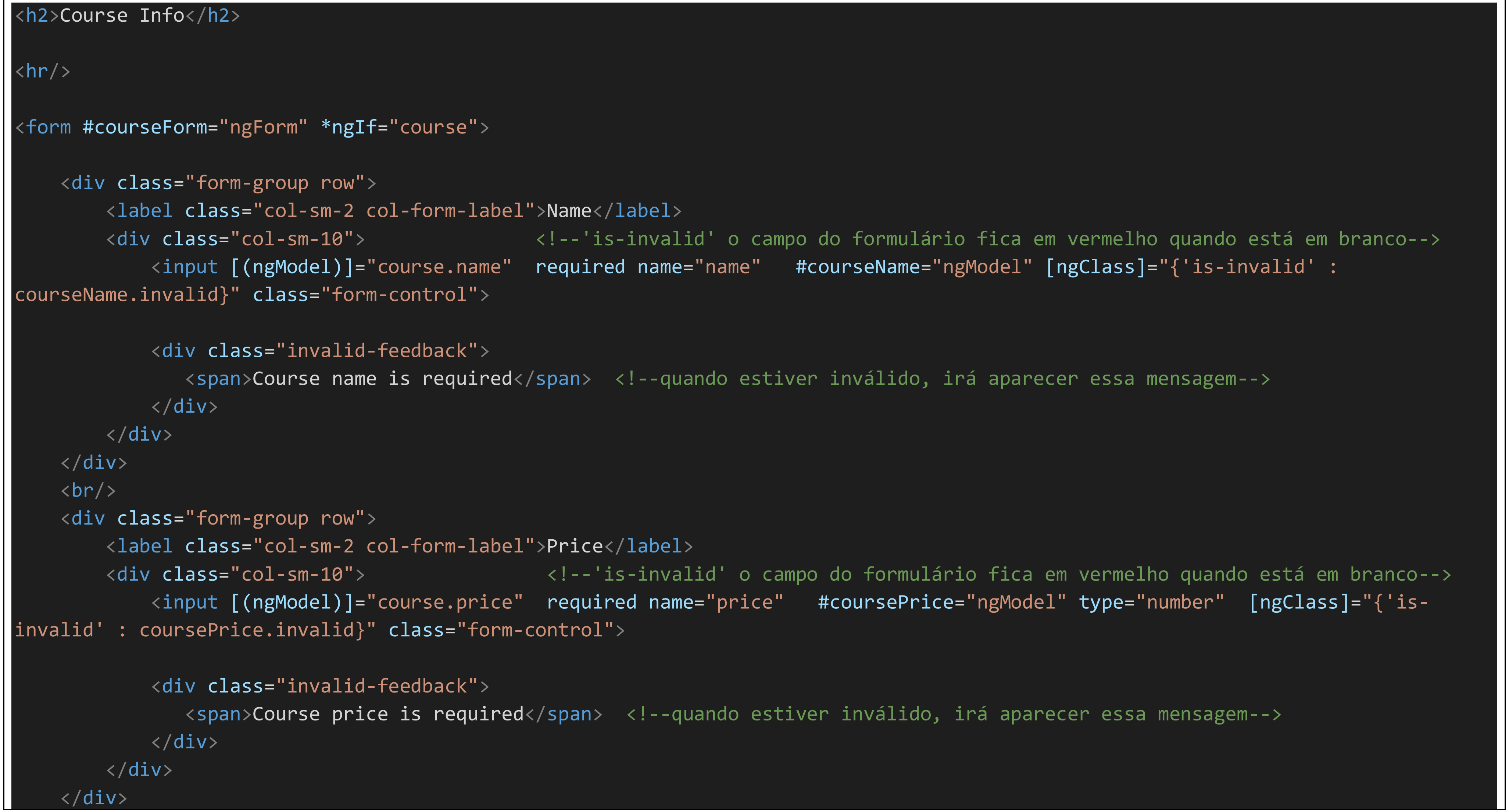
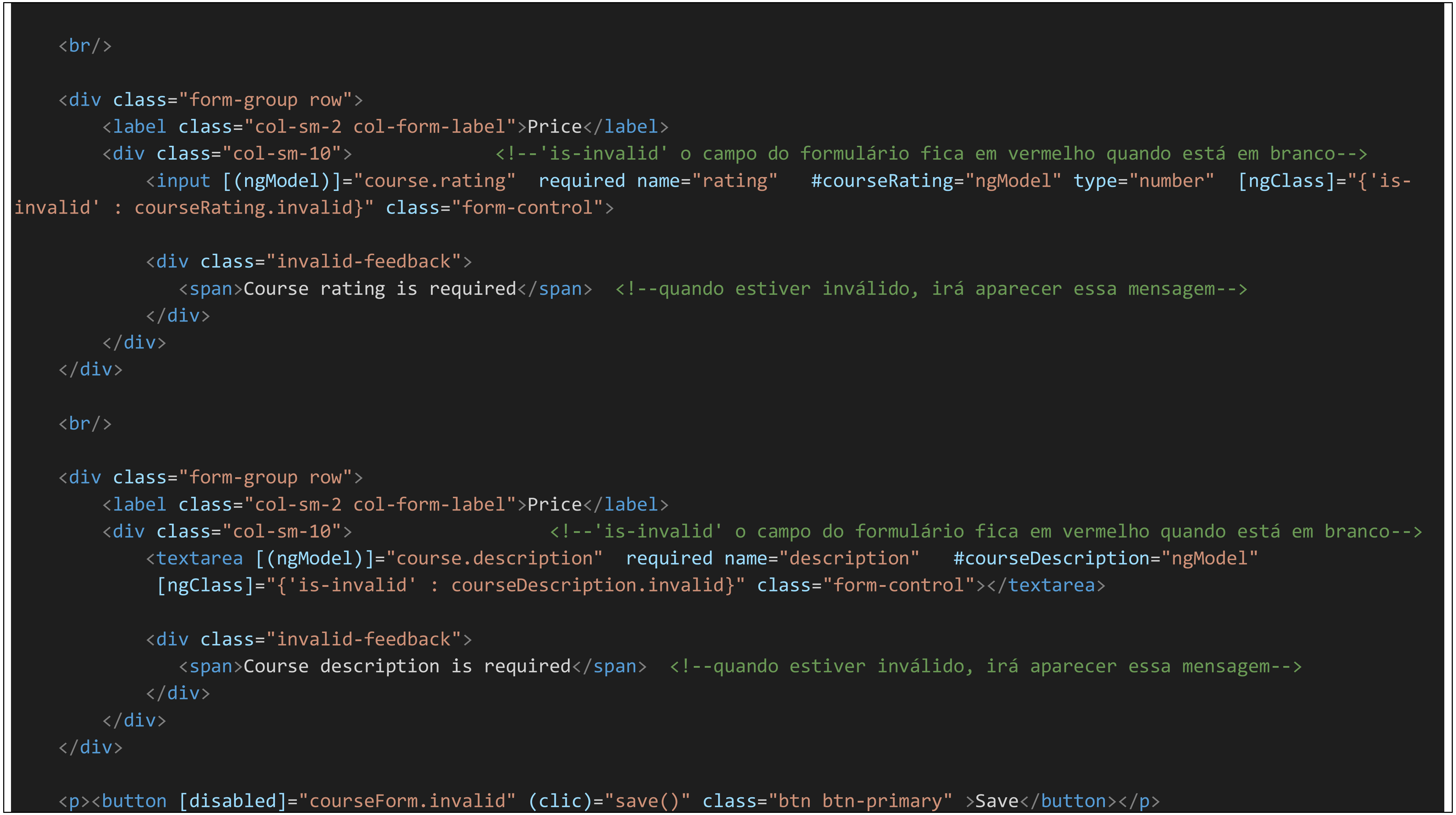
ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)

