Apuntes open:

--routing
npm install -> siempre

ng update -> para visualizar versión de angular

ng update @angular/core @angular/cli -> actualizer angular

ng add @angular/material

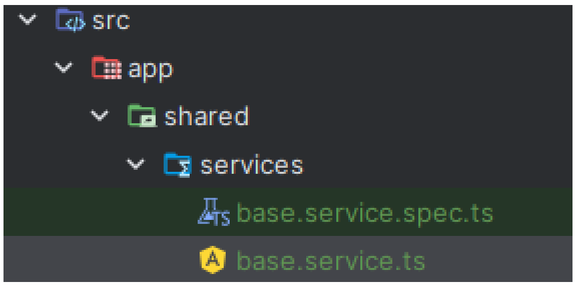
npm i json-server

cd server
json-server –watch db.json –routes routes.json

Ruta 1: (una entidad)

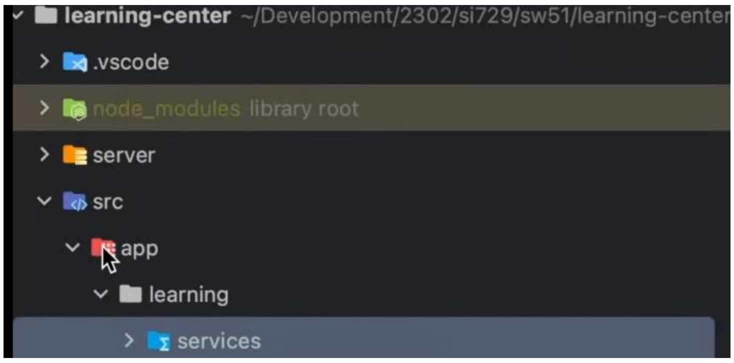
Primero crear carpeta server con db y routes y meterle datos

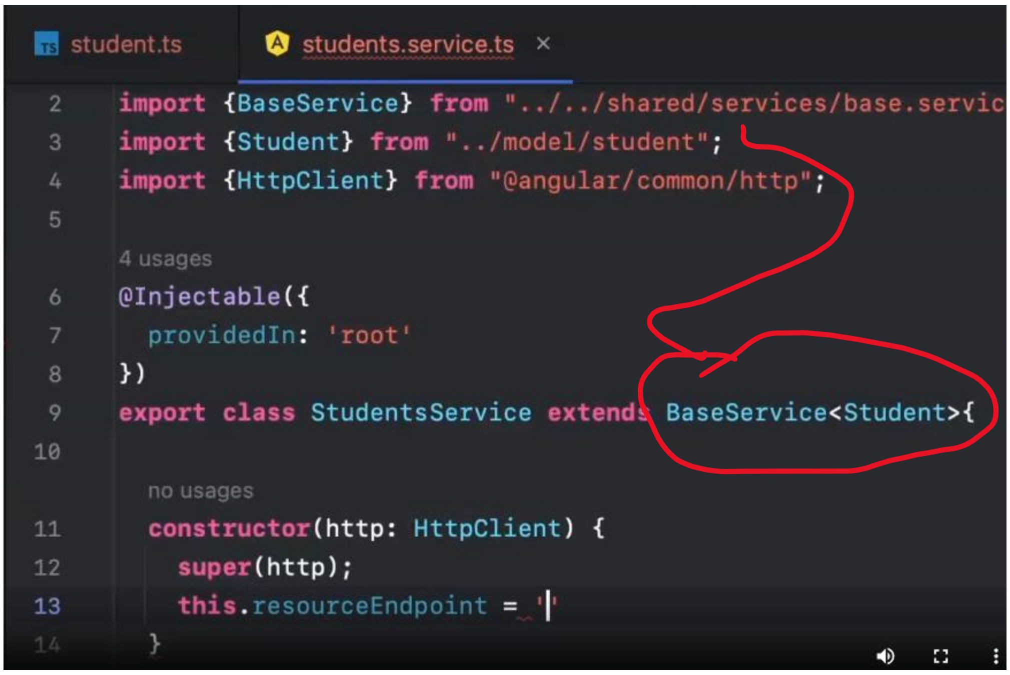
Crear como clase:



