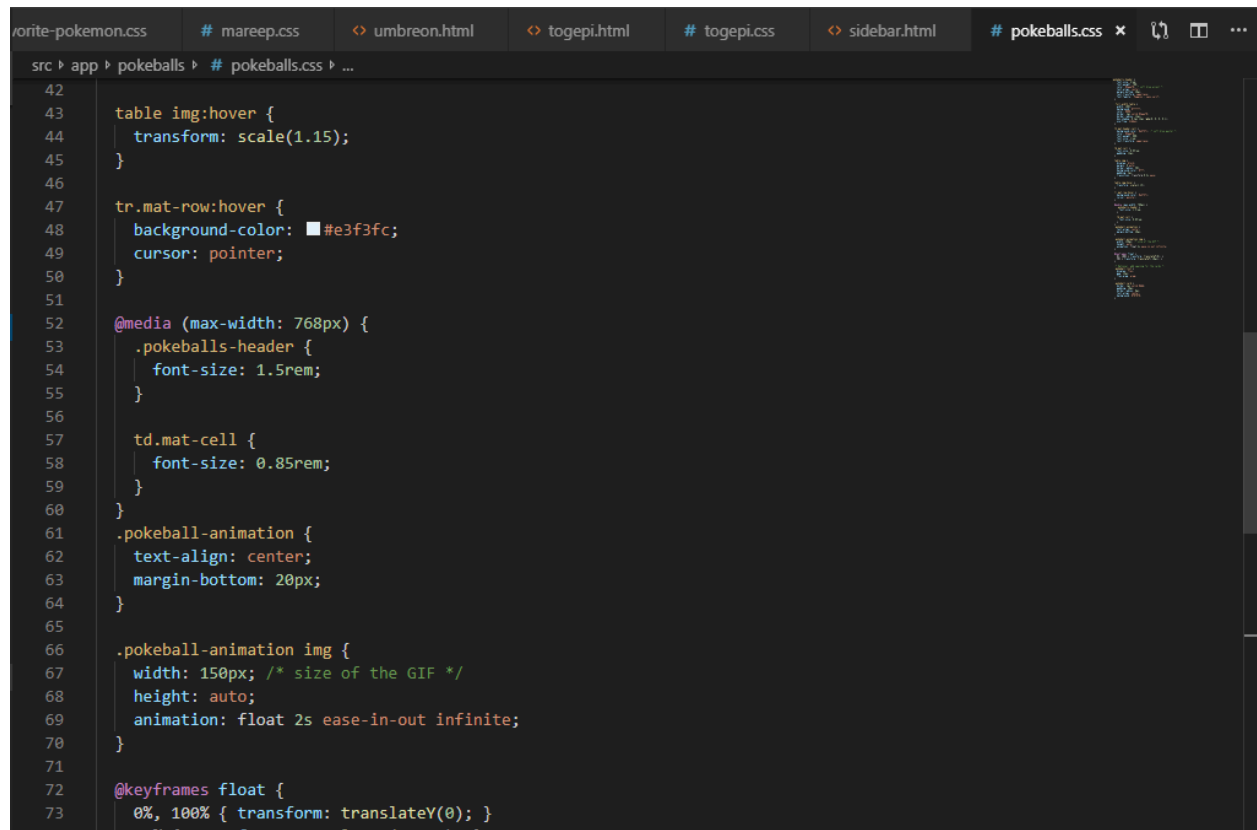
src > app > pokeballs > TS pokeballs.ts > ...
1  import { Component } from '@angular/core';
2  import { MatTableModule } from '@angular/material/table';
3  import { PokeballsService } from '../service/pokeballs';
4
5  @Component({
6    selector: 'app-pokeballs',
7    standalone: true,
8    imports: [MatTableModule],
9    templateUrl: './pokeballs.html',
10    styleUrls: ['./pokeballs.css']
11  })
12  export class Pokeballs {
13    dataSource: { name: string; type: string; image: string }[] = [];
14    displayedColumns: string[] = ['image', 'name', 'type'];
15
16    constructor(private p: PokeballsService) {}
17
18    ngOnInit(): void {
19      this.dataSource = this.p.getPokeballs();
20    }
21  }
22
```

```
pokeballs.html x TS app-routing-module.ts TS cyndaquil.ts TS favorite-pokemon.ts TS chikorita.ts cyndaquil.html
src > app > pokeballs > < pokeballs.html > ...
1 <h1 class="pokeballs-header"> Poké Balls Collection</h1>
2
3 <table mat-table [dataSource]="dataSource" class="mat-elevation-z8 full-width-table">
4
5   <ng-container matColumnDef="image">
6     <th mat-header-cell *matHeaderCellDef> Poké Ball </th>
7     <td mat-cell *matCellDef="let ball">
8       <img [src]="ball.image" alt="{{ ball.name }}" width="40" height="40">
9     </td>
10  </ng-container>
11
12  <ng-container matColumnDef="name">
13    <th mat-header-cell *matHeaderCellDef> Name </th>
14    <td mat-cell *matCellDef="let ball">{{ ball.name }}</td>
15  </ng-container>
16
17  <ng-container matColumnDef="type">
18    <th mat-header-cell *matHeaderCellDef> Type </th>
19    <td mat-cell *matCellDef="let ball">{{ ball.type }}</td>
20  </ng-container>
21
22  <tr mat-header-row *matHeaderRowDef="displayedColumns"></tr>
23  <tr mat-row *matRowDef="let row; columns: displayedColumns"></tr>
24 </table>
25
26
27
```



```
10
11 .full-width-table {
12   width: 100%;
13   background: #ffffff;
14   color: #000;
15   border: 2px solid #3aaef0;
16   border-radius: 12px;
17   box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
18   overflow: hidden;
19 }
20
21 th.mat-header-cell {
22   background-color: #e3f3fc; /* soft blue pastel */
23   color: #3aaef0;
24   font-weight: 600;
25   font-size: 1rem;
26   text-transform: uppercase;
27 }
28
29 td.mat-cell {
30   font-size: 0.95rem;
31   padding: 12px;
32 }
33
34 table img {
35   display: block;
36   margin: 0 auto;
37   border-radius: 50%;
38   background-color: #fff;
39   padding: 4px;
40   transition: transform 0.3s ease;
41 }
42
```





```
42
43 table img:hover {
44   transform: scale(1.15);
45 }
46
47 tr.mat-row:hover {
48   background-color: #e3f3fc;
49   cursor: pointer;
50 }
51
52 @media (max-width: 768px) {
53   .pokeballs-header {
54     font-size: 1.5rem;
55   }
56
57   td.mat-cell {
58     font-size: 0.85rem;
59   }
60 }
61
62 .pokeball-animation {
63   text-align: center;
64   margin-bottom: 20px;
65 }
66
67 .pokeball-animation img {
68   width: 150px; /* size of the GIF */
69   height: auto;
70   animation: float 2s ease-in-out infinite;
71 }
72
73 @keyframes float {
74   0%, 100% { transform: translateY(0); }
75   50% { transform: translateY(-50%); }
```