course-info.component.html



```html
<h2>Course Info</h2>

<hr/>

<form #courseForm="ngForm" *ngIf="course">

    <div class="form-group row">
        <label class="col-sm-2 col-form-label">Name</label>
        <div class="col-sm-10">              <!--'is-invalid' o campo do formulário fica em vermelho quando está em branco-->
            <input [(ngModel)]="course.name"  required name="name"   #courseName="ngModel" [ngClass]="{'is-invalid' : courseName.invalid}" class="form-control">

            <div class="invalid-feedback">
               <span>Course name is required</span>  <!--quando estiver inválido, irá aparecer essa mensagem-->
            </div>
        </div>
    </div>
    <br/>
    <div class="form-group row">
        <label class="col-sm-2 col-form-label">Price</label>
        <div class="col-sm-10">               <!--'is-invalid' o campo do formulário fica em vermelho quando está em branco-->
            <input [(ngModel)]="course.price"  required name="price"   #coursePrice="ngModel" type="number"  [ngClass]="{'is-invalid' : coursePrice.invalid}" class="form-control">

            <div class="invalid-feedback">
               <span>Course price is required</span>  <!--quando estiver inválido, irá aparecer essa mensagem-->
            </div>
        </div>
    </div>
```



```
    <br/>

    <div class="form-group row">
        <label class="col-sm-2 col-form-label">Price</label>
        <div class="col-sm-10">             <!--'is-invalid' o campo do formulário fica em vermelho quando está em branco-->
            <input [(ngModel)]="course.rating"  required name="rating"   #courseRating="ngModel" type="number"  [ngClass]="{'is-
invalid' : courseRating.invalid}" class="form-control">

            <div class="invalid-feedback">
               <span>Course rating is required</span>  <!--quando estiver inválido, irá aparecer essa mensagem-->
            </div>
        </div>
    </div>

    <br/>

    <div class="form-group row">
        <label class="col-sm-2 col-form-label">Price</label>
        <div class="col-sm-10">                  <!--'is-invalid' o campo do formulário fica em vermelho quando está em branco-->
            <textarea [(ngModel)]="course.description"  required name="description"   #courseDescription="ngModel"
             [ngClass]="{'is-invalid' : courseDescription.invalid}" class="form-control"></textarea>

            <div class="invalid-feedback">
               <span>Course description is required</span>  <!--quando estiver inválido, irá aparecer essa mensagem-->
            </div>
        </div>
    </div>

    <p><button [disabled]="courseForm.invalid" (clic)="save()" class="btn btn-primary" >Save</button></p>
```

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)

```
                <!--[disabled]="courseForm.invalid"   vai ser habilitado se todo campo do formulário estiver preenchido-->
    <button [routerLink]="['/courses']" class="btn btn-secondary" >Back</button>

</form>
```

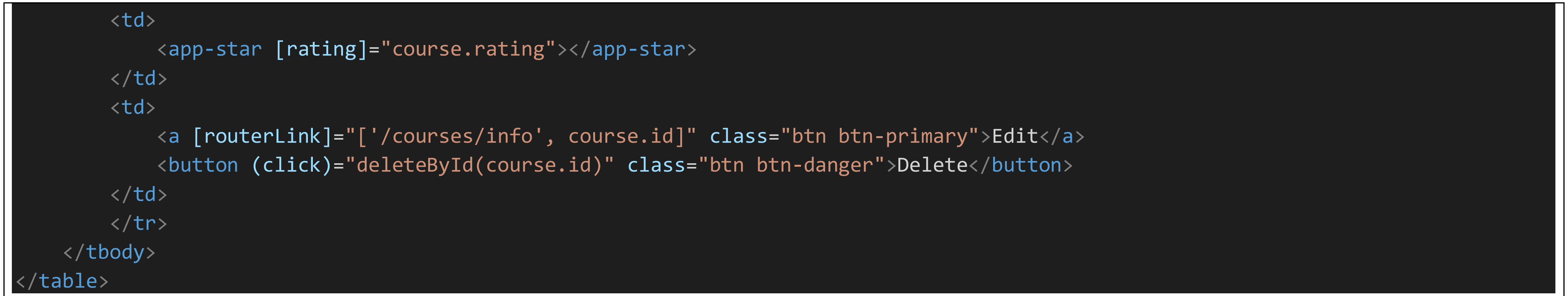
course-info.component.ts

```
@Component({
    templateUrl: './course-info.component.html'
})

export  class CourseInfoComponent implements OnInit{

    course: Course;

    constructor(private activatedRoute: ActivatedRoute, private courseService: CurseService){ }


    ngOnInit(): void {
        this.courseService.retrieveById(+this.activatedRoute.snapshot.paramMap.get('id')).subscribe({
            next: course => this.course = course,
            error: err => console.log('Error', err)
        });
    }

    save(): void {
        this.courseService.save(this.course).subscribe({
            next: course => console.log('Saved with sucess', course),
            error: err => console.log('Error', err)
        });
    }
}
```

