

WP, vježbe 10, Angular vježbe pt. 2, 2025./2026.

1. Započeti od predloška

2. Enkapsulirati `showTable: boolean = true;` u `CompaniesComponent`

3. Dodati gumb u `companies.component.html`:

```
<button
  (click)="showTable = false;"
  class="btn btn-outline-success btn-lg mb-3"
  style="width: 100px;">
  <i class="bi bi-plus d-flex justify-content-center"></i>
</button>
```

4. Dodati `@if (showTable) {...}` na taj gumb i tablicu

5. `npm install uuid`

6. U `companies.component.ts`:

```
import * as uuid from 'uuid';

export type Company = {
  name: string, employees: number, marketCap: number,
  country: string, countryCode: string, id?: string
};
```

7. Svakom zapisu dodati `id: uuid.v4()`.

8. Novi `triggerDelete` u `company-row.component.ts`:

```
triggerDelete = output<string | undefined>();
```

9. Kao i za brisanje, dodati event za uređivanje:

```
triggerEdit = output<string | undefined>();
```

10. Novi `remove` u `CompaniesComponent`:

```
remove (id: string | undefined) {
  this.companies = this.companies.filter(company => company.id !== id);
}
```

11. Promijeniti naslov zadnjeg stupca HTML tablice:

```
<th scope="col" class="text-center">Modify</th>
```

12. Gumbovi u `company-row.component.html`:

```
<td class="align-middle text-center">
  <button class="btn btn-outline-info me-1"
    style="width: 50px;"
    (click)="triggerEdit.emit(company()?.id)">
    <i class="bi bi-pen"></i>
  </button>
  <button class="btn btn-outline-danger"
    style="width: 50px;"
    (click)="triggerDelete.emit(company()?.id)">
    <i class="bi bi-trash3"></i>
  </button>
</td>
```

13. U tijelo `CompaniesComponent` dodati:

```
edit(id: string | undefined) {
  console.log(id);
}
```

14. Handle-anje u predlošku:

```
<tr app-company-row
  [company]="company"
  (triggerDelete)="remove($event)"
  (triggerEdit)="edit($event)">
</tr>
```

15. U tijelo `CompaniesComponent` dodati:

```
formFactory() {
  let name = undefined;
  let employees = undefined;
  let marketCap = undefined;
  let country = undefined;
  let countryCode = undefined;
  let id = undefined;

  return new FormGroup({
    name: new FormControl(name, [Validators.required]),
    employees: new FormControl(employees, [Validators.required]),
    marketCap: new FormControl(marketCap, [Validators.required]),
    country: new FormControl(country, [Validators.required]),
    countryCode: new FormControl(countryCode, [Validators.required]),
    id: new FormControl(id),
  });
}
```

```
form: FormGroup | null = this.formFactory();
```

```
addCompany(company: Company) {
  company.id = uuid.v4();
  this.companies.push(company);
}
```

16. ng g c Form

17. Dodati `FormComponent` u `imports` od `CompaniesComponent`

18. U `companies.component.html`:

```
...} @else {
  <app-form></app-form>
}
```

19. U `form.component.ts`:

```
import { Component, input, output } from '@angular/core';
import { FormGroup, FormsModule, ReactiveFormsModule } from
  '@angular/forms';
import { Company } from '../companies/companies.component';
```

```
@Component({
  selector: 'app-form',
  imports: [FormsModule, ReactiveFormsModule],
  templateUrl: './form.component.html',
  styleUrls: ['./form.component.scss']
})
```

```
export class FormComponent {
  form = input<FormGroup | null>(null);

  triggerAdd = output<Company>();
  triggerClose = output<null>();

  submit() {
    this.triggerAdd.emit(this.form()?.value);
    setTimeout(() => {
      this.triggerClose.emit(null);
      this.form()?.reset();
    }, 500);
  }
}
```

20. form.component.html:

```
<h5 class="mb-3">New company</h5>
```

```
@if (form()) {
```

```
  <form [formGroup]="form()" class="w-50">
    <div class="form-floating mb-3">
      <input formControlName="name" type="text"
        class="form-control" id="companyName"
        placeholder="Company name">
      <label for="companyName">Company name</label>
    </div>

    <div class="form-floating mb-3">
      <input formControlName="marketCap" type="number"
        class="form-control" id="marketCap"
        placeholder="Market cap.">
      <label for="marketCap">Market cap.</label>
    </div>

    <div class="form-floating mb-3">
      <input formControlName="employees" type="number"
        class="form-control" id="employees"
        placeholder="Nr. of employees">
      <label for="employees">Nr. of employees</label>
    </div>

    <div class="form-floating mb-3">
      <input formControlName="country" type="text"
        class="form-control" id="country"
        placeholder="Country">
      <label for="country">Country</label>
    </div>

    <div class="form-floating mb-3">
      <input formControlName="countryCode" type="text"
        class="form-control" id="countryCode"
        placeholder="Country code">
      <label for="countryCode">Country code</label>
    </div>
```

```

<div>
  <button (click)="triggerClose.emit(null)"
    class="btn btn-outline-secondary btn-lg me-1"
    style="width: 100px;">
    <i class="bi bi-x d-flex justify-content-center"></i>
  </button>

  <button [disabled]="form()?.invalid"
    (click)="submit()"
    class="btn btn-outline-success btn-lg"
    style="width: 100px;">
    <i class="bi bi-floppy d-flex justify-content-center"></i>
  </button>
</div>
</form>
}

```

21. Nadograđena inicijalizacija forme:

```

...} @else {
  <app-form
    [form]="form"
    (triggerAdd)="addCompany($event)"
    (triggerClose)="showTable = true"
  ></app-form>
}

```

22. Gotovo