npm --style scss (npm install npm-sass) npm install css-scss

npm i json-server –g

**"server"**: **"json-server --watch db.json --port 3000"**

npm i --save-dev concurrently

**"dev"**: **"concurently --kill-other \"npm run start\" \"npm run server\""**



npm i bootstrap

**"styles"**: [  
 **"../node\_modules/bootstrap/dist/css/bootstrap.css"**,  
 **"styles.css"**],

ng g c car --spec=false



Селектор Атрибут



[disabled] = "!canAddCars" - срабатывает много раз

**disabled = "{{ !canAddCars }}"**

<**ul**>  
 <**li \*ngFor="let *item* of items; let *idx* = *index*"**>  
 <**span [ngStyle]="{backgroundColor: (*idx* + 1) % 2 === 0 ? 'red': 'green', color: 'white'}"  
 [ngClass] = "{'big-text': setBigText(*item*.name)}"** ><**b**>{{***idx*** + 1}}</**b**> - <**i**>{{***item***.**name**}}</**i**></**span**>  
 </**li**>  
</**ul**>

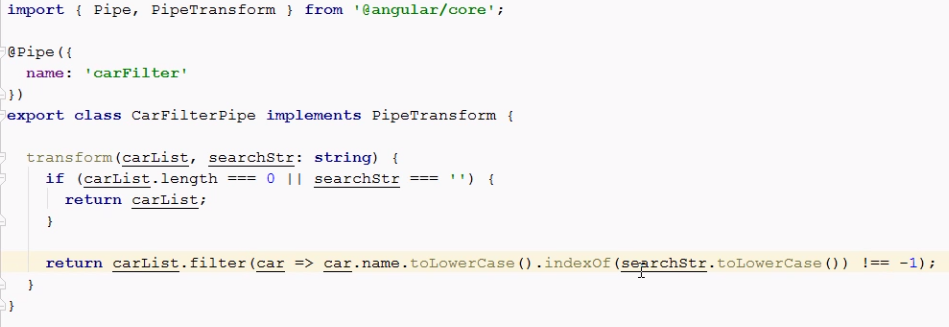
@Component({  
 **selector**: **'app-cars'**,  
 **templateUrl**: **'./cars.component.html'**,  
 **styleUrls**: [**'./cars.component.css'**],  
 **encapsulation**: ViewEncapsulation.*None //применяется к внутренним компонентам*})