Luego hacer los models

Luego crear services


learning-center ~/Development/2302/si729/sw51/learning-center
.vscode
node_modules library root
server
src
app
learning
services

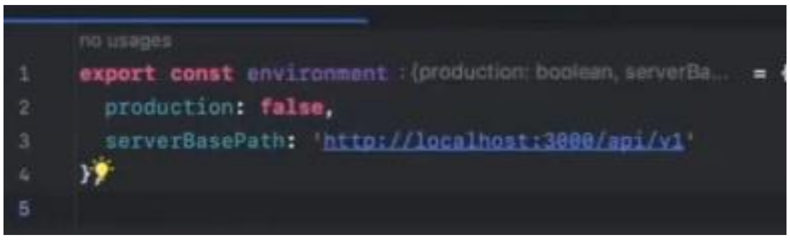


```
import {BaseService} from "../../shared/services/base.servic
import {Student} from "../model/student";
import {HttpClient} from "@angular/common/http";

4 usages
@Injectable({
  providedIn: 'root'
})
export class StudentsService extends BaseService<Student>{

  no usages
  constructor(http: HttpClient) {
    super(http);
    this.resourceEndpoint = ''
  }
```

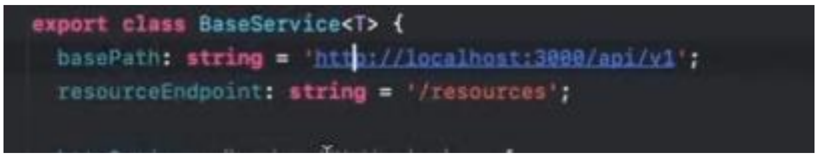
Luego hacer los componentes

Luego hacer routes

Hacer el environment y cambiarlo en el localhost dentro de base.service

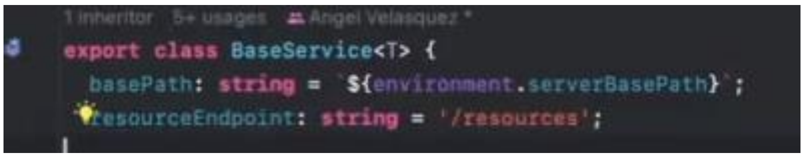


```
no usages
export const environment :{production: boolean, serverBa...  = {
  production: false,
  serverBasePath: 'http://localhost:3000/api/v1'
}
```



```
export class BaseService<T> {
  basePath: string = 'http://localhost:3000/api/v1';
  resourceEndpoint: string = '/resources';
```

A



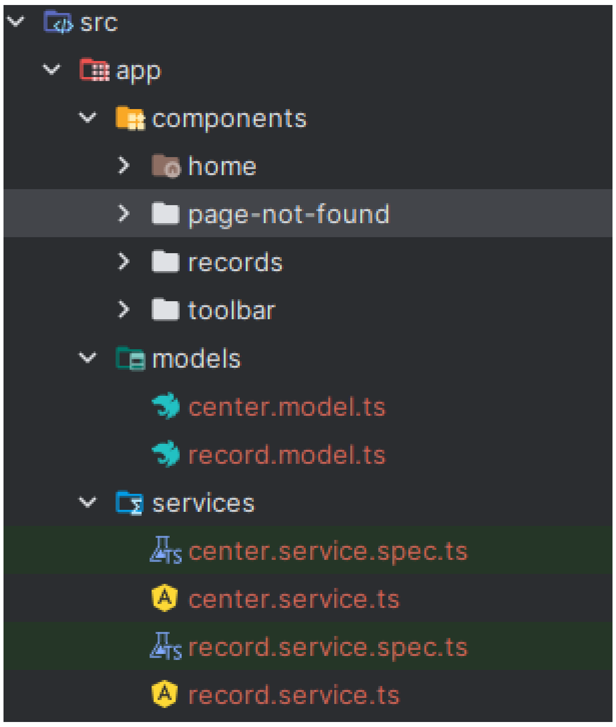
```
1 inheritor  5+ usages  Angel Velasquez *
export class BaseService<T> {
  basePath: string = `${environment.serverBasePath}`;
  resourceEndpoint: string = '/resources';
```

