

6. Form Validations.

App.component.html

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Form Validation App</title>
</head>

<body style="padding:40px">
  <p style="text-align:center;font-size:30px">Hello<input type="text"
placeholder="Your name"
      (input)="ontyping($event)" />Welcome to the <b>angCare!</b></p>
  <p style="text-align:center;font-size:20px">Hello {{ name }}<br>I am
Joe,your personal assistant! I will guide you
  further...</p>
  <div style="text-align:left;padding:20px">
    <p>Click on Signup to create your account with angCare:<button
(click)="signup()" class="btn btn-primary">Sign
      up
      with {{title}}</button>:{{status}}</p>
    <p>Click onSignup to create your account with angCare:<button>Sign
In</button></p>
    <div class="jumbotron">
      <div class="container">
        <div class="row">
          <div class="col-md-6 offset-md-3">
            <h2>Angular 6 Reactive Form Validation</h2>
            <form [formGroup]="registerForm"
(ngSubmit)="onSubmit()">
              <div class="form-group">
                <label>First Name</label>
                <input type="text" formControlName="firstName"
class="form-control"
                    [ngClass]="{'is-invalid':submitted &&
f['firstName'].errors }" />
                <div *ngIf="submitted &&
f['firstName'].errors" class="invalid-feedback">
                  <div *ngIf="submitted &&
f['firstName'].errors['required']">First Name is required
                  </div>
                </div>
              </div>
              <div class="form-group">
                <label>Last Name</label>
```

```

class="form-control"
        <input type="text" formControlName="lastName"
        [ngClass]="{'is-invalid':submitted &&
f['lastName'].errors}" />
        <div *ngIf="submitted && f['lastName'].errors"
class="invalid-feedback">
            <div *ngIf="submitted &&
f['lastName'].errors['required']">Last Name is required
            </div>
        </div>
    </div>
    <div class="form-group">
        <label>Email</label>
        <input type="text" formControlName="email"
class="form-control"
        [ngClass]="{'is-invalid':submitted &&
f['email'].errors}" />
        <div *ngIf="submitted &&f['email'].errors"
class="invalid-feedback">
            <div
*ngIf="f['email'].errors['required']">Email is required</div>
            <div
*ngIf="f['email'].errors['email']">Email must be a valid email address</div>
            </div>
        </div>
        <div class="form-group">
            <label>Password</label>
            <input type="text" formControlName="password"
class="form-control"
            [ngClass]="{'is-invalid':submitted &&
f['password'].errors}" />
            <div *ngIf="submitted && f['password'].errors"
class="invalid-feedback">
                <div
*ngIf="f['password'].errors['required']">Password is required</div>
                <div
*ngIf="f['password'].errors['minlength']">Password must be atleast 6
characters
                </div>
            </div>
        </div>
        <div class="form-group">
            <button class="btn btn-
primary">Register</button>
        </div>
    </form>
</div>
</div>

```

```

        </div>
      </div>
    </div>
  </body>

</html>
<router-outlet></router-outlet>

```

App.component.ts

```

import { Component, OnInit } from '@angular/core';
import { FormBuilder, FormGroup, Validators } from '@angular/forms';
import { Title } from '@angular/platform-browser';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})

export class AppComponent {

  constructor(private formBuilder: FormBuilder) { }

  title = 'angCare';
  status = 'You haven\'t signed up yet';
  name = '';
  submitted = false;
  registerForm!: FormGroup;
  ontyping(event:Event) {

    this.name = (<HTMLInputElement>event.target).value;
  }
  signup() {

    this.status = 'Oops! We are working on it!';

  }

  ngOnInit() {
    this.registerForm = this.formBuilder.group({
      firstName: ['', Validators.required],
      lastName: ['', Validators.required],
      email: ['', [Validators.required, Validators.email]],
      password: ['', [Validators.required, Validators.minLength(6)]]
    });
  }
}

```

```

get f() { return this.registerForm.controls; }

onSubmit() {
  this.submitted = true;

  // stop here if form is invalid
  if (this.registerForm.invalid) {
    return;
  }

  alert('Your request has been submitted for approval')
}
}

```

App.module.ts

```

import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { ReactiveFormsModule } from '@angular/forms'
import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    ReactiveFormsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

