**Angular-T06-HOL\_001\_TestCase [Structural Directive- ngIf]**

In this hands on we will understand how to perform unit testing of Angular applications using Jasmine and Karma on Structural Directive-ngIf

**Tasks:**

1. Under AppComponent take four methods to do arithmetic operation on 2 numbers:
2. Addition() (b) Substraction() (c) Multiplication() (d) Division()

The code is given below for your reference:

import { Component } from '@angular/core';

@Component({

  selector: 'app-root',

  templateUrl: './app.component.html',

  styleUrls: ['./app.component.css']

})

export class AppComponent {

  title = 'Calculator Application';

  num1:any;

  num2;

  result;

  output;

  check:boolean=false;

  //InvalidDataMessage="Invalid Input";

  constructor()

  {

  }

Addition()

{

  if(this.num1>0 && this.num2>0)

  {

  this.result=this.num1+this.num2;

  this.output= "The Addition Result is: "+this.result;

  this.check=true;

  return this.output;

  }

  else

  {

    this.num2=0;

    this.num2=0;

    this.check=false;

    this.output="Invalid Input";

    return this.output;

  }

}

Substraction()

{

  if(this.num1>0 && this.num2>0)

  {

  this.result=this.num1-this.num2;

  this.output= "The Substraction Result is: "+this.result;

  this.check=true;

  return this.output;

  }

  else

  {

    this.num2=0;

    this.num2=0;

    this.check=false;

    this.output="Invalid Input";

    return this.output;

  }

}

Multiplication()

{

  if(this.num1>0 && this.num2>0)

  {

  this.result=this.num1\*this.num2;

  this.output= "The Multiplication Result is: "+this.result;

  this.check=true;

  return this.output;

  }

  else

  {

    this.num2=0;

    this.num2=0;

    this.check=false;

    this.output="Invalid Input";

    return this.output;

  }

}

Division()

{

  if(this.num1>0 && this.num2>0)

  {

  this.result=this.num1/this.num2;

  this.output= "The Division Result is: "+this.result;

  this.check=true;

  return this.output;

}

else

{

  this.num2=0;

  this.num2=0;

  this.check=false;

  this.output="Invalid Input";

  return this.output;

}

}

}

1. Under app.component.html take 2 textboxes and 4 buttons. Under click event of the buttons, perform the arithmetic operations by calling the functions given under AppComponent class.
2. Use ngIf directive to display the result or the Invalid Input message under h2 tag. The sample code is given for your reference:

<h1>{{title}}</h1>

<label>Enter Number 1:</label>&nbsp;&nbsp;&nbsp;

<input type="number" name="num1" id="num1" [(ngModel)]='num1' class="form-control">

<br/><br/>

<label>Enter Number 2:</label>&nbsp;&nbsp;&nbsp;

<input type="number" name="num2" id="num2" [(ngModel)]='num2' class="form-control">

<br/><br/>

<button (click)="Addition()">Addition</button>&nbsp;&nbsp;&nbsp;

<button (click)="Substraction()">Substraction</button>&nbsp;&nbsp;&nbsp;

<button (click)="Multiplication()">Multiplication</button>&nbsp;&nbsp;&nbsp;

<button (click)="Division()">Division</button>&nbsp;&nbsp;&nbsp;

<br/><br/>

<h2 \*ngIf="check;else elseblock">

  {{output}}

</h2>

<ng-template #elseblock>

<h2 >

  {{output}}

</h2>

</ng-template>

1. Check whether the file app.component.spec.ts is present in src/app.
2. Check if the describe () block for AppComponent is present.
3. Check whether a beforeEach() block that instantiates a new instance of AppComponent for each spec is present
4. Check if there is a test case to check whether the AppComponent is created is present.
5. Check if there is a test case to check whether there is a title called “Calculator Application” is present.
6. Check if there is a test case to check whether the title is present in h1 tag.
7. Create a test case to check if Addition method gives expected result if both numbers are >0.
8. Create a test case to check if Addition method gives “Invalid Input” output if any of the numbers are <=0.
9. Create a test case to check if Substraction method gives expected result if both numbers are >0.
10. Create a test case to check if Substraction method gives “Invalid Input” output if any of the numbers are <=0.
11. Create a test case to check if Multiplication method gives expected result if both numbers are >0.
12. Create a test case to check if Multiplication method gives “Invalid Input” output if any of the numbers are <=0.
13. Create a test case to check if Division method gives expected result if both numbers are >0.
14. Create a test case to check if Division method gives “Invalid Input” output if any of the numbers are <=0.
15. Create a test case to check if the result for valid input is presented in an h2 tag.
16. Create a test case to check if the invalid message for invalid input is presented in an h2 tag.
17. Run the command ng test to execute your test.
18. Open the Karma window and click the debug button to get your jasmine test results. Sample output is given for your reference.