course-list.component.html

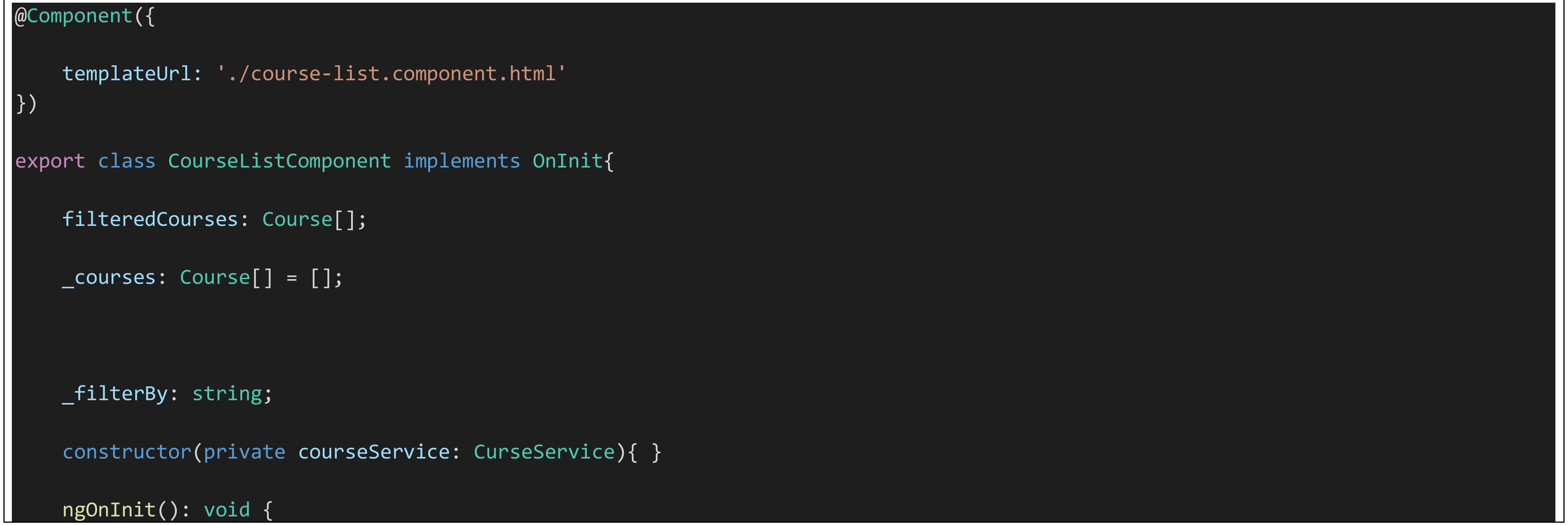
```
<h2>Course list</h2>

<div class="form-group row">
<label class="col-sm-2 col-form-label">Filter By</label>
<div class="col-sm-10">
    <input [(ngModel)]="filter" class="form-control">
</div>
</div>
<table class="table table-striped">
    <thead>
        <tr>
        <th>Image</th>
        <th>Name</th>
        <th>Price</th>
        <th>Code</th>
        <th>Release Date</th>
        <th>Rating</th>
        <th>Options</th>
        </tr>
    </thead>
    <tbody>
        <tr *ngFor="let course of filteredCourses">
        <td><img [src]="course.imageUrl" width="40" height="40"></td>
        <td>{{course.name}}</td>
        <td>{{course.price}}</td>
        <td>{{course.code | lowercase | replace: '-': ' '}}</td>
        <td>{{course.releaseDate | date: 'dd/MM/yyyy'}}</td>
```

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)

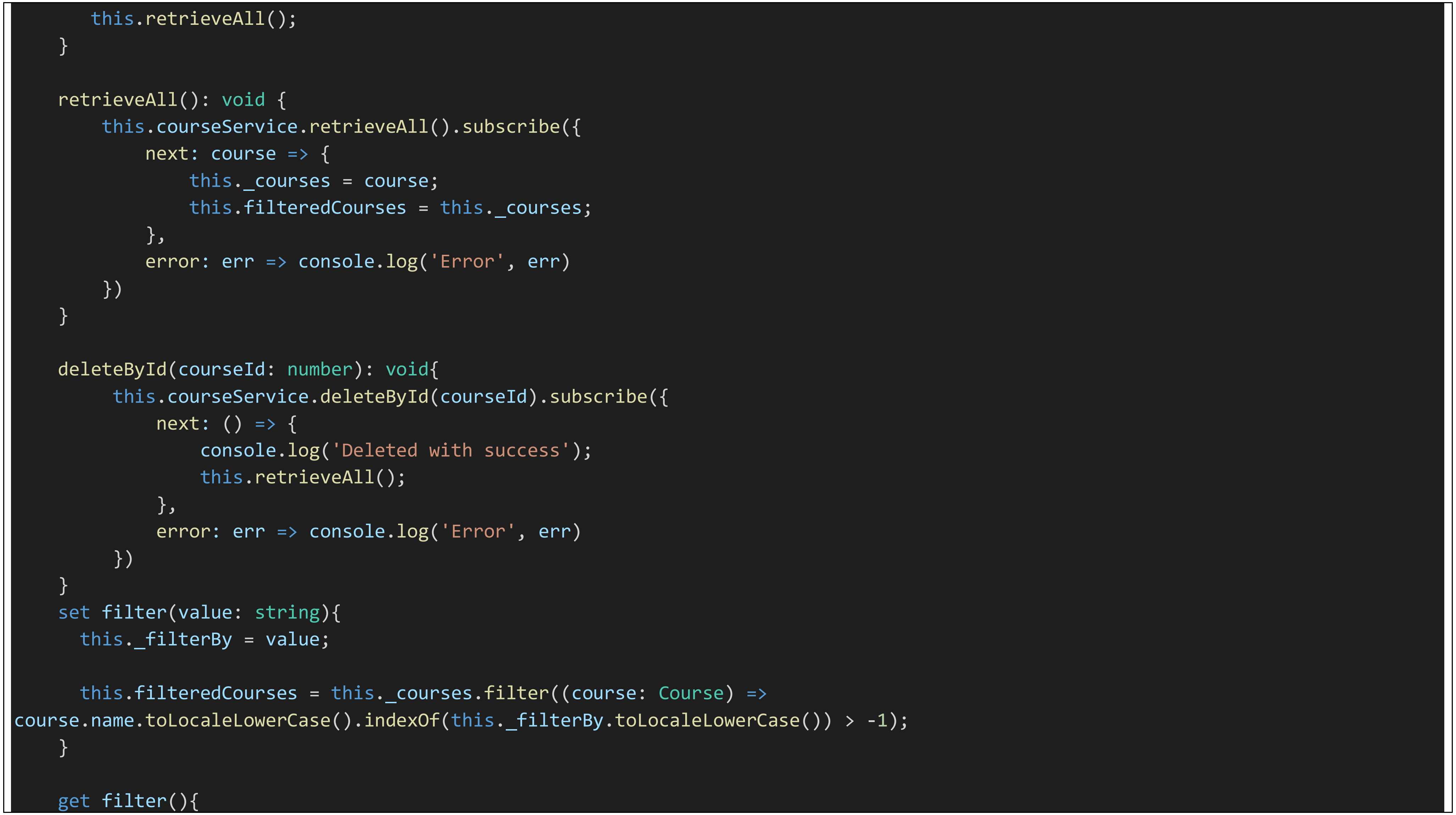


```
        <td>
            <app-star [rating]="course.rating"></app-star>
        </td>
        <td>
            <a [routerLink]="['/courses/info', course.id]" class="btn btn-primary">Edit</a>
            <button (click)="deleteById(course.id)" class="btn btn-danger">Delete</button>
        </td>
        </tr>
    </tbody>
</table>
```

course0list.component.ts



```
@Component({

    templateUrl: './course-list.component.html'
})

export class CourseListComponent implements OnInit{

    filteredCourses: Course[];

    _courses: Course[] = [];



    _filterBy: string;

    constructor(private courseService: CurseService){ }

    ngOnInit(): void {
```