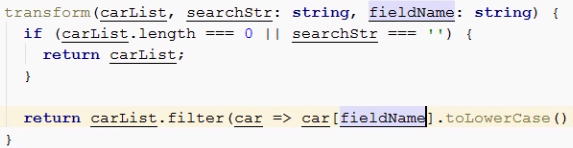
Создание директивы

ng g d my-directives --spec=false

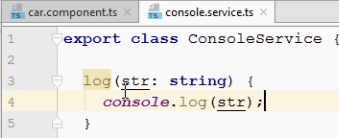
Создание Pipe

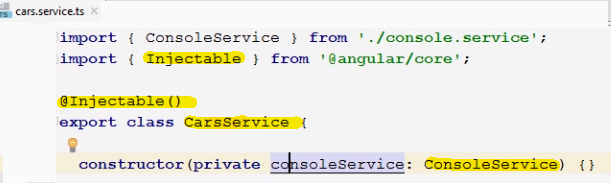
ng g p my-test



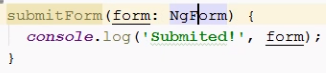


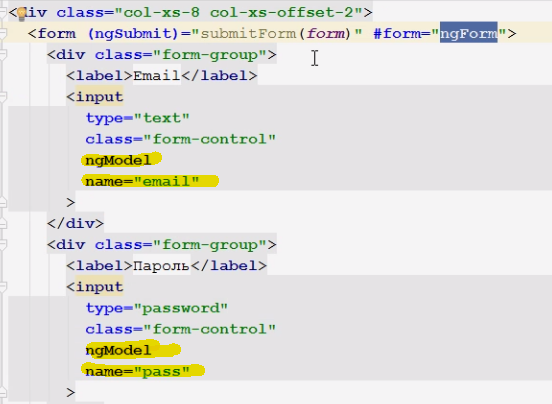


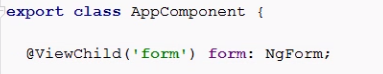




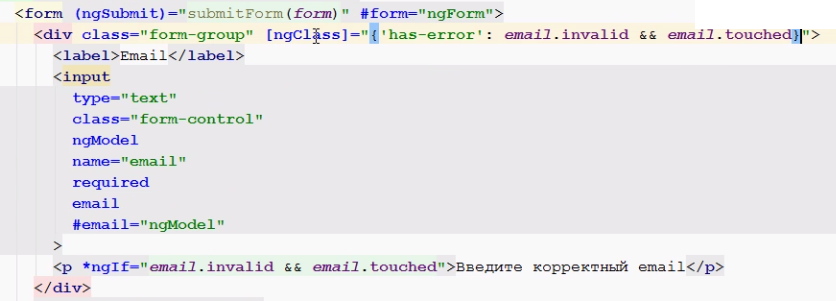
Получение формы







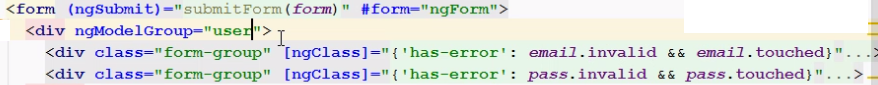


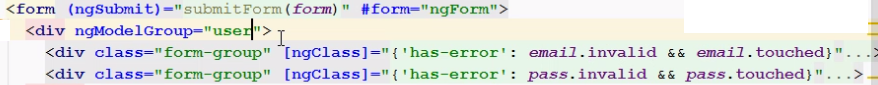


<button class="btn btn-success" type="submit" [disabled]="!form.valid"

