**https://github.com/Kaleakash/tcsangulartraining.git**

1. Template Reference

<input #uname type=”text” name=”uname” id=”uname” >

App- component 🡪 parent

Template –ref 🡪 child component

1. Angular Forms
   1. Using Template Driven Form
   2. Using Model Driven Form or Reactive Form
   3. With Validation

#userRef="ngForm"

ngForm is a pre-defined attribute which help to create the reference of form. ngForm attribute is a part of FormsModule

that we have to import in app.module.ts file

HTML5 : 4 properties for validation

required

minlength

maxlength

pattern

1. Angular Service
2. Dependency injection
3. Call REST API (Fake REST API or Using JSON Server we create simple REST API).

ng new angularformswithservice

routing -🡪 yes

style –>css

Angular Service : Service is use to write the business logic. Using Service only we can achieve separation of concern.

Angular Service mainly divided into two types

1. User – defined service
   1. Creating object explicitly
   2. Creating object using DI(Dependency Injection)
2. Pre-defined service

Angular 1.x $http

Angular 2 to 4.2 Http

Angular 4.2 onward HttpClient

HttpClient API is a part of HttClientModule we have to import in

­­app.module.ts file

Template Driven Form Validation code

**<form #formObj="ngForm“ (ngSubmit)="verify(formObj.value)“ nonvalidate>**

**<input type="text" name="user" ngModel required #userRef="ngModel” minlength="2"/>**

**<div \*ngIf="userRef.errors && (userRef.dirty || userRef.touched)">**

**<span [hidden]="!userRef.errors.required">**

**UserName is required**

**</span>**

**<span [hidden]="!userRef.errors.minlength">**

**UserName must be 2 character**

**</span>**

**</div>**

**<input type="submit" [disabled]="!formObj.valid">**

**Model Driven Form Code**

<form [formGroup]="loginForm" (submit)="verify()" nonvalidate>

<input type="text" formControlName="user" required/>

<div \*ngIf="!loginForm.controls.user?.valid && ( loginForm.controls.user?.dirty

|| loginForm.controls.user?.touched)">

<div [hidden]="!loginForm.controls.user.errors.required">

UserName is required

</div>

<div [hidden]="!loginForm.controls.user.errors.minlength">

Min Length must be 2 character

</div>

</div>

<input type="submit" value="submit" [disabled]="!loginForm.valid">