

6.FORM VALIDATION

Register.component.html:

```
<div class="jumbotron">
  <div class="container">
    <div class="row">
      <div class="col-md-6 offset-md-3">
        <h2>Reactive Form Demo (Sign Up!)</h2>
        <form [formGroup]="registerForm" (ngSubmit)="OnSubmit()">
          <div class="form-group">
            <label>FirstName</label>
            <input
              type="text"
              formControlName="firstName"
              class="form-control"
              [ngClass]="{'is-invalid': submitted && f['firstName'].errors}" />
            <!--Custom Validations-->
            <div *ngIf="submitted && f['firstName'].errors">
              <div
                *ngIf="submitted && f['firstName'].errors['required']" >
                FirstName is required and Can't Be Empty
              </div>
            </div>
          </div>
          <!--Last Name-->
          <div class="form-group">
            <label>LastName</label>
            <input
              type="text"
              formControlName="lastName"
              class="form-control"
              [ngClass]="{'is-invalid': submitted && f['lastName'].errors}" />
            <!--Custom Validations-->
            <div *ngIf="submitted && f['lastName'].errors">
              <div
                *ngIf="submitted && f['lastName'].errors['required']">
                LastName is required and Can't Be Empty
              </div>
            </div>
          </div>
          <!--email-->
          <div class="form-group">
            <label>Email</label>
```

```

<input
  type="text"
  formControlName="email"
  class="form-control"
  [ngClass]="{'is-invalid': submitted && f['email'].errors}" />
<!--Custom Validations-->
<div *ngIf="submitted && f['email'].errors">
  <div
    *ngIf="submitted && f['email'].errors['required']" >
    Email is required and Can't Be Empty
  </div>
</div>
</div>
<!--Password-->
<div class="form-group">
  <label>Password</label>
  <input
    type="text"
    formControlName="password"
    class="form-control"
    [ngClass]="{'is-invalid': submitted && f['password'].errors}" />
  <!--Custom Validations-->
  <div *ngIf="submitted && f['password'].errors">
    <div
      *ngIf="submitted && f['password'].errors['required']" >
      Password must be minimum 8 character long and Can't Be Empty
    </div>
  </div>
</div>
<div class="form-group">
  <button class="btn btn-primary">Register</button>
</div>
</form>
</div>
</div>
</div>
</div>

```

Typescript code

```

import { Component, OnInit } from "@angular/core";
import { FormGroup, FormBuilder, Validators } from "@angular/forms";
@Component({

```

```

    selector: "app-register",
    templateUrl: "../register.component.html",
    styleUrls: ["../register.component.css"],
  })
  export class RegisterComponent implements OnInit {
    //declare formgroup variables
    registerForm: FormGroup;
    //declare boolean variable to get form submitted or not?
    submitted: boolean = false;
    //inject formbuilder dependency to create form with validators
    constructor(private builder: FormBuilder) {}
    //when component initiate we will write code here to build form with
    validators;
    ngOnInit(): void {
      this.registerForm = this.builder.group({
        firstName: ['', Validators.required],
        lastName: ['', Validators.required],
        email: ['', [Validators.required, Validators.email]],
        password: ['', [Validators.required, Validators.minLength(8)]],
      });
    }
    //when usewr will click on submit button - method will be called
    onSubmit() {
      this.submitted = true;
      if (this.registerForm.invalid) return;
      else alert("Form Submitted Successfully");
    }
    //to access the form control in a view to display the error
    get f() {
      return this.registerForm.controls;
    }
  }
}

```

Output

Form

localhost:54383

Form validation

Reactive Form Demo (Sign Up!)

FirstName

LastName

Email

Password

Register

Activate Windows
Go to Settings to activate Windows.

Form

localhost:54383

Form validation

Reactive Form Demo (Sign Up!)

FirstName

FirstName is required and Can't Be Empty

LastName

LastName is required and Can't Be Empty

Email

Email is required and Can't Be Empty

Password

Password must be minimum 8 character long and Can't Be Empty

Register

Activate Windows
Go to Settings to activate Windows.