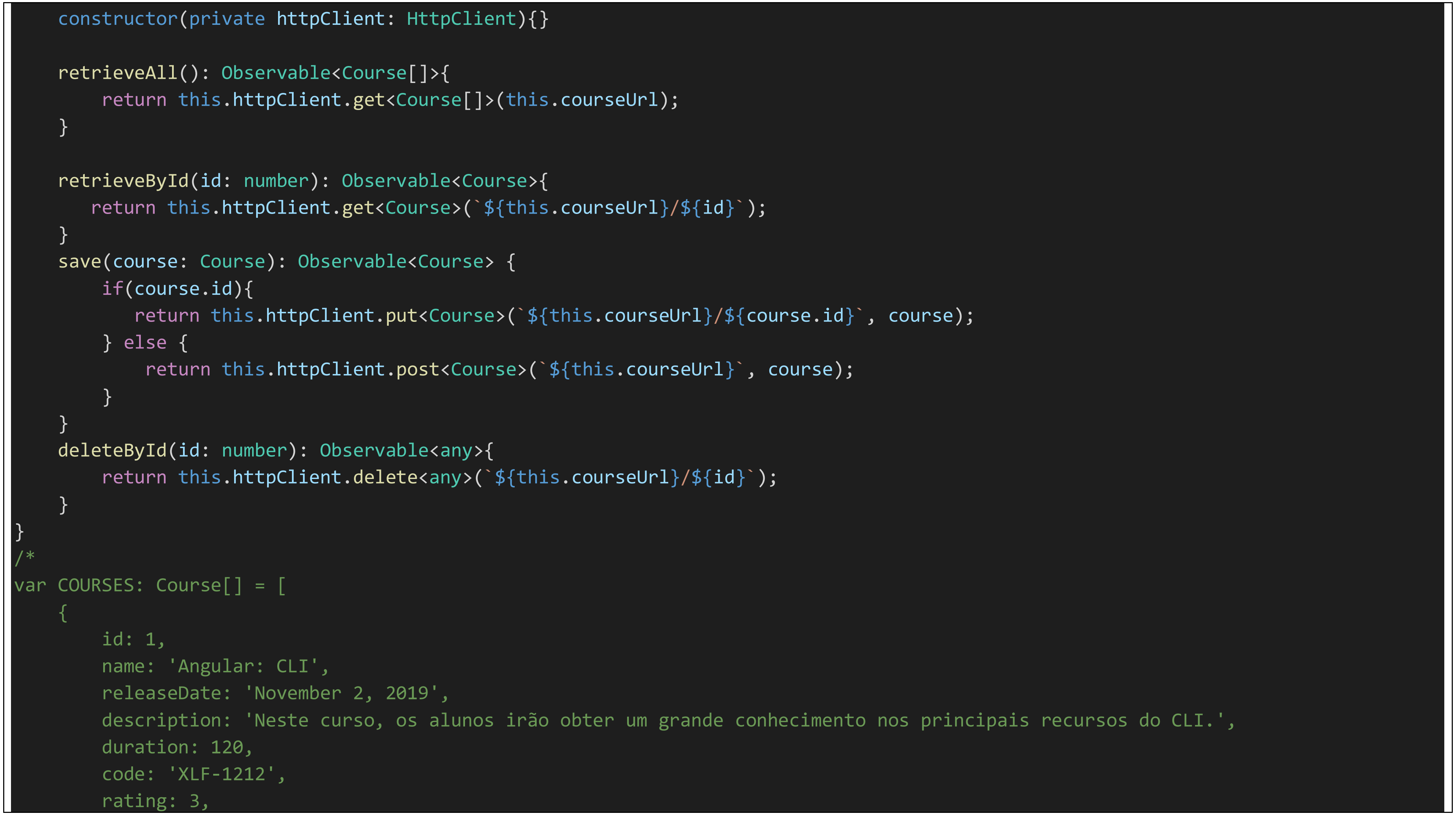
```
       this.retrieveAll();
    }

    retrieveAll(): void {
        this.courseService.retrieveAll().subscribe({
            next: course => {
                this._courses = course;
                this.filteredCourses = this._courses;
            },
            error: err => console.log('Error', err)
        })
    }

    deleteById(courseId: number): void{
         this.courseService.deleteById(courseId).subscribe({
             next: () => {
                 console.log('Deleted with success');
                 this.retrieveAll();
             },
             error: err => console.log('Error', err)
         })
    }
    set filter(value: string){
      this._filterBy = value;

      this.filteredCourses = this._courses.filter((course: Course) =>
course.name.toLocaleLowerCase().indexOf(this._filterBy.toLocaleLowerCase()) > -1);
    }

    get filter(){
        return this._filterBy;
    }
}
```