Ruta 2: (dos entidades)

Primero crear carpeta server con db y routes y meterle datos

Crear los models

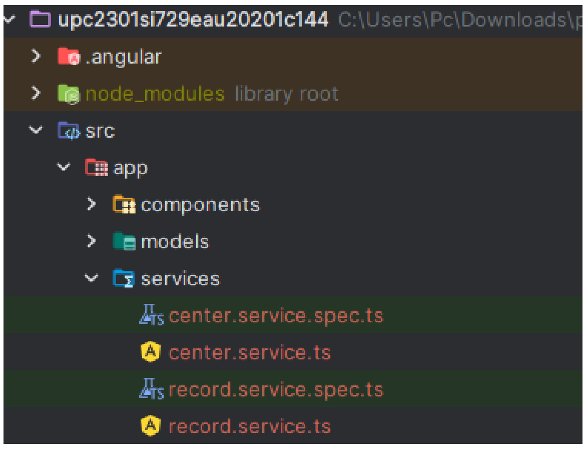
Ng g cl



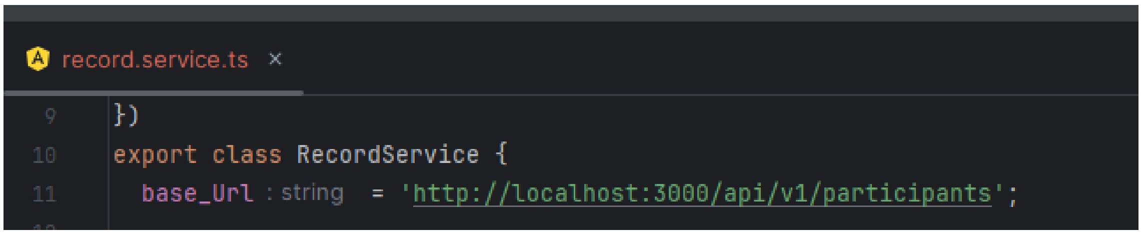
```
src
  app
    components
      home
      page-not-found
      records
      toolbar
    models
      center.model.ts
      record.model.ts
    services
      center.service.spec.ts
      center.service.ts
      record.service.spec.ts
      record.service.ts
```

Crear los services como services

Ng g s

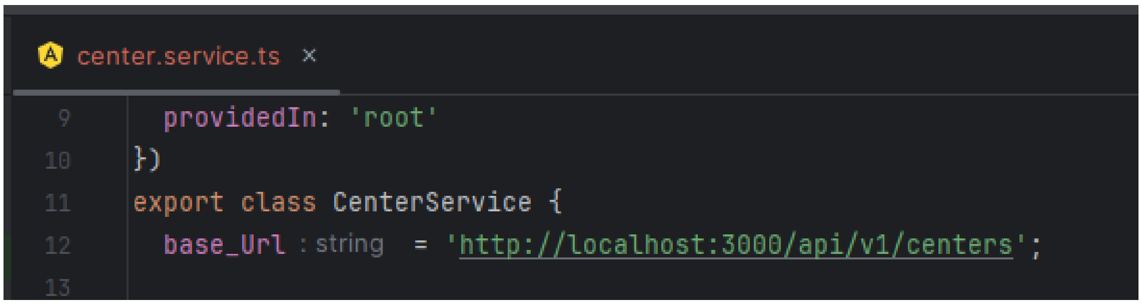


```
upc2301si729eau20201c144  C:\Users\Pc\Downloads\p
  .angular
  node_modules  library root
  src
    app
      components
      models
      services
        center.service.spec.ts
        center.service.ts
        record.service.spec.ts
        record.service.ts
```



record.service.ts ×

```
})
export class RecordService {
  base_Url : string  = 'http://localhost:3000/api/v1/participants';
```

