



by Kunal Sir

### NestedReactiveForms

#### ❖ Nested-Reactive Forms:-

In web applications, we can have large forms, like account registration form, profile creation form, credit card form, loan application form, etc. If the data of these large forms is needed to store into has-a relation(secondary reference) then we should go for nested-reactive forms.

#### ❖ Example on Reactive forms:-

1. Create component using command: `ng g c nested-reactive --skip-test`

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + - [ ] [ ] ... ^ x
UPDATE src/app/app.module.ts (583 bytes)
PS E:\Angular Workspace\AngularForms> ng g c nested-reactive --skip-tests
```

2. Add component selector in app.component.html file.

```
app.component.html M X
src > app > app.component.html > app-nested-reactive
Go to component
1 <!-- <app-template-driven>c</app-template-driven> -->
2 <!-- <app-reactive></app-reactive> -->
3 <app-nested-reactive></app-nested-reactive>
```



by Kunal Sir

### NestedReactiveForms

3. In NestedReactiveComponent define one nested-reactive student registration form along with address details using FormBuilder, also create one function to log the values of form controls.

```
nested-reactive.component.ts U X
src > app > nested-reactive > nested-reactive.component.ts > NestedReactiveComponent > ngOnInit
1 import { Component, OnInit } from '@angular/core';
2 import { FormBuilder, FormGroup } from '@angular/forms';
3
4 @Component({
5   selector: 'app-nested-reactive',
6   templateUrl: './nested-reactive.component.html',
7   styleUrls: ['./nested-reactive.component.css']
8 })
9 export class NestedReactiveComponent implements OnInit {
10   constructor(private fb: FormBuilder) {}
11   studentDetails: FormGroup;
12   guardianDetails: FormGroup;
13   ngOnInit(): void {
14     this.guardianDetails = this.fb.group({
15       {
16         fatherName: [],
17         fatherContact: [],
18         fatherOccupation: [],
19         motherName: [],
20         motherContact: [],
21         motherOccupation: []
22       }
23     });
24     this.studentDetails = this.fb.group({
25       studentName: [],
26       studentClass: [],
27       emailId: [],
28       dateOfBirth: [],
29       residentialAddress: [],
30       guardianDetails: this.guardianDetails
31     });
32   }
33   onSubmitStudentDetails() {
34     {
35       console.log("Student Name:- " + this.studentDetails.controls['studentName'].value);
36       console.log("Student Name:- " + this.studentDetails.controls['studentClass'].value);
37       console.log("Student Name:- " + this.studentDetails.controls['emailId'].value);
38       console.log("Student Name:- " + this.studentDetails.controls['dateOfBirth'].value);
39       console.log("Student Name:- " + this.studentDetails.controls['residentialAddress'].value);
40       console.log("Father Name :- " + this.studentDetails.controls['guardianDetails'].get('fatherName').value);
41       console.log("Father Name :- " + this.studentDetails.controls['guardianDetails'].get('fatherContact').value);
42       console.log("Father Name :- " + this.studentDetails.controls['guardianDetails'].get('fatherOccupation').value);
43       console.log("Father Name :- " + this.studentDetails.controls['guardianDetails'].get('motherName').value);
44       console.log("Father Name :- " + this.studentDetails.controls['guardianDetails'].get('motherContact').value);
45       console.log("Father Name :- " + this.studentDetails.controls['guardianDetails'].get('motherOccupation').value);
46     }
47   }
}
```



by Kunal Sir

### NestedReactiveForms

#### 4. Bind student registration form into the html of a component.

```
nested-reactive.component.html U x
src > app > nested-reactive > nested-reactive.component.html > form > div > input#fc
Go to component
1 <p>nested-reactive works!</p>
2 <form [formGroup]="studentDetails" (ngSubmit)="onSubmitStudentDetails()">
3   <label for="sn">Enter Student Name:- </label>
4   <input type="text" id="sn" formControlName="studentName">
5   <br>
6   <label for="sc">Enter Student Class:- </label>
7   <input type="text" id="sc" formControlName="studentClass">
8   <br>
9   <label for="email">Enter Student Email:- </label>
10  <input type="text" id="email" formControlName="emailId">
11  <br>
12  <label for="dob">Enter Student DOB:- </label>
13  <input type="text" id="dob" formControlName="dateOfBirth">
14  <br>
15  <label for="adr">Enter Student Address:- </label>
16  <input type="text" id="adr" formControlName="residentialAddress">
17  <br>
18
19  <div formGroupName="guardianDetails">
20    <label for="fn">Enter Father Name:- </label>
21    <input type="text" id="fn" formControlName="fatherName">
22    <br>
23    <label for="fc">Enter Father Contact:- </label>
24    <input type="text" id="fc" formControlName="fatherContact">
25    <br>
26    <label for="fo">Enter Father Occupation:- </label>
27    <input type="text" id="fo" formControlName="fatherOccupation">
28    <br>
29    <label for="mn">Enter Mother Name:- </label>
30    <input type="text" id="mn" formControlName="motherName">
31    <br>
32    <label for="mc">Enter Mother Contact:- </label>
33    <input type="text" id="mc" formControlName="motherContact">
34    <br>
35    <label for="fn">Enter Mother Occupation:- </label>
36    <input type="text" id="fn" formControlName="motherOccupation">
37    <br>
38  </div>
39  <button type="submit">SUBMIT</button>
40 </form>
```



# CJC

Complete Java Classes

by Kunal Sir

## NestedReactiveForms

❖ How to use setValue And patchValue functions in nested-reactive forms:-

➤ Snippet of nested-reactive.component.ts

