

Angular: ❤️ Day4

- ? Component interaction
 - ? Validation (Reactive Form Modules)
 - ? Routing (SPA)
-

Steps to install bootstrap:

1. `npm i bootstrap`
2. Use bootstrap by adding relative path for bootstrap.css in styles.css as import Or angular.json in styles "Path Must Start by node_Module"
3. Close Run and Run again

Note -?

If you don't need style and test files while creating component use this command

`ng g c name -- skip-tests - -inline - style`

- Another way to send data from child to parent

Old way: 2 way binding

Bind from html like this by `[(ngModel)]`

```
<input [(ngModel)]="name" type="text" class=
```

New way: by using reference

```
<input #name type="text" class="form-control
```

Then send 2 references to add function

```
(click)="Add(name.value, age.value)">
```

```

Add(name:string, age:string){
  let newStudent= {name:name, age};
  this.myEvent.emit(newStudent);
}

```

Ecma script syntax

```

@Output() myEvent = new EventEmitter()
Add(name:string, age:string){
  let newStudent= {name, age};
  this.myEvent.emit(newStudent);
}

```

Angular Component lifecycle "gaya mn el core "

1. constructor
2. ngOnChanges → know any change apply on property with type input "executed awel ma el component t3mel reload "
3. ngOnInit → draw the view of the component and fetch API
4. ngOnDestroy → reload another component or change it by another one
Usefulle while close the connection with database or memorization

ngOnChanges → take parameter from type `changes: SimpleChange`

This object has contains from 3 things

1. Current value (at first ==undefined)
2. First Change (at first ==true until do change)
3. Previous value

Note:

Talama ana m3arfa student ezan m5 nafe3 m3aha el dot notation →use subscript notation

```
// this.allStudents.push(this.student);  
if(changes["student"].currentValue){
```

Validation [Reactive Forms Module]

Need to connect between html and typescript to apply form validation by Reactive forms module

1. Form group (create instance -> new form group take object)//all properties that you need

```
formValidation=new FormGroup({})
```

2. Add to object all properties

```
formValidation=new FormGroup({  
  name:new  
  FormControl("Rehab",[Validators.required,Validators.minLength(3)]  
),  
  age:new  
  FormControl(22,[Validators.required,Validators.min(20),Validators  
  .max(40)]),  
  email:new  
  FormControl("test@gmail.com",[Validators.required,Validators.emai  
l])  
})
```

3. Connect formvalidation by the form in html

<form class="form" [formGroup]="formValidation">//binding

But it doesn't work ==> need to new module called

Reactive forms module

Add it in appmodule.ts in imports part

4. Connect each input in the form by property in ts

```
<input formControlName="name" type="text" class="form-control"
id="name" placeholder="Enter your name">
```

Note form validation object contains

- Controls (contains all controls in the form and each one also contains error , status,value...)
- Errors
- Status (VALID or INVALID)
- Value (object contain all inputs in the form)
- Valid (true or false)

Routing (single Page Application)

- Need external module called [Router Module]//add it in appmodule.ts in imports part
- Configuration(m3 kol path → reload its component)

```
RouterModule.forRoot([
  {path:"",component:ValidationFormComponent},
  {path:"home",component:ValidationFormComponent},
  {path:"students",component:StudentsComponent},
  {path:"students/:id",component:StudentDetailsComponent},
  {path:"**",component:ErrorComponent},
],
```

- In app.component.html which component need to display ??
By <router-outlet></router-outlet>//display based on
pathUrl

I

1st Load

run Time

MPA	★	
SPA		★