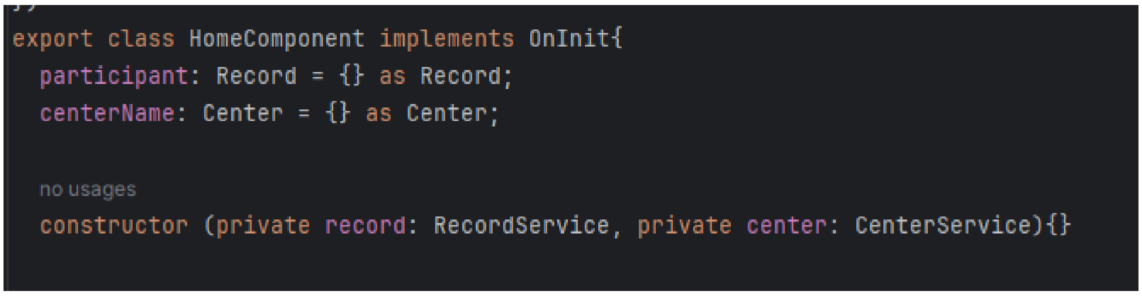


```
center.service.ts ×
  providedIn: 'root'
})
export class CenterService {
  base_Url : string  = 'http://localhost:3000/api/v1/centers';
```

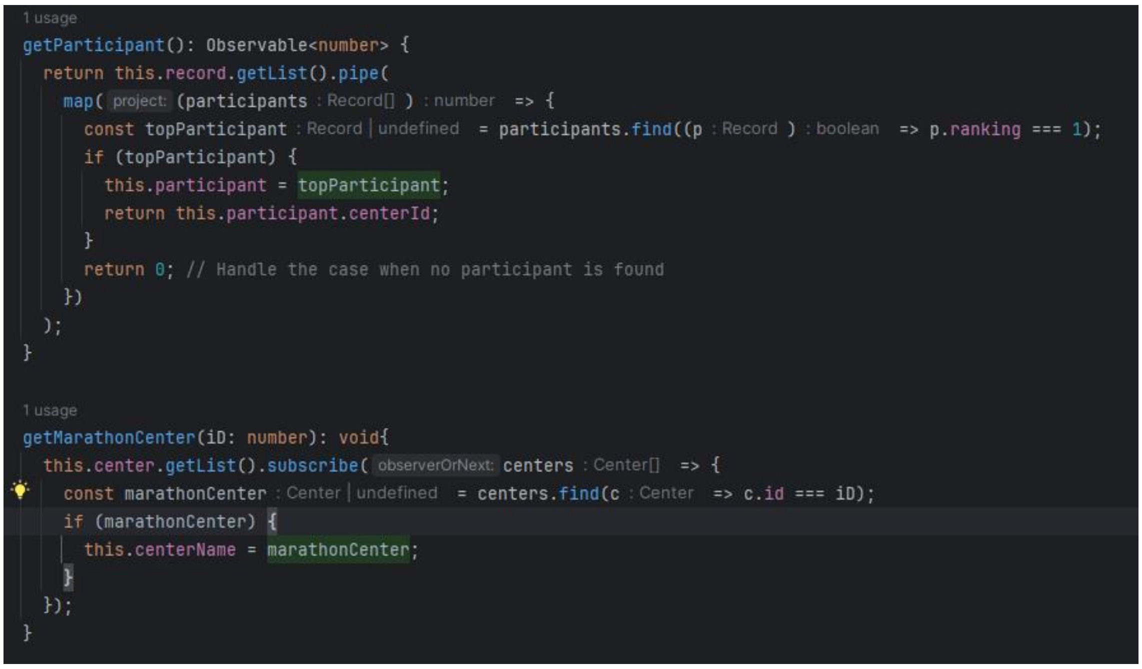
Crear los components

Ng g c

En home:

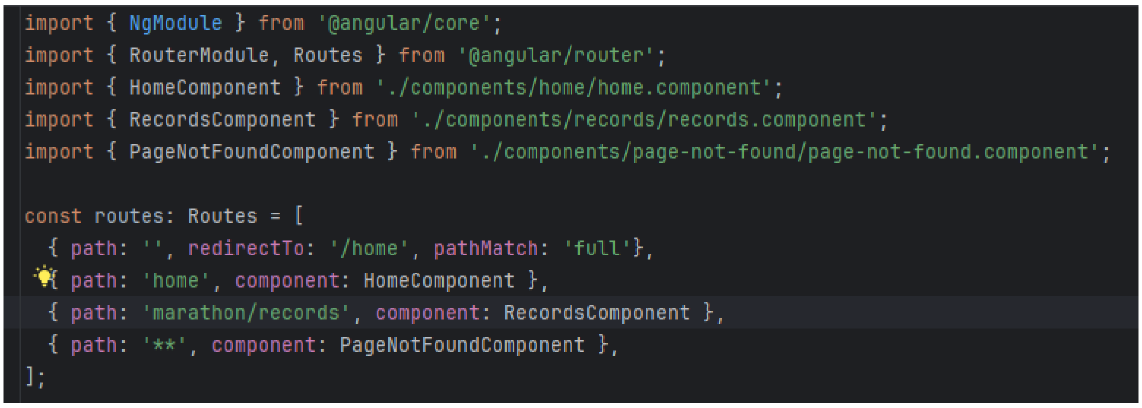


```
export class HomeComponent implements OnInit{
  participant: Record = {} as Record;
  centerName: Center = {} as Center;

  no usages
  constructor (private record: RecordService, private center: CenterService){}
```



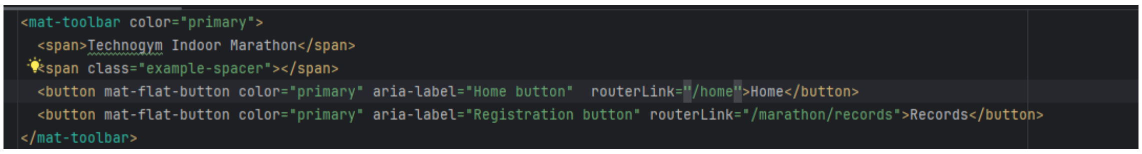
```
1 usage
getParticipant(): Observable<number> {
  return this.record.getList().pipe(
    map( project: (participants : Record[] ) : number  => {
      const topParticipant : Record | undefined  = participants.find((p : Record ) : boolean  => p.ranking === 1);
      if (topParticipant) {
        this.participant = topParticipant;
        return this.participant.centerId;
      }
      return 0; // Handle the case when no participant is found
    })
  );
}

1 usage
getMarathonCenter(iD: number): void{
  this.center.getList().subscribe( observerOrNext: centers : Center[]  => {
    const marathonCenter : Center | undefined  = centers.find(c : Center  => c.id === iD);
    if (marathonCenter) {
      this.centerName = marathonCenter;
    }
  });
}
```

Routeo:



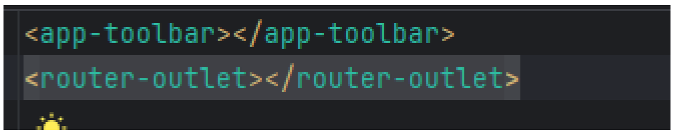
```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { HomeComponent } from './components/home/home.component';
import { RecordsComponent } from './components/records/records.component';
import { PageNotFoundComponent } from './components/page-not-found/page-not-found.component';

const routes: Routes = [
  { path: '', redirectTo: '/home', pathMatch: 'full'},
  { path: 'home', component: HomeComponent },
  { path: 'marathon/records', component: RecordsComponent },
  { path: '**', component: PageNotFoundComponent },
];
```

En toolbar:



```
<mat-toolbar color="primary">
  <span>Technogym Indoor Marathon</span>
  <span class="example-spacer"></span>
  <button mat-flat-button color="primary" aria-label="Home button"  routerLink="/home">Home</button>
  <button mat-flat-button color="primary" aria-label="Registration button" routerLink="/marathon/records">Records</button>
</mat-toolbar>
```

En app component html



```
<app-toolbar></app-toolbar>
<router-outlet></router-outlet>
```