course.ts

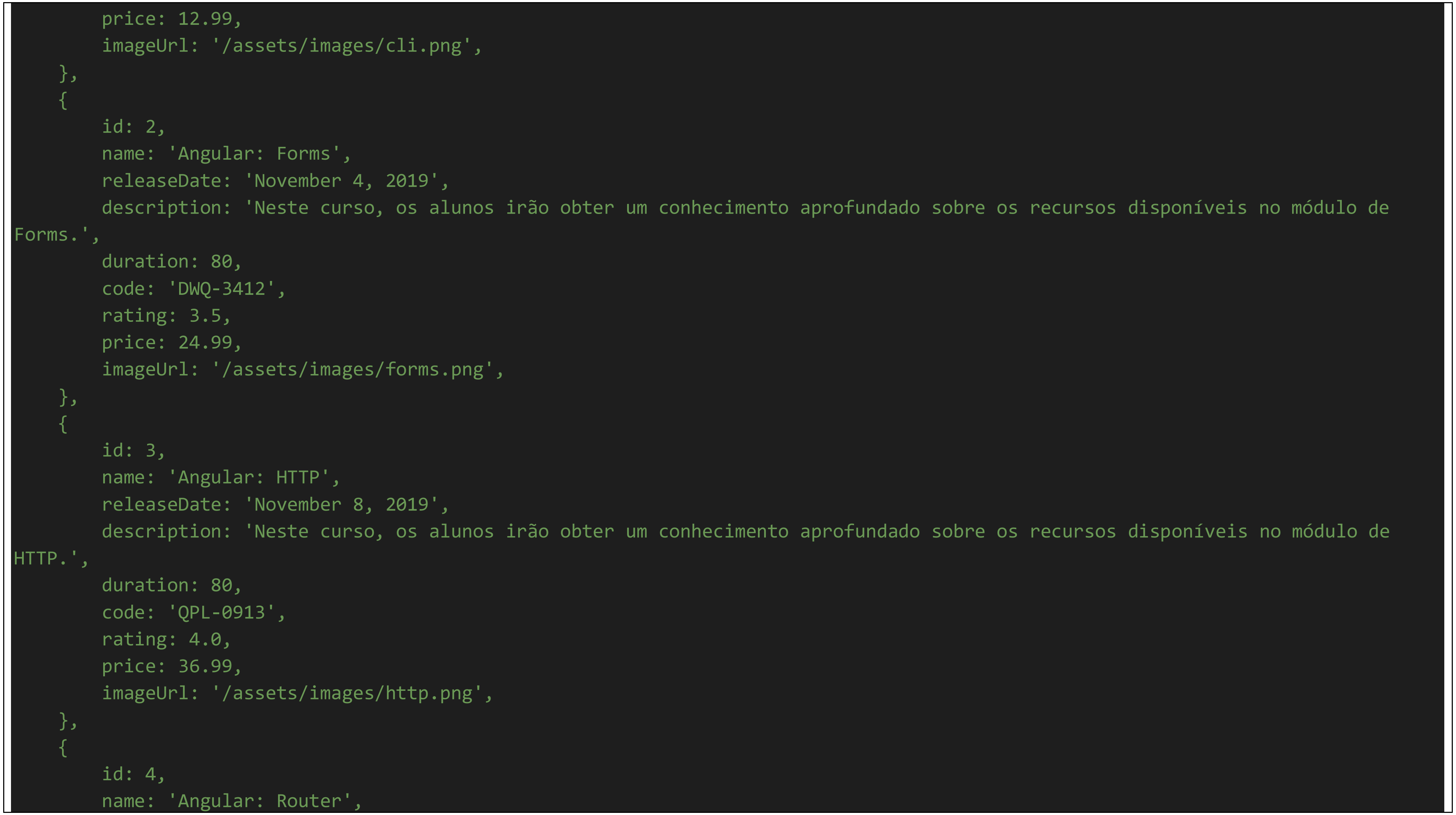
```
export class Course{
    id: number;
    name: string;
    imageUrl: String;
    price: number;
    code: string;
    duration: number;
    rating: number;
    releaseDate: string;
    description: string;
}
```

curse.service.ts

```
import { HttpClient } from "@angular/common/http";
import { Injectable } from "@angular/core";
import { Observable } from "rxjs";
import { Course } from "./course";

@Injectable({
    providedIn: 'root'
})

export class CurseService{
    private courseUrl: string = 'http://localhost:3100/api/courses';
```



```
    constructor(private httpClient: HttpClient){}

    retrieveAll(): Observable<Course[]>{
        return this.httpClient.get<Course[]>(this.courseUrl);
    }

    retrieveById(id: number): Observable<Course>{
       return this.httpClient.get<Course>(`${this.courseUrl}/${id}`);
    }
    save(course: Course): Observable<Course> {
        if(course.id){
           return this.httpClient.put<Course>(`${this.courseUrl}/${course.id}`, course);
        } else {
            return this.httpClient.post<Course>(`${this.courseUrl}`, course);
        }
    }
    deleteById(id: number): Observable<any>{
        return this.httpClient.delete<any>(`${this.courseUrl}/${id}`);
    }
}
/*
var COURSES: Course[] = [
    {
        id: 1,
        name: 'Angular: CLI',
        releaseDate: 'November 2, 2019',
        description: 'Neste curso, os alunos irão obter um grande conhecimento nos principais recursos do CLI.',
        duration: 120,
        code: 'XLF-1212',
        rating: 3,
```



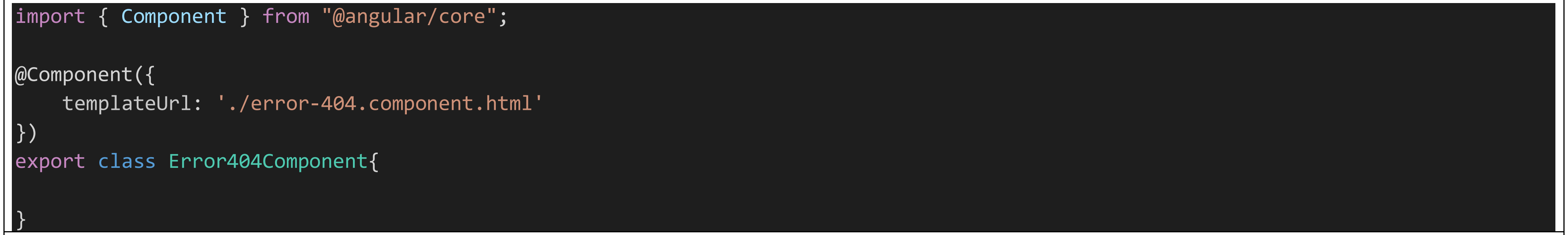
```
        price: 12.99,
        imageUrl: '/assets/images/cli.png',
    },
    {
        id: 2,
        name: 'Angular: Forms',
        releaseDate: 'November 4, 2019',
        description: 'Neste curso, os alunos irão obter um conhecimento aprofundado sobre os recursos disponíveis no módulo de
Forms.',
        duration: 80,
        code: 'DWQ-3412',
        rating: 3.5,
        price: 24.99,
        imageUrl: '/assets/images/forms.png',
    },
    {
        id: 3,
        name: 'Angular: HTTP',
        releaseDate: 'November 8, 2019',
        description: 'Neste curso, os alunos irão obter um conhecimento aprofundado sobre os recursos disponíveis no módulo de
HTTP.',
        duration: 80,
        code: 'QPL-0913',
        rating: 4.0,
        price: 36.99,
        imageUrl: '/assets/images/http.png',
    },
    {
        id: 4,
        name: 'Angular: Router',
        releaseDate: 'November 16, 2019',
        description: 'Neste curso, os alunos irão obter um conhecimento aprofundado sobre os recursos disponíveis no módulo de Router.',
        duration: 80,
        code: 'OHP-1095',
        rating: 4.5,
        price: 46.99,
        imageUrl: '/assets/images/router.png',
    },
    {
        id: 5,
        name: 'Angular: Animations',
        releaseDate: 'November 25, 2019',
        description: 'Neste curso, os alunos irão obter um conhecimento aprofundado sobre os recursos disponíveis sobre Animation.',
        duration: 80,
        code: 'PWY-9381',
        rating: 5,
        price: 56.99,
        imageUrl: '/assets/images/animations.png',
    }
];  */
```

