Core concepts of Angular

1. Module
2. Components
3. Directives
4. Pipes
5. Services

Module: General Concepts of Module.

Scenario: You are developing Ecommerce application where we will have to implement number of functionalities like as admin we should be able to add products, update product, delete product get all product or get single product information and as a customer we can explore products, do purchase then if we are going to implement these all functionalities in one class file or in an entity it will become very complex so, in order to make it's implementation we can divide this application in different modules like product module, customer module, purchase module.

Let’s consider

Product Module:

a. Add product
b. Delete product
c. Update product
d. Get All product
e. Get product by ID

Customer Module:

a. Customer Registration
b. Customer Login
c. Customer Forget password
d. Customer Feedback

Purchase:

a. View Purchase details
b. Add in cart

So, next time wherever we want any of this functionality then we can import it to use it.

When we talk about module in angular more precisely, we should know that Angular is **modular-based architecture.**

In angular, we have inbuilt module like

1. BrowserModule
2. FormModule
3. HttpClientModule and many more

   but we can also create our own modules as well.

In simple words, we can say Module is group of functionalities.

Switch to ppt slide 35

Note: Modules in Angular also helps us to optimized and speedup our application. Modules also helps to structure our application properly and easier to maintain.

Here comes the question that how Module is going to help us to optimise our application?

So, we have seen a root module of angular application that is app.module.ts. Whatever we are using like components, pipes, directives or services these all we will have to register in app.module.ts file.

Internally when we register components, pipes, directives, services then modules tells angular that in our application we are using these things. So, here angular won't scan for all stuffs. It will just check for those things which we have registered in our module and angular use only those components and build application.

When we are going to add all these components, pipes, directives and services in app.module.ts then our app.module.ts file become very lengthy. It's fine to have such long file but as I told you we will have to optimize our code and maintain readability. For this purpose we can create multiple modules and divide functionalities.

Open slide 39 and follow steps

Components:

Read all points from slide number 40.

How we can conclude components in Angular?

1. Components are the most important and basic building blocks of Angular apps
2. Component is a smallest functionality that you will implement in your application
3. When we group multiple Components it becomes a module
4. We can have parent-child relationship of components
5. Every component has 4 files auto-generated with it
   a. - component.html -> view or html or template file -> UI

b. - component.ts -> it will be a class file which will have methods -> Logic

c. - component.spec.ts -> It will have the unit test script for component

d. - component.scss -> stylesheet of the component

Component decorator inside the component.ts file

selector ->

unique identifier for the component -> id of the component -> using this selector we will use the component

templateUrl ->

your HTML code

- component.html file

styleURLS -> for linking your component stylesheet - component.scss

---

ng g m customer

CREATE src/app/customer/customer.module.ts (194 bytes)

PS C:\Users\BhushanP5\Desktop\LastTry> ng g c customer/reg

CREATE src/app/customer/reg/reg.component.html (18