

## ANGULAR WEEK-3 ASSIGNMENT

1. Develop a Angular Project that show cases One way and two way data binding (Property binding and Event Binding).

### app.component.html

```
<h1><b>.....PROPERTY BINDING.....</b></h1>
<h3>WELCOME...<span [innerHTML]='firstname'></span></h3>
<h1><b>.....EVENT BINDING.....</b></h1>
<button (click)="onClick()">Click me</button>
<h1><b>.....TWO WAY BINDING.....</b></h1>
Enter the value : <input [(ngModel)] ='data1'>
<br>
        Entered value : Hi  {{data1}}
```

### app.component.ts

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

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

export class AppComponent {
  firstname: string = "SAI LAKSHMI";
  onClick(): void {
    console.log('you clicked me!!');
    let data1: String='';
  }
}
```

### app.component.ts

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
```

```
import { AppRoutingModuleModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { FormsModule } from '@angular/forms';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModuleModule,
    FormsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})

export class AppModule { }
```

## OUTPUT:

.....PROPERTY BINDING.....

WELCOME...SAI LAKSHMI

.....EVENT BINDING.....

.....TWO WAY BINDING.....

Enter the value : sai  
Entered value : Hi sai

Angular is running in development mode. Call enableProdMode() to enable production mode.

[WDS] Live Reloading enabled.

you clicked me!!

2. Develop a Angular Project that show cases the following Event binding :- a) Text Box's key entry c) Text Box Key Enter d) Text Area Keyup + Shift + A b) Button on Click

### app.component.html

```
<!-- Text Box's key entry -->
<input (keyup)="onkey($event)" />
    <br>
    <p>This is box key entry value is {{name}}</p>
```

```

<!-- Text Box key enter -->

    <input #box (keyup.enter)="onenter(box.value)"/>
    <br>
    <p>This is box key enter value is {{value}}</p>

<!-- Text area keyup + shift + A -->
    <input #box1 (keydown.Shift.A)="onenter1(box1.value)"/>
    <br>
    <p>This is box keyup+shift+a is {{val}}</p>
    <!-- Button on click -->

    <button (click)="Hit()">Submit</button>

```

## app.component.ts

```

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'angularAssignment1';
  name:any='';
  onkey(event:any){
    this.name=event.target.value;
  }

  value:any='';
  onenter(value:string){
    this.value=value;
  }

  val:any='';
  onenter1(key:string){
    this.val=key;
  }

  User:any;
  Hit(){
    this.User='This is button click event';
  }
}

```

3. Develop a Angular project that uses template [reference](#) variable for text box, text area and select drop down control

app.component.html

```
<div class="container">
  <div>
    <select #select [(ngModel)]="selectedcity" (change)="log(select.value)">
      <option *ngFor="let city of cities" [value]="city.id">{{city.name}}</option>
    </select>
  </div>
  <br>
  <div>
    <button (click)="Str=''>Clear</button>
  </div>

  {{Str}}
</div>
```

app.component.ts

```
import { stringify } from '@angular/compiler/src/util';
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'angularAssignment1';
  selectedcity:number=0;

  public cities:any=[
    {
      id:1,
      name:'Chennai',
      state:'TN'
    },
    {
      id:2,
      name:'Hyderabad',
      state:'TS'
    },
    {
      id:3,
```

```
name: 'Vizag',  
state: 'AP'
```

4. Develop a Angular Project with one child control. And pass data from parent control (ex app.component.ts) to the child data via property binding

app.component.ts

```
import { BrowserModule } from '@angular/platform-browser';  
import { NgModule } from '@angular/core';  
import { FormsModule } from '@angular/forms';  
  
import { AppRoutingModuleModule } from './app-routing.module';  
import { AppComponent } from './app.component';  
import { ChildComponent } from './child/child.component';  
  
@NgModule({  
  declarations: [  
    AppComponent,  
    ChildComponent  
  ],  
  imports: [  
    BrowserModule,  
    AppRoutingModuleModule,  
    FormsModule  
  ],  
  providers: [],  
  
  bootstrap: [AppComponent]  
})  
export class AppModule { }
```

app.component.html

```
<app-child *ngFor="let user of Users" [value1]="user.fname"></app-child>  
child.component.html
```

```
<p>The selected city is {{value1}}</p>
```

child.component.ts

```
import { Component, Input, OnInit } from '@angular/core';  
  
@Component({
```

```

    selector: 'app-child',
    templateUrl: './child.component.html',
    styleUrls: ['./child.component.css']
  })

  export class ChildComponent implements OnInit {

    constructor() { }

    @Input()
    public value1:String;

    ngOnInit(): void {
    }

  }

