Fast Learner Activity

Name:Aditya Salunke
UCID:2022510053
Subject : Web Technology
Class:FY MCA

A basic angular app with routing and data passing functionalities.

Create new app
ng new (your-app-name)

```
C:\Users\Aditya\Downloads\Fast Learner>ng new fastL
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE fastL/angular.json (2606 bytes)
```

Generate two components , here we generate comp1 and comp2

ng g c (your-component-name)

```
C:\Users\Aditya\Downloads\Fast Learner\fastL>ng g c comp1
CREATE src/app/comp1/comp1.component.html (20 bytes)
CREATE src/app/comp1/comp1.component.spec.ts (592 bytes)
CREATE src/app/comp1/comp1.component.ts (198 bytes)
CREATE src/app/comp1/comp1.component.css (0 bytes)
UPDATE src/app/app.module.ts (471 bytes)

C:\Users\Aditya\Downloads\Fast Learner\fastL>ng g c comp2
CREATE src/app/comp2/comp2.component.html (20 bytes)
CREATE src/app/comp2/comp2.component.spec.ts (592 bytes)
CREATE src/app/comp2/comp2.component.ts (198 bytes)
CREATE src/app/comp2/comp2.component.css (0 bytes)
UPDATE src/app/app.module.ts (549 bytes)
```

Start app

ng serve

```
C:\Users\Aditya\Downloads\Fast Learner\fastL>ng serve
✓Browser application bundle generation complete.

Initial Chunk Files    | Names      | Raw Size
vendor.js              | vendor     |   2.04 MB |
polyfills.js           | polyfills  | 314.26 kB |
styles.css, styles.js  | styles     | 209.39 kB |
main.js                | main       |   8.98 kB |
runtime.js             | runtime    |   6.51 kB |

                       | Initial Total |  2.56 MB
```
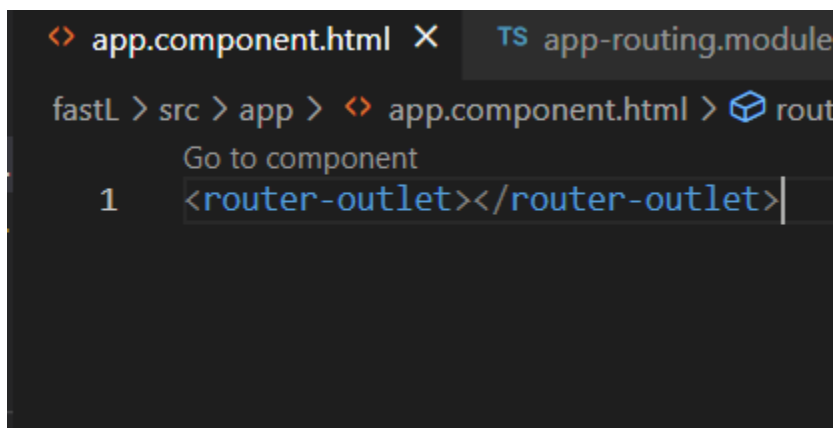
Routing

Add the lines highlighted below comments to create a route in app-routing.module.ts

```typescript
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { Comp1Component } from './comp1/comp1.component';
import { Comp2Component } from './comp2/comp2.component';
const routes: Routes = [
  {
    // add Path here , we set default to load comp1
    path:"",component:Comp1Component,
  },
  {
    // we add another path which will component 2,
    path:"comp2",component:Comp2Component

  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

 In app.component.html

Type <router-outlet> tag

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

How to setup NgModel

In app.module.ts add the lines specified in comments

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


//----------------------------------------------------//
//import FormsModule //
import { FormsModule } from '@angular/forms';
//----------------------------------------------------//

import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { Comp1Component } from './comp1/comp1.component';
import { Comp2Component } from './comp2/comp2.component';


@NgModule({
  declarations: [
    AppComponent,
    Comp1Component,
    Comp2Component
  ],
  imports: [
    //-----------------------------------//
    //add it here//
    FormsModule,
    //-----------------------------------//
    BrowserModule,
    AppRoutingModule
```

