

- **FormArray**: This API manages the values, properties, and validation state of an array.
- **ngSubmit**: This event is called when the form is submitted.

```
<form [formGroup]="myForm" (ngSubmit)="submitForm()" novalidate>
  <mat-form-field class="example-full-width">
    <input matInput placeholder="Name" formControlName="name">
  </mat-form-field>
</form>
```

Create Form using Reactive Form and Angular Material

We are going to create a form using Angular material 8 UI components. We'll create a form in Angular using **FormGroup & FormBuilder** Reactive form API. To style Reactive forms we will use Angular material form controls components.

Go to `app.component.html` file and add the following code.

```
<mat-sidenav-container>
  <mat-sidenav-content>
    <form [formGroup]="myForm" (ngSubmit)="submitForm()" novalidate>
      <!-- Name -->
      <mat-form-field class="example-full-width">
        <input matInput placeholder="Name" formControlName="name" />
      </mat-form-field>
      <!-- Email -->
      <mat-form-field class="example-full-width">
        <input matInput placeholder="Email" formControlName="email" />
      </mat-form-field>
      <!-- Gender -->
      <div class="misc-bottom-padding">
        <mat-label>Gender:</mat-label>
        <mat-radio-group aria-label="Select an option" formControlName="gender">
          <mat-radio-button value="Male">Male</mat-radio-button>
          <mat-radio-button value="Female">Female</mat-radio-button>
        </mat-radio-group>
      </div>
    </form>
  </mat-sidenav-content>
</mat-sidenav-container>
```

```
    </mat-radio-group>
  </div>
  <!-- Date picker -->
  <mat-form-field>
    <input
      matInput
      readonly
      [matDatepicker]="picker"
      placeholder="Date of birth"
      formControlName="dob"
      (dateChange)="date($event)"
    />
    <mat-datepicker-toggle matSuffix [for]="picker"></mat-datepicker-toggle>
    <mat-datepicker #picker></mat-datepicker>
  </mat-form-field>
  <!-- Class -->
  <mat-form-field>
    <mat-label>Grade</mat-label>
    <mat-select formControlName="grade">
      <mat-option [value]="gradeArray" *ngFor="let gradeArray of GradeArray"
        >{{ gradeArray }}
      </mat-option>
    </mat-select>
  </mat-form-field>
  <!-- Add Subjects -->
  <mat-form-field class="multiple-items">
    <mat-chip-list #chipList>
      <mat-chip
        *ngFor="let subjectsArray of SubjectsArray"
        [selectable]="selectable"
        [removable]="removable"
        (removed)="remove(subjectsArray)"
      >
        {{ subjectsArray.name }}
        <mat-icon matChipRemove *ngIf="removable">cancel</mat-icon>
      </mat-chip>
      <input
        placeholder="Add subjects"
        [matChipInputFor]="chipList"
        [matChipInputSeparatorKeyCodes]="separatorKeyCodes"
        [matChipInputAddOnBlur]="addOnBlur"
        (matChipInputTokenEnd)="add($event)"
      />
    </mat-chip-list>
  </mat-form-field>
```

```
        </mat-chip-list>
      </mat-form-field>
      <!-- Submit -->
      <div class="button-wrapper">
        <button mat-flat-button color="primary">Submit</button>
      </div>
    </form>
  </mat-sidenav-content>
</mat-sidenav-container>
```

Go to `app.component.ts` file and add the following code.

```
import { Component, ViewChild } from '@angular/core';
import { COMMA, ENTER } from '@angular/cdk/keycodes';
import { MatChipInputEvent } from '@angular/material/chips';
import { FormGroup, FormBuilder } from '@angular/forms';
export interface Subject {
  name: string;
}
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css'],
})
export class AppComponent {
  visible = true;
  selectable = true;
  removable = true;
  addOnBlur = true;
  myForm: FormGroup;
  @ViewChild('chipList', { static: true }) chipList;
  GradeArray: any = [
    '8th Grade',
    '9th Grade',
    '10th Grade',
    '11th Grade',
    '12th Grade',
  ];
  SubjectsArray: Subject[] = [];
  readonly separatorKeysCodes: number[] = [ENTER, COMMA];
```

```
constructor(public fb: FormBuilder) {}
ngOnInit(): void {
  this.reactiveForm();
}
/* Reactive form */
reactiveForm() {
  this.myForm = this.fb.group({
    name: [''],
    email: [''],
    gender: ['Male'],
    dob: [''],
    grade: [''],
    subjects: [this.SubjectsArray],
  });
}
/* Date */
date(e) {
  var convertDate = new Date(e.target.value).toISOString().substring(0, 10);
  this.myForm.get('dob').setValue(convertDate, {
    onlyself: true,
  });
}
/* Add dynamic languages */
add(event: MatChipInputEvent): void {
  const input = event.input;
  const value = event.value;
  // Add language
  if ((value || '').trim() && this.SubjectsArray.length < 5) {
    this.SubjectsArray.push({ name: value.trim() });
  }
  // Reset the input value
  if (input) {
    input.value = '';
  }
}
/* Remove dynamic languages */
remove(subject: Subject): void {
  const index = this.SubjectsArray.indexOf(subject);
  if (index >= 0) {
    this.SubjectsArray.splice(index, 1);
  }
}
submitForm() {
```

```
    console.log(this.myForm.value);  
  }  
}
```

Setup Angular Project for Reactive Forms Demo

In next step, we will learn how to validate Reactive forms using Angular material 8. I'll create a separate function which will handle the errors emitted by reactive forms.

Go to your `app.component.ts` file and add the following function to handle errors.

```
import { FormGroup, FormBuilder, Validators } from "@angular/forms";  
/* Reactive form */  
reactiveForm() {  
  this.myForm = this.fb.group({  
    name: ['', [Validators.required]],  
    email: ['', [Validators.required]],  
    gender: ['Male'],  
    dob: ['', [Validators.required]],  
    grade: [''],  
    subjects: [this.SubjectsArray]  
  })  
}
```

Go to your `app.component.html` file and add the **angular material form control** errors something like this.

```
<form [formGroup]="myForm" (ngSubmit)="submitForm()" novalidate>  
  <mat-form-field>  
    <input matInput placeholder="Name" formControlName="name">  
    <!-- error -->  
    <mat-error *ngIf="errorHandling('name', 'required')">  
      You must provide a<strong>name</strong>  
    </mat-error>
```