error-404.component.html

```
<h2>Error 404</h2>
```

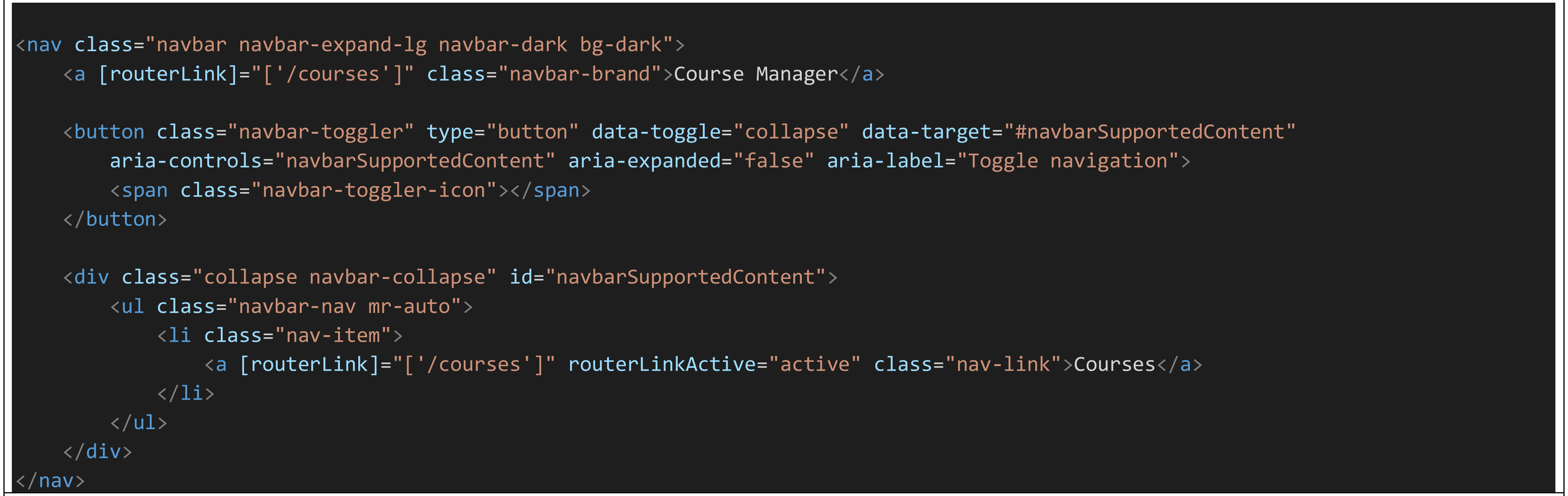
error-404.component.ts

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)



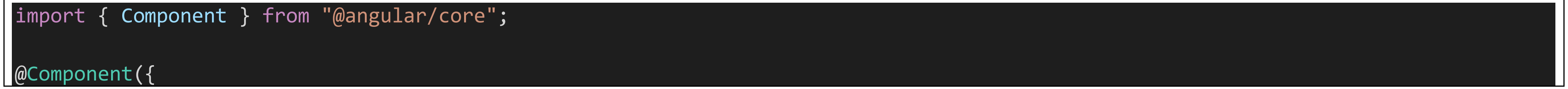
```
import { Component } from "@angular/core";

@Component({
    templateUrl: './error-404.component.html'
})
export class Error404Component{

}
```

nav-bar.component.html



```

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <a [routerLink]="['/courses']" class="navbar-brand">Course Manager</a>

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
        aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav mr-auto">
            <li class="nav-item">
                <a [routerLink]="['/courses']" routerLinkActive="active" class="nav-link">Courses</a>
            </li>
        </ul>
    </div>
</nav>
```

nav-bar.component.ts



```
import { Component } from "@angular/core";

@Component({
```

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)

```
    selector: 'app-nav-bar',
    templateUrl: './nav-bar.component.html'
})
export class NavBarComponent{

}
```

replace.pipe.ts

```
import { Pipe, PipeTransform  } from "@angular/core";

@Pipe({
    name: 'replace' /* nome do pipe que será utilizado no html */
})

export class ReplacePipe implements PipeTransform{

    transform(value: string, char: string, valueToReplace: string) {
        return value.replace(char, valueToReplace);
    }
}
```

star.component.css

```
.crop{
    overflow: hidden;
}
```

star.component.html

```
<div class="crop" [style.width.px]="starWidth" [title]="rating">
<div style="width: 100px;">
    <span class="fa fa-star"></span>
    <span class="fa fa-star"></span>
    <span class="fa fa-star"></span>
    <span class="fa fa-star"></span>
    <span class="fa fa-star"></span>
</div>
</div>
```

star.component.ts

```
@Component({
    selector: 'app-star',
    templateUrl:'./star.component.html',
    styleUrls: ['./star.component.css']
})
export class StarComponent implements OnChanges{

    @Input()
    rating: number = 0;

    starWidth: number;

    ngOnChanges(): void {
        this.starWidth = this.rating * 74 / 5;

    }
}
```

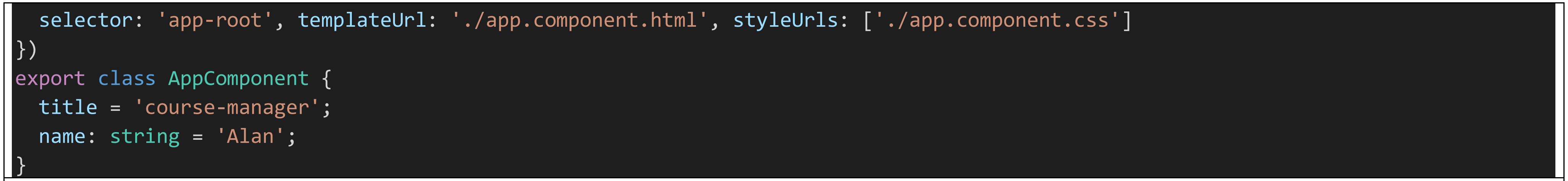
app.component.html

```
<app-nav-bar></app-nav-bar>

<div class="container nt-4">
  <router-outlet></router-outlet>
</div>
```