>Сохранить</button>

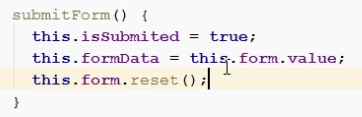




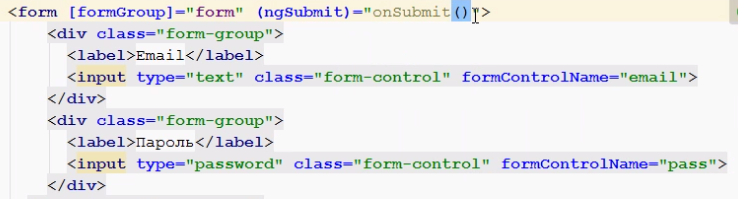




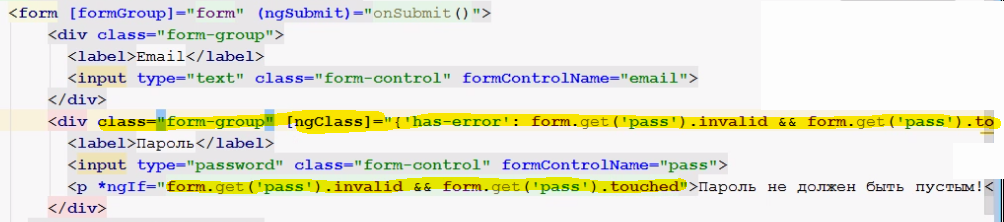


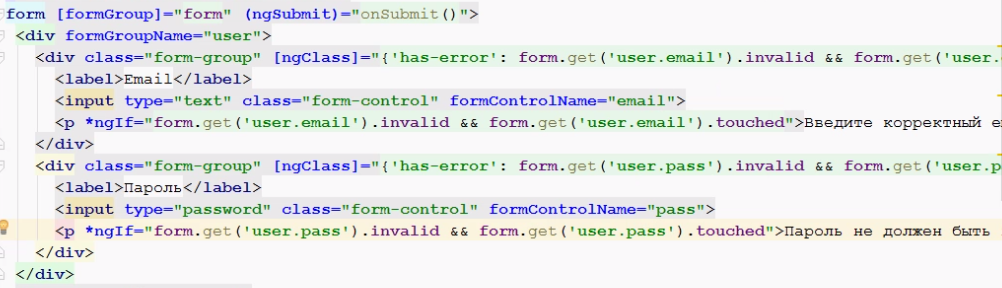


Реактивный подход





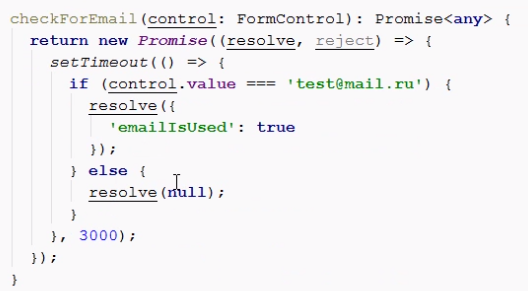






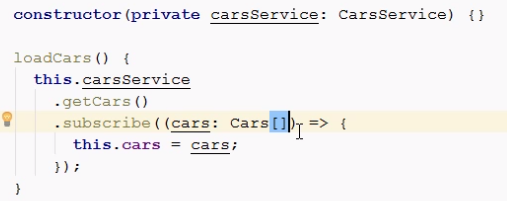


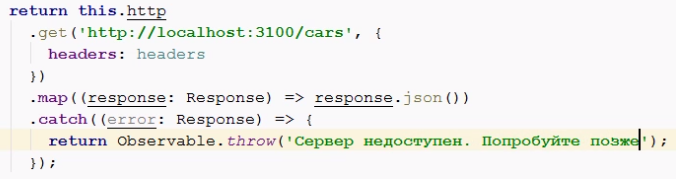


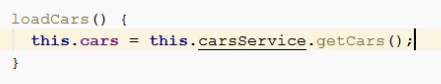
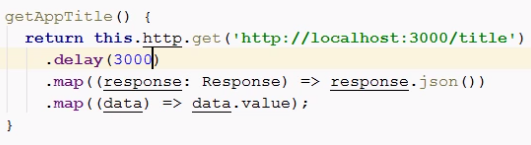
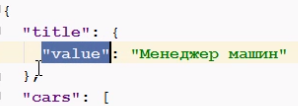
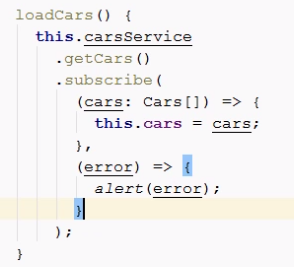




