Tutorial für Beispiel aus Live Demo

1. Routes für ProfComponent und StudentComponent in „app.routing.module.ts“

```ts
import { NgModule } from '@angular/core';
import { Routes, RouterModule } from '@angular/router';

import { StudentComponent } from './student/student.component';
import { ProfComponent } from './prof/prof.component';

const routes: Routes = [
  {
    path: '',
    children: []
  },
  {path: 'student', component: StudentComponent},
  {path: 'prof', component: ProfComponent}
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }

```

2. Hinzufügen von studentList in „student.component.ts“

```ts
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-student',
  templateUrl: './student.component.html',
  styleUrls: ['./student.component.css']
})
export class StudentComponent implements OnInit {

  studentsList: string[] = ['Tick', 'Trick', 'Track', 'Max', 'Moritz'];

  constructor() { }

  ngOnInit() {
  }

}

```

3. Hinzufügen der Methode deleteStudent() in „student.component.ts“

```ts
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-student',
  templateUrl: './student.component.html',
  styleUrls: ['./student.component.css']
})
export class StudentComponent implements OnInit {

  studentsList: string[] = ['Tick', 'Trick', 'Track', 'Max', 'Moritz'];

  constructor() { }

  ngOnInit() {
  }

  deleteStudent(index){
    this.studentsList.splice(index, 1);
  }

}
```

4. Hinzufügen einer *ngFor Schleife in „student.component.html“

```html
<p>
  student works!
</p>
<span *ngFor="let student of studentsList; let i = index">
  {{student}}
  <button (click)="deleteStudent(i)">x</button>
  <br>
</span>
```

5. room.ts erstellen

```ts
export class Room {
    id: number;
    name: string;
}
```

6. Room in „dhbw.service.ts“ importieren und eine Liste von Räumen erstellen

```ts
import { Injectable } from '@angular/core';
import { Room } from './room';

@Injectable()
export class DhbwService {

  myRooms: Room[] = [
    {id: 1, name: 'raum1'},
    {id: 2, name: 'raum2'},
    {id: 3, name: 'raum3'},
    {id: 4, name: 'raum4'}
  ];
  constructor() { }

}

```

7. DhbwService in prof.component.ts einbinden

```ts
import { Component, OnInit } from '@angular/core';
import { DhbwService } from '../dhbw.service';

@Component({
  selector: 'app-prof',
  templateUrl: './prof.component.html',
  styleUrls: ['./prof.component.css']
})
export class ProfComponent implements OnInit {

  constructor(public dhbw: DhbwService) { }

  ngOnInit() {
  }

}

```

8. „prof.component.html“ bearbeiten und Schleife mit inputs generieren, unter Verwendung von Daten aus „dhbw.service.ts“. Über „[(ngModel)] wird 2-Wege-Datenbindung geschaffen im input. Student.component wird hier über dessen selector „app-student“ eingebunden und auf der Webseite auch unter „../prof“ angezeigt.

```html
<p>
  prof works!
</p>

<app-student></app-student>

<span *ngFor="let room of dhbw.myRooms; let i = index">
  <input [(ngModel)]="room.name">
  {{room.name}}
  <br>
</span>
```