// add the following line specified in comment in com1.component.ts

```typescript
astL > src > app > comp1 > TS comp1.component.ts > Comp1Component
1    import { Component } from '@angular/core';
2
3    @Component({
4      selector: 'app-comp1',
5      templateUrl: './comp1.component.html',
6      styleUrls: ['./comp1.component.css']
7    })
8
9    export class Comp1Component {
10     // we create variable of datatype string in here in comp1 component //
11     textData:String = "";
12
13
14   }
15
```

Add this line in comp1.component.html

```html
<!-- add this line -->
<input type="text" [(ngModel)]="textData">
```

Now we can access the input data that user may enter and we store it in out textData variable, please do make sure both the variable name are same

Data Passing

How to create a Service

ng g s (your-service-name)

```
C:\Users\Aditya\Downloads\Fast Learner\fastL>ng g s PassData
CREATE src/app/pass-data.service.spec.ts (368 bytes)
CREATE src/app/pass-data.service.ts (137 bytes)
```

We will use this service to pass data from input field of comp1 component (for which we set up ngmodel earlier) to comp2 component

Add this code in newly generated service

Here we create two function which set data and pass the data wherever it may be required

```typescript
import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root'
})
export class PassDataService {

  constructor() { }
  data:String = "dummy";
   setData(input:String){
    console.log(input);
     this.data = input;
  }

   getData(){
     return this.data;
   }
}
```

Put below code in comp1 component (ts) file

Here we are creating a function which itself would call the service function (setData) function

```ts
export class Comp1Component {

    constructor( private _passData:PassDataService){

    }

    // we create variable of datatype string in here in comp1 component //
    textData:String = "test1";


    getInputFromUser(){
        console.log(this.textData);
        this._passData.setData(this.textData)

    }

}
```

In comp2 component add this code in (ts) file

```ts
import { Component } from '@angular/core';
import { PassDataService } from '../pass-data.service';

@Component({
    selector: 'app-comp2',
    templateUrl: './comp2.component.html',
    styleUrls: ['./comp2.component.css']
})
export class Comp2Component {
    testData2:String = "";
    constructor(private passData:PassDataService){
        this.testData2 = passData.getData();
    }
}
```

Add this code in comp2 html file

```
app.component.html        <> comp2.component.html  ×     <> c
astL > src > app > comp2 > <> comp2.component.html > ...
  1      <h1>{{testData2}}</h1>>
```

PARENT to CHILD data Passing using @INPUT()

Create a new child component (comp2CHILD) for our comp2 component

```
C:\Users\Aditya\Downloads\Fast Learner\fastL>ng g c comp2CHILD
CREATE src/app/comp2-child/comp2-child.component.html (26 bytes)
CREATE src/app/comp2-child/comp2-child.component.spec.ts (628 bytes)
CREATE src/app/comp2-child/comp2-child.component.ts (221 bytes)
CREATE src/app/comp2-child/comp2-child.component.css (0 bytes)
UPDATE src/app/app.module.ts (966 bytes)
```

Add this lines in comp2CHILD component (ts) file

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

@Component({
  selector: 'app-comp2-child',
  templateUrl: './comp2-child.component.html',
  styleUrls: ['./comp2-child.component.css']
})
export class Comp2CHILDComponent {


  @Input() textData :String = "";

}
```

Add this line in comp2.component.html

```html
<h1>Component 2</h1>
<h2>{{testData2}}</h2>>
<!-- add this line in our comp2.component.html -->
<app-comp2-child [textData]="testData2"></app-comp2-child>
```

We are basically the data from the user which is transferred to use through service we created earlier

Then we created another child for comp2

So comp2 would be parent of comp2CHILD

Then we pass data to comp2CHILD using @Input() decorator