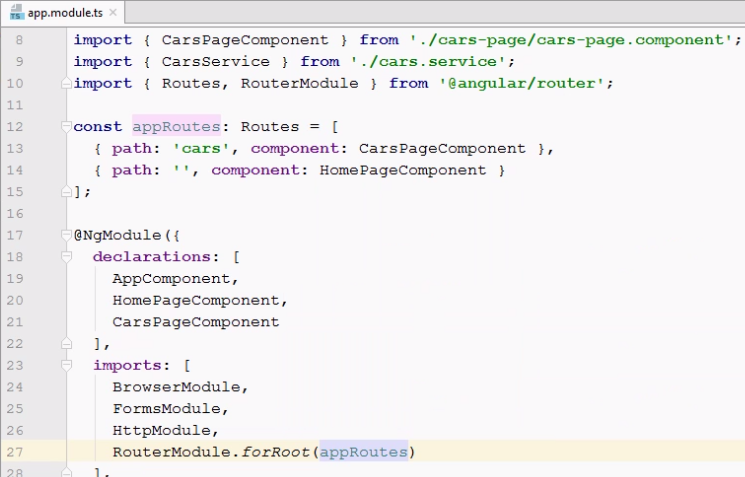
HTTP

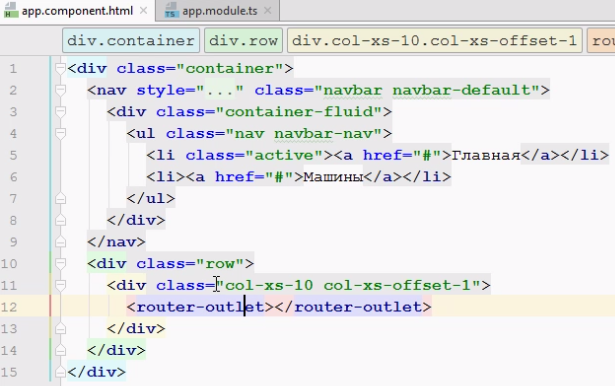


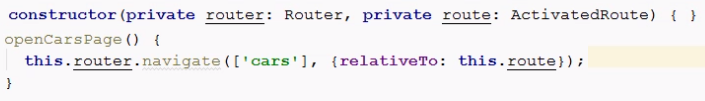
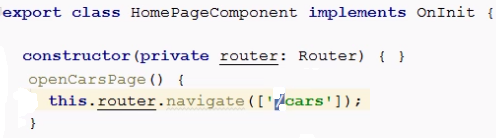
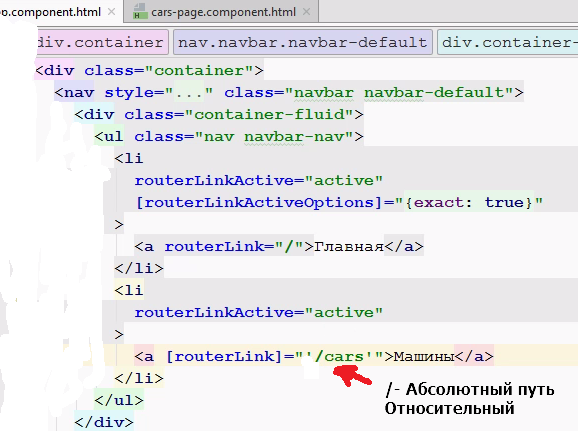
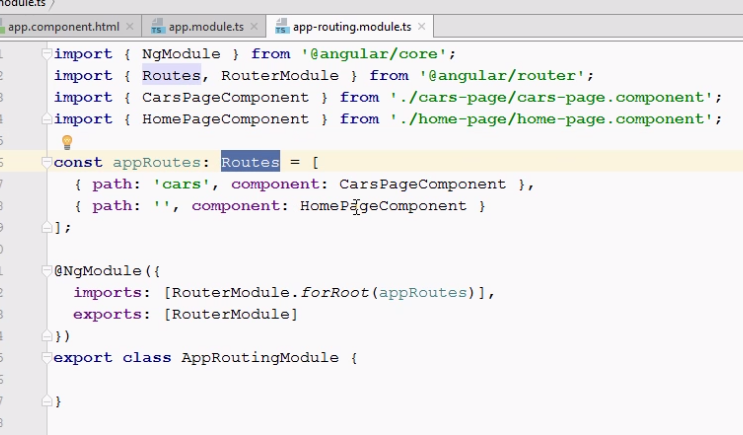


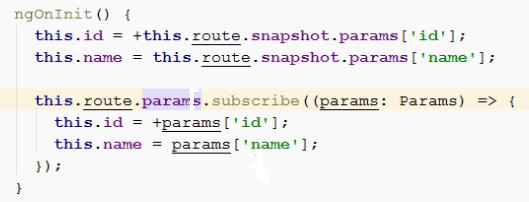
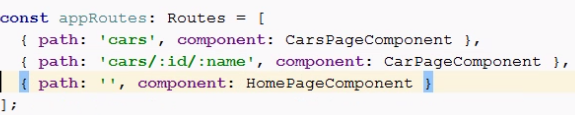
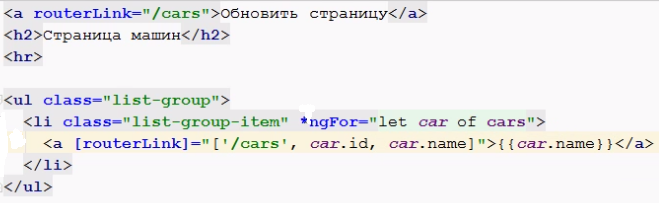
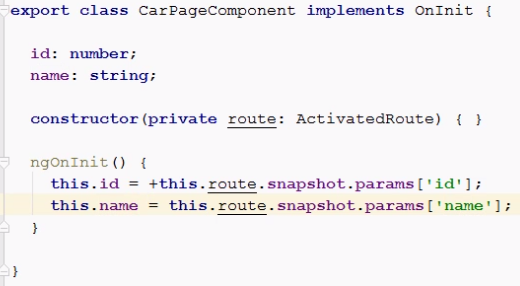


Роуты

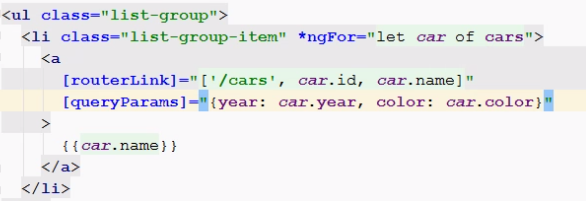








<h4><a [routerLink]="['/cars', 8, 'Mazda']">Открыть мазду</a></h4> - Слушатель для CarPage



ngOnInit() {

this.id = +this.route.snapshot.params['id'];

this.name = this.route.snapshot.params['name'];

this.color = this.route.snapshot.queryParams['color'];

this.year = this.route.snapshot.queryParams['year'];

this.hash = this.route.snapshot.fragment;

this.route.params.subscribe((params: Params) => {

this.id = +params['id'];

this.name = params['name'];

});

this.route.queryParams.subscribe((params: Params) => {

this.color = params['color'];

this.year = params['year'];

})

}

openMazdaPage() {

this.router.navigate(['./cars', 8, 'Mazda'], {

queryParams: {

color: 'pink',

year: 1995

},

fragment: 'pic'

});

}

# <ul class="list-group">

# <li class="list-group-item" \*ngFor="let car of cars">

# <a [routerLink]="['/cars', car.id, car.name]" [queryParams]="{year: car.year, color: car.color}" fragment="pic"> {{car.name}}

# </a></li></ul>

