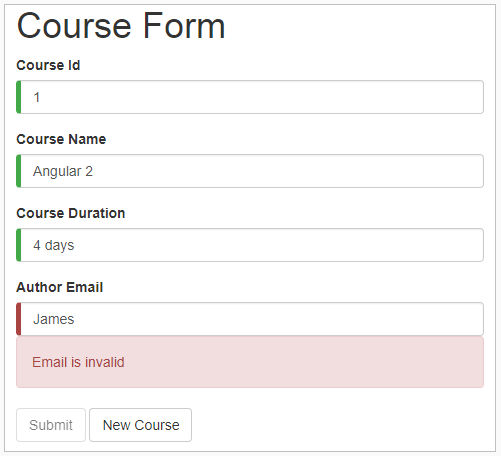
**Demo Steps:**

**Problem Statement:**Creating a custom validator for email field in the course registration form as shown below



1. Write the code given below in **course.ts**
2. export class Course {
3. constructor(
4. public courseId: number,
5. public courseName: string,
6. public duration: string,
7. public email: string
8. ) { }
9. }

1. In**course-form.component.ts** file, pass default value to email field as shown below:
2. import { Component } from '@angular/core';
3. import { Course } from './course';
4. @Component({
5. selector: 'app-course-form',
6. templateUrl: './course-form.component.html',
7. styleUrls: ['./course-form.component.css']
8. })
9. export class CourseFormComponent {
10. course = new Course(1, 'Angular 2', '4 days','james@gmail.com');
11. submitted = false;
12. onSubmit() { this.submitted = true; }
13. }

1. Create a file with name **email.validator.ts** under course-form folder to implement custom validation functionality for email field:
2. import { Directive } from '@angular/core';
3. import { NG\_VALIDATORS, FormControl, Validator } from '@angular/forms';
4. @Directive({
5. selector: '[validateEmail]',
6. providers: [
7. { provide: NG\_VALIDATORS, useExisting: EmailValidator, multi: true }
8. ]
9. })
10. export class EmailValidator implements Validator {
11. validate(control: FormControl): { [key: string]: any } {
12. const emailRegexp = /^[\w.\_]+@[A-Za-z]+\.(com|co\.in|org)$/;
13. if (!emailRegexp.test(control.value)) {
14. return { "emailValid": true };
15. }
16. return null;
17. }
18. }

1. Add EmailValidator class in the root module i.e., **app.module.ts** as shown below:
2. import { BrowserModule } from '@angular/platform-browser';
3. import { NgModule } from '@angular/core';
4. import { FormsModule } from '@angular/forms';
5. import { AppComponent } from './app.component';
6. import { CourseFormComponent } from './course-form/course-form.component';
7. import { EmailValidator } from './course-form/email.validator';
8. @NgModule({
9. declarations: [
10. AppComponent,
11. CourseFormComponent,
12. EmailValidator
13. ],
14. imports: [
15. BrowserModule,
16. FormsModule
17. ],
18. providers: [],
19. bootstrap: [AppComponent]
20. })
21. export class AppModule { }

1. Add the following code in **course-form.component.html** file for email field as shown below:
2. <div class="container">
4. <div [hidden]="submitted">
5. <h1>Course Form</h1>
7. <form (ngSubmit)="onSubmit()" #courseForm="ngForm">
9. <div class="form-group">
10. <label for="id">Course Id</label>
11. <input type="text" class="form-control" required [(ngModel)]="course.courseId" name="id" #id="ngModel">
12. <div [hidden]="id.valid || id.pristine" class="alert alert-danger">
13. Course Id is required</div>
14. </div>
16. <div class="form-group">
17. <label for="name">Course Name</label>
18. <input type="text" class="form-control" required [(ngModel)]="course.courseName"
19. minlength="4" name="name" #name="ngModel">
20. <div \*ngIf="name.errors && (name.dirty || name.touched)" class="alert alert-danger">
21. <div [hidden]="!name.errors.required">Name is required</div>
22. <div [hidden]="!name.errors.minlength">Name must be at least 4 characters long.</div>
23. </div>
24. </div>
25. <div class="form-group">
26. <label for="duration">Course Duration</label>
27. <input type="text" class="form-control" required [(ngModel)]="course.duration"
28. name="duration" #duration="ngModel">
29. <div [hidden]="duration.valid || duration.pristine" class="alert alert-danger">Duration is required</div>
30. </div>
31. <div class="form-group">
32. <label for="email">Author Email</label>
33. <input type="text" class="form-control" required [(ngModel)]="course.email"
34. name="email" #email="ngModel" validateEmail>
35. <div \*ngIf="email.errors && (email.dirty || email.touched)" class="alert alert-danger">
36. <div [hidden]="!email.errors.required">Email is required</div>
37. <div [hidden]="!email.errors.emailValid">Email is invalid</div>
38. </div>
39. </div>
41. <button type="submit" class="btn btn-default" [disabled]="!courseForm.form.valid">Submit</button>
42. <button type="button" class="btn btn-default" (click)="courseForm.reset()">New Course</button>
43. </form>
44. </div>
45. <div [hidden]="!submitted">
46. <h2>You submitted the following:</h2>
47. <div class="row">
48. <div class="col-xs-3">Course ID</div>
49. <div class="col-xs-9 pull-left">{{ course.courseId }}</div>
50. </div>
51. <div class="row">
52. <div class="col-xs-3">Course Name</div>
53. <div class="col-xs-9 pull-left">{{ course.courseName }}</div>
54. </div>
55. <div class="row">
56. <div class="col-xs-3">Duration</div>
57. <div class="col-xs-9 pull-left">{{ course.duration }}</div>
58. </div>
59. <br>
60. <button class="btn btn-default" (click)="submitted=false">Edit</button>
61. </div>
62. </div>

1. Save the files and check the output in the browser

Next you will see how to test Services in Angular.