Angular Crud tutorial Link <https://www.youtube.com/watch?v=JYPyy-hvjYc&index=1&list=PL6n9fhu94yhXwcl3a6rIfAI7QmGYIkfK5>

Lecture :1

**Installing the tools required**   
  
**Node :** Install the latest version of node. Here is the link to download and install the latest version  
<https://nodejs.org/en/download/>

After install we can verify with node –v Command

**Angular CLI :**Install the latest version of Angular CLI by executing the following command from the windows command prompt.   
npm install -g @angular/cli

For verifying we can check with ng –v

After Create the latest version of Node and Angular CLI installed, launch windows command prompt as an administrator and execute the following command. This creates a new AngularProject with name AngularCrud. We do not want test files to be generated for the root component AppComponent, so we have set "skip-tests" option to true. We will discuss unit testing components in a later video.  
ng new AngularCrud --skip-tests true

We will be using Bootstrap for styles in our application. So install Bootstrap by executing the following command from the command prompt.  
npm install bootstrap@3 --save  
  
Once Bootstrap is installed, open .angular-cli.json file and specify the path to the Bootstrap stylesheet (bootstrap.min.css) in the styles property as shown below. 

"styles": [

  "../node\_modules/bootstrap/dist/css/bootstrap.min.css",

  "styles.css"

]

Lecture: 2 Reading data in angular

I have Created employee model

export class Employee {  
    id: number;  
    name: string;  
    gender: string;  
    email?: string;  
    phoneNumber?: number;  
    contactPreference: string;  
    dateOfBirth: Date;  
    department: string;  
    isActive: boolean;  
    photoPath?: string;  
}

First created component using Angular CLI Command

ng g c employees/listEmployees --spec false --flat true

and also created the employee type array and hard coded the value

employees: Employee[] = [

{…..}

];

Lecture 3: Angular Routing and Navigation

First create the Component ng g c employees/createEmployee --spec false --flat true

Set <base href="/"> in the application host page which is index.html. The <base href> tells the angular router how to compose navigation URLs. This is already done for us by the Angular CLI, when we created this project.

<base href="/">

need to import the Router Module in our AppModule.

import { RouterModule } from '@angular/router';

and also added in imports array

  imports: [

    BrowserModule,

    RouterModule

  ],

Next step: Configure the application routes.

const appRoutes: Routes = [

  { path: 'list', component: ListEmployeesComponent },

  { path: 'create', component: CreateEmployeeComponent },

  { path: '', redirectTo: '/list', pathMatch: 'full' }

];

// pass "appRoutes" constant to forRoot(appRoutes) method

imports: [

  BrowserModule,

  RouterModule.forRoot(appRoutes)

],

Create the application menu and tie the routes to it. Notice we are using routerLink directive. This directive tells the router where to navigate when the user clicks the link.

<div class="container">

    <nav class="navbar navbar-default">

        <ul class="nav navbar-nav">

            <li>

                <a routerLink="list">List</a>

            </li>

            <li>

                <a routerLink="create">Create</a>

            </li>

        </ul>

    </nav>

    <router-outlet></router-outlet>

</div>

Lecture 4 : Angular base href

During development we usually set this to a single forward slash as shown below.   
<base href="/">

This means all the URLs now will be relative to the root of the application. So when we navaigate to "/list", the path "/list" will be appended to root UR and the complete URL will be as shown below. Notice "/list" is relative to the root URL.  
<http://localhost:4200/list>

When we deploy our application to a server, we typically deploy it to a sub folder on the server. For example, if we are deploying our application in a sub-folder called **"emp"**, then we set the base href element to /emp/ as shown below.  
<base href="/emp/">

During development, if you want a different base path other than "/", simply execute the "ng serve" command with --base-href option set to your desired base path as shown below.  
ng serve -o --base-href /emp/