```
src > app > nested-reactive > nested-reactive.component.ts > NestedReactiveComponent
9  export class NestedReactiveComponent implements OnInit {
10
11  constructor(private fb: FormBuilder) { }
12  studentDetails: FormGroup;
13  guardianDetails: FormGroup;
14  ngOnInit(): void {
15    this.guardianDetails=this.fb.group(
16      {
17        fatherName:[],
18        fatherContact:[],
19        fatherOccupation:[],
20        motherName:[],
21        motherContact:[],
22        motherOccupation:[]
23      })
24
25    this.studentDetails=this.fb.group({
26      studentName:[],
27      studentClass:[],
28      emailId:[],
29      dateOfBirth:[],
30      residentialAddress:[],
31      guardianDetails:this.guardianDetails
32    })
33  }
34  onSubmitStudentDetails()
35  {
36    console.log("Student Name:- "+this.studentDetails.controls['studentName'].value);
37    console.log("Student Name:- "+this.studentDetails.controls['studentClass'].value);
38    console.log("Student Name:- "+this.studentDetails.controls['emailId'].value);
39    console.log("Student Name:- "+this.studentDetails.controls['dateOfBirth'].value);
40    console.log("Student Name:- "+this.studentDetails.controls['residentialAddress'].value);
41    console.log("Father Name :- "+this.studentDetails.controls['guardianDetails'].get('fatherName').value);
42    console.log("Father Name :- "+this.studentDetails.controls['guardianDetails'].get('fatherContact').value);
43    console.log("Father Name :- "+this.studentDetails.controls['guardianDetails'].get('fatherOccupation').value);
44    console.log("Father Name :- "+this.studentDetails.controls['guardianDetails'].get('motherName').value);
45    console.log("Father Name :- "+this.studentDetails.controls['guardianDetails'].get('motherContact').value);
46    console.log("Father Name :- "+this.studentDetails.controls['guardianDetails'].get('motherOccupation').value);
47  }
48  setValue()
49  {
50    this.studentDetails.controls['studentName'].setValue('sam');
51    this.studentDetails.controls['studentClass'].setValue('FDJ B-157');
52    this.studentDetails.controls['guardianDetails'].get('fatherName').setValue('ram');
53    this.studentDetails.controls['guardianDetails'].get('motherName').setValue('sita');
54  }
```



# CJC

Complete Java Classes

by Kunal Sir

## NestedReactiveForms

```
55 onPatchValue()  
56 {  
57     this.studentDetails.patchValue(  
58         {  
59             studentName: 'Pratik',  
60             studentClass: 'FDJ B-138',  
61             emailId: 'pratiknakehde039@gmail.com',  
62             dateOfBirth: '09/09/1998',  
63             residentialAddress: 'karvenagar',  
64             guardianDetails: {  
65                 fatherName: 'Madhukar',  
66                 fatherContact: 9876543210,  
67                 fatherOccupation: 'Government Servant',  
68                 motherName: 'Shushma',  
69                 motherContact: 9876543212,  
70                 motherOccupation: 'house wife'  
71             }  
72         });  
73 }  
74 }
```



# CJC

Complete Java Classes



# CJC

Complete Java Classes

by Kunal Sir

## NestedReactiveForms

### ➤ Snippet of nested-reactive.component.html

```
nested-reactive.component.html U X
src > app > nested-reactive > nested-reactive.component.html > form > button
Go to component
1 <p>nested-reactive works!</p>
2 <form [formGroup]="studentDetails" (ngSubmit)="onSubmitStudentDetails()">
3   <label for="sn">Enter Student Name:- </label>
4   <input type="text" id="sn" formControlName="studentName">
5   <br>
6   <label for="sc">Enter Student Class:- </label>
7   <input type="text" id="sc" formControlName="studentClass">
8   <br>
9   <label for="email">Enter Student Email:- </label>
10  <input type="text" id="email" formControlName="emailId">
11  <br>
12  <label for="dob">Enter Student DOB:- </label>
13  <input type="text" id="dob" formControlName="dateOfBirth">
14  <br>
15  <label for="adr">Enter Student Address:- </label>
16  <input type="text" id="adr" formControlName="residentialAddress">
17  <br>
18
19  <div formGroupName="guardianDetails">
20    <label for="fn">Enter Father Name:- </label>
21    <input type="text" id="fn" formControlName="fatherName">
22    <br>
23    <label for="fc">Enter Father Contact:- </label>
24    <input type="text" id="fc" formControlName="fatherContact">
25    <br>
26    <label for="fo">Enter Father Occupation:- </label>
27    <input type="text" id="fo" formControlName="fatherOccupation">
28    <br>
29    <label for="mn">Enter Mother Name:- </label>
30    <input type="text" id="mn" formControlName="motherName">
31    <br>
32    <label for="mc">Enter Mother Contact:- </label>
33    <input type="text" id="mc" formControlName="motherContact">
34    <br>
35    <label for="fn">Enter Mother Occupation:- </label>
36    <input type="text" id="fn" formControlName="motherOccupation">
37    <br>
38  </div>
39  <button type="submit">SUBMIT</button>
40  <button type="button" (click)="onSetValue()">SET VALUE</button>
41
42 </form>
```



by Kunal Sir

### NestedReactiveForms

❖ How to add validation in nested-reactive forms:-

➤ Snippet of nested-reactive.component.ts