app.component.ts

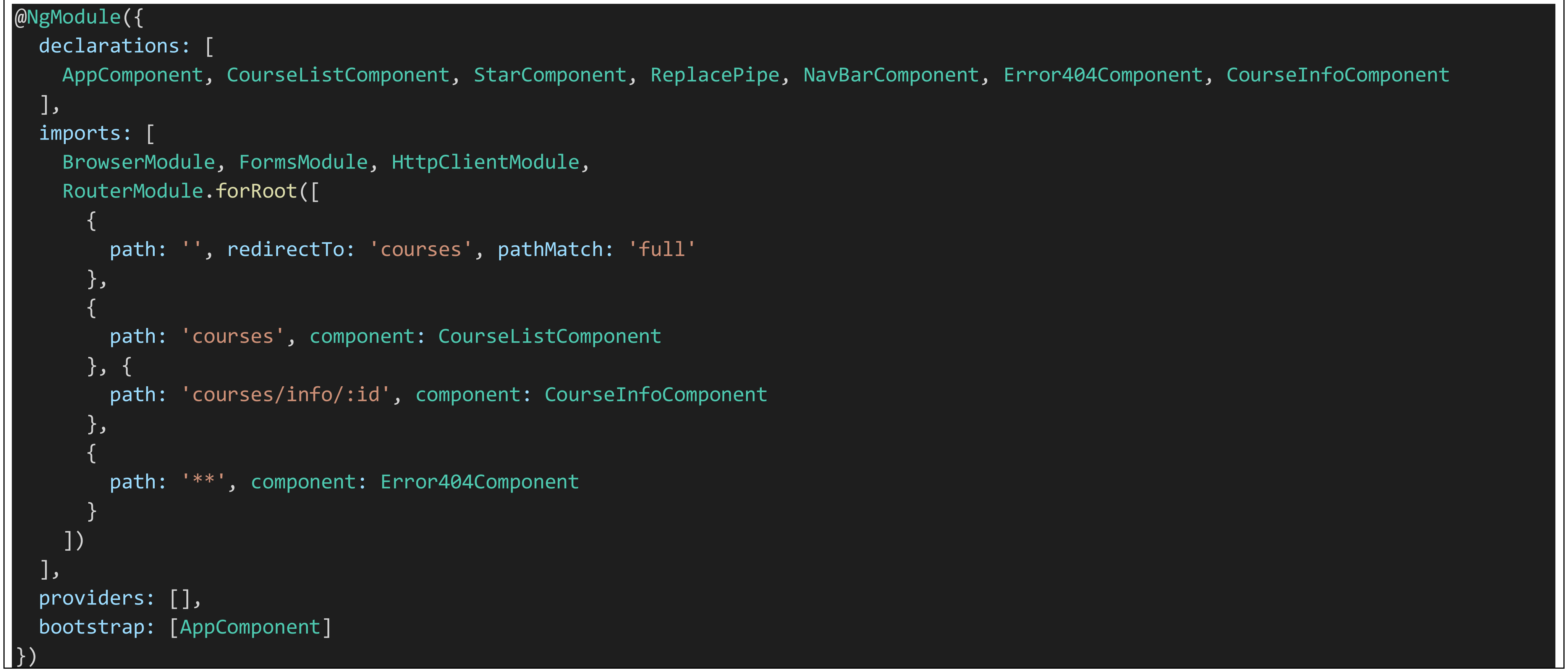
```

@Component({
```

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)



```
  selector: 'app-root', templateUrl: './app.component.html', styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'course-manager';
  name: string = 'Alan';
}
```

app.module.ts



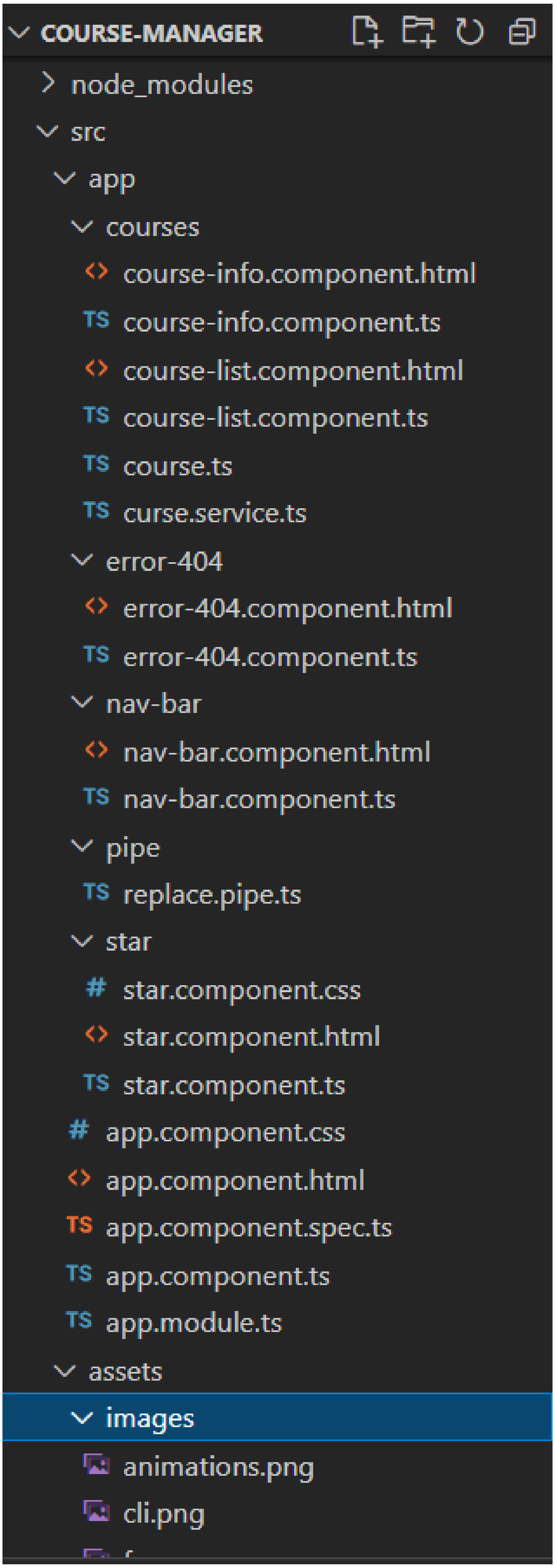
```
@NgModule({
  declarations: [
    AppComponent, CourseListComponent, StarComponent, ReplacePipe, NavBarComponent, Error404Component, CourseInfoComponent
  ],
  imports: [
    BrowserModule, FormsModule, HttpClientModule,
    RouterModule.forRoot([
      {
        path: '', redirectTo: 'courses', pathMatch: 'full'
      },
      {
        path: 'courses', component: CourseListComponent
      }, {
        path: 'courses/info/:id', component: CourseInfoComponent
      },
      {
        path: '**', component: Error404Component
      }
    ])
  ],
  providers: [],
  bootstrap: [AppComponent]
})
```

```
export class AppModule { }
```