```

app.module.ts

```

import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';

import { AppRoutingModuleModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { ChildComponent } from './child/child.component';

@NgModule({
  declarations: [
    AppComponent,
    ChildComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModuleModule,
    FormsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

5. Develop a Angular Project that displays a list of Products details (id, name,description,cost) in a grid format. The solution should be able to display unique identifier for each record, and able to show even and odd record with efficient data update technique.

app.component.html

```
<div class="container mt-5">
  <table class="table">
    <tr>
      <th>ID</th>
      <th>Product</th>
      <th>Description</th>
      <th>Cost</th>
      <th>Even</th>
      <th>Odd</th>
    </tr>
    <tr *ngFor="let person of persons;even as e;odd as o">
      <td>{{person.Id}}</td>
      <td>{{person.Product}}</td>
      <td>{{person.Description}}</td>
      <td>{{person.Cost}}</td>
      <td>{{e}}</td>
      <td>{{o}}</td>
    </tr>
  </table>
<div class="row">
  <div class="col">
    <input type="number" class="form-control" placeholder="Id" [(ngModel)]="Id">
  </div>
  <div class="col">
    <input type="text" class="form-
control" placeholder="Product Name" [(ngModel)]="Product">
  </div>
  <div class="col">
    <input type="text" class="form-
control" placeholder="Description" [(ngModel)]="Description">
  </div>
  <div class="col">
    <input type="text" class="form-control" placeholder="Cost" [(ngModel)]="Cost">
  </div>
  <div class="col">
    <button class="btn btn-primary btn-
block" (click)="add(Id,Product,Description,Cost)">Add</button>
  </div>
</div>
</div>
```

app.component.ts

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'angularAssignment1';
  persons = [

  ];

  Id:number;
  Product: String;
  Description:String;
  Cost:String;

  add(Id,Product,Description,Cost){
    this.persons.push({
      "Id":Id,
      "Product": Product,
      "Description":Description,
      "Cost":Cost

    });

    // resetting the input field

    this.Id = undefined;
    this.Product = "";
    this.Description="";
    this.Cost="";
  }
}
```



6. Develop a Angular Project that displays a list of countries(ex India.Singapore,USA,UK) in a drop down, based on the selected country we need to display specific contents to the user. If India is selected, the page should display a section asking the user to enter the name and aadhaar number, if US is selected the page should display a section asking the user to enter SSN and Passport number, in case of UK, the user should see the section to enter only the password.

```
<div class="container p-20">
  <h3>Please select the country:</h3>
  <select [(ngModel)]="k">
    <option *ngFor="let cnt of countries" [value]="cnt.Id">{{cnt.Country}}</option>
  </select>
</div>
<br>
<div class="container">
  <div [ngSwitch]="k">
    <div *ngSwitchCase="1">
      <div class="container">
        <div>
          <label>Enter the name:</label>
          <input type="text" class="form-control" placeholder="Username">
        </div>
        <div>
          <label>Enter the Aadhar number:</label>
          <input type="text" class="form-control" placeholder="Aadhar">
        </div>
      </div>
    </div>
    <div *ngSwitchCase="2">
      <div class="container">
        <div>
          <label>Enter the SSN number:</label>
          <input type="number" class="form-control" placeholder="SSN number">
        </div>
        <div>
          <label>Enter the passport number:</label>
          <input type="number" class="form-control" placeholder="Passprt number">
        </div>
      </div>
    </div>
    <div *ngSwitchCase="3">
      <div class="container">
        <div>
          <label>Enter the Password:</label>
          <input type="password" class="form-control" placeholder="Password">
        </div>
      </div>
    </div>
  </div>
</div>
```

app.component.ts

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'angularAssignment1';
  k:number=0
  countries=[
    {Id:1,Country:"India"},
    {Id:2,Country:"AUSTRALIA"},
    {Id:3,Country:"JAPAN"},
  ]
}
```