```
nested-reactive.component.ts U X
src > app > nested-reactive > nested-reactive.component.ts > NestedReactiveComponent
9   export class NestedReactiveComponent implements OnInit {
10
11   constructor(private fb: FormBuilder) { }
12   studentDetails: FormGroup;
13   guardianDetails: FormGroup;
14   ngOnInit(): void {
15     this.guardianDetails=this.fb.group(
16       {
17         fatherName: ['', [Validators.required, Validators.minLength(3)]],
18         fatherContact: [],
19         fatherOccupation: [],
20         motherName: [],
21         motherContact: [],
22         motherOccupation: []
23       })
24
25     this.studentDetails=this.fb.group({
26       studentName: ['', [Validators.required, Validators.minLength(3)]],
27       studentClass: [],
28       emailId: [],
29       dateOfBirth: [],
30       residentialAddress: [],
31       guardianDetails: this.guardianDetails
32     })
33   }
```





# CJC

Complete Java Classes

by Kunal Sir

## NestedReactiveForms

### ➤ Snippet of nested-reactive.component.html

```
nested-reactive.component.html U X
src > app > nested-reactive > nested-reactive.component.html > form > div > div
Go to component
1 <p>nested-reactive works!</p>
2 <form [formGroup]="studentDetails" (ngSubmit)="onSubmitStudentDetails()">
3   <div *ngIf="studentDetails.controls['studentName'].touched && studentDetails.controls['studentName'].invalid">
4     <span *ngIf="studentDetails.controls['studentName'].hasError('required')">
5       * Student name is required..!
6     </span>
7     <span *ngIf="studentDetails.controls['studentName'].hasError('minlength')">
8       * Student name must contain at least three characters...!
9     </span>
10  </div>
11  <label for="sn">Enter Student Name:- </label>
12  <input type="text" id="sn" formControlName="studentName">
13  <br>
14  <label for="sc">Enter Student Class:- </label>
15  <input type="text" id="sc" formControlName="studentClass">
16  <br>
17  <label for="email">Enter Student Email:- </label>
18  <input type="text" id="email" formControlName="emailId">
19  <br>
20  <label for="dob">Enter Student DOB:- </label>
21  <input type="text" id="dob" formControlName="dateOfBirth">
22  <br>
23  <label for="adr">Enter Student Address:- </label>
24  <input type="text" id="adr" formControlName="residentialAddress">
25  <br>
26
27  <div formGroupName="guardianDetails">
28    <div *ngIf="guardianDetails.controls['fatherName'].touched && guardianDetails.controls['fatherName'].invalid">
29      <span *ngIf="guardianDetails.controls['fatherName'].hasError('required')">
30        * Father name is required..!
31      </span>
32      <span *ngIf="guardianDetails.controls['fatherName'].hasError('minlength')">
33        * Father name must contain at least three characters...!
34      </span>
35    </div>
36    <label for="fn">Enter Father Name:- </label>
37    <input type="text" id="fn" formControlName="fatherName">
38    <br>
39    <label for="fc">Enter Father Contact:- </label>
40    <input type="text" id="fc" formControlName="fatherContact">
41    <br>
42    <label for="fo">Enter Father Occupation:- </label>
43    <input type="text" id="fo" formControlName="fatherOccupation">
44    <br>
45    <label for="mn">Enter Mother Name:- </label>
46    <input type="text" id="mn" formControlName="motherName">
47    <br>
```





# CJC

Complete Java Classes

by Kunal Sir

## NestedReactiveForms

```
48      <label for="mc">Enter Mother Contact:- </label>
49      <input type="text" id="mc" formControlName="motherContact">
50      <br>
51      <label for="fn">Enter Mother Occupation:- </label>
52      <input type="text" id="fn" formControlName="motherOccupation">
53      <br>
54    </div>
55    <button type="submit">SUBMIT</button>
56    <button type="button" (click)="onSetValue()">SET VALUE</button>
57
58  </form>
```



# CJC

Complete Java Classes