angular.json

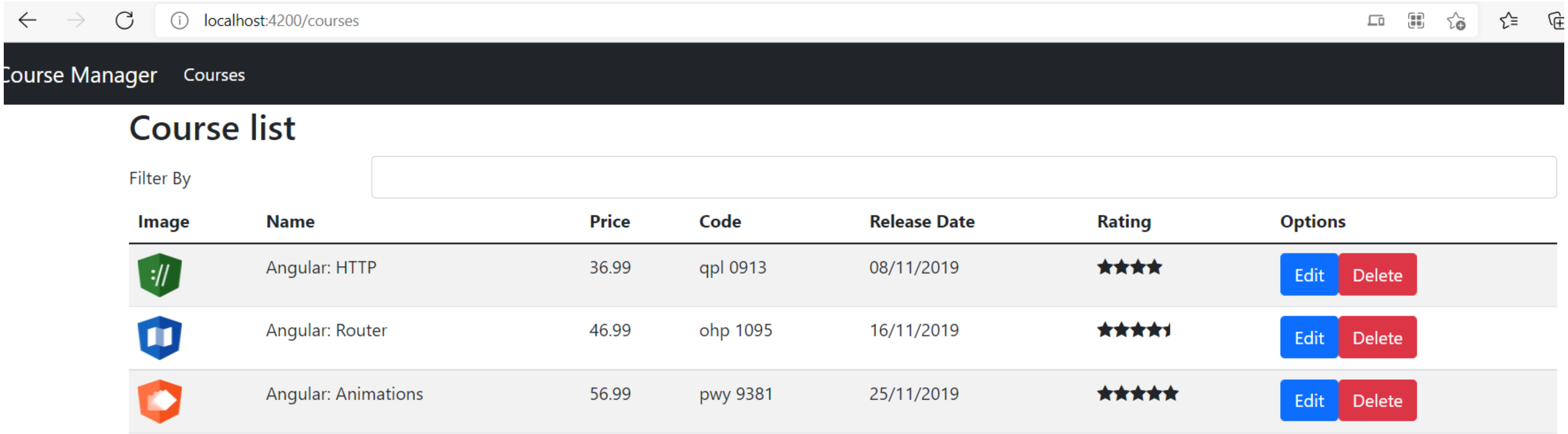
```
 "styles": [
              "src/styles.css",
              "node_modules/bootstrap/dist/css/bootstrap.min.css",
              "node_modules/font-awesome/css/font-awesome.min.css"
            ],
```

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)



```
∨ COURSE-MANAGER
  > node_modules
  ∨ src
    ∨ app
      ∨ courses
        <> course-info.component.html
        TS course-info.component.ts
        <> course-list.component.html
        TS course-list.component.ts
        TS course.ts
        TS curse.service.ts
      ∨ error-404
        <> error-404.component.html
        TS error-404.component.ts
      ∨ nav-bar
        <> nav-bar.component.html
        TS nav-bar.component.ts
      ∨ pipe
        TS replace.pipe.ts
      ∨ star
        #  star.component.css
        <> star.component.html
        TS star.component.ts
      #  app.component.css
      <> app.component.html
      TS app.component.spec.ts
      TS app.component.ts
      TS app.module.ts
    ∨ assets
      ∨ images
          animations.png
          cli.png
```

ANGULAR 8: Parte 1, Parte 2 e Parte 3 (até o item 2)



localhost:4200/courses

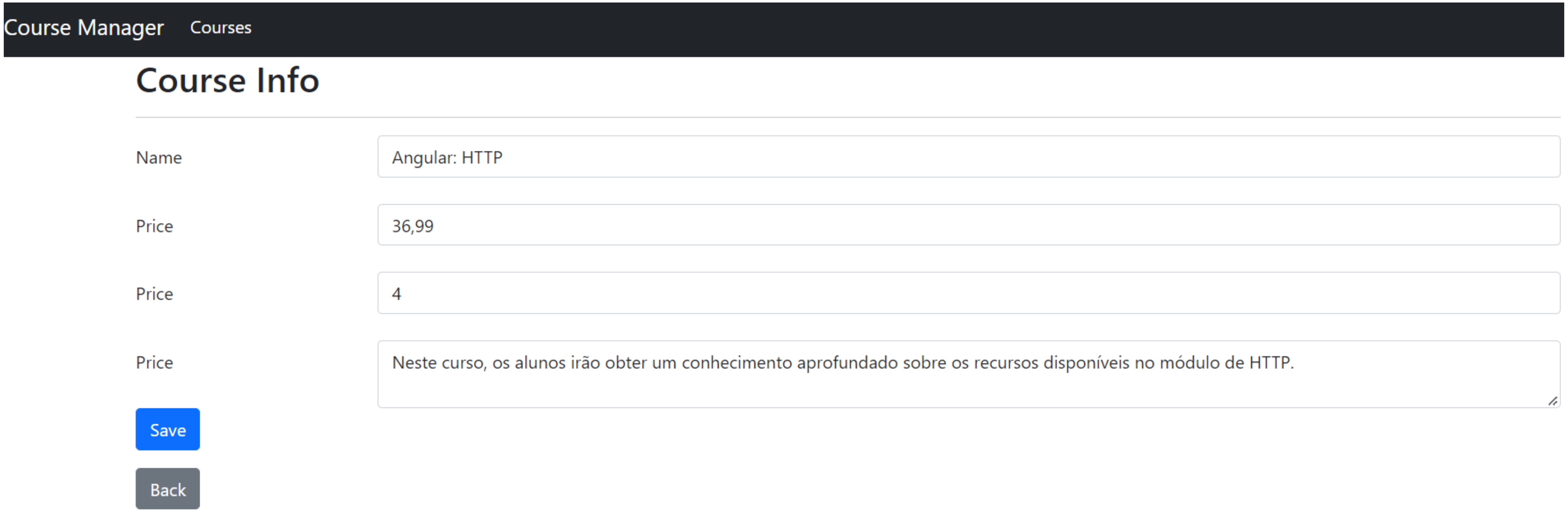
ourse Manager Courses

Course list

Filter By

Image	Name	Price	Code	Release Date	Rating	Options
	Angular: HTTP	36.99	qpl 0913	08/11/2019	★★★★	Edit Delete
	Angular: Router	46.99	ohp 1095	16/11/2019	★★★★⯪	Edit Delete
	Angular: Animations	56.99	pwy 9381	25/11/2019	★★★★★	Edit Delete

Deletado 2



Course Manager Courses

Course Info

Name	Angular: HTTP
Price	36,99
Price	4
Price	Neste curso, os alunos irão obter um conhecimento aprofundado sobre os recursos disponíveis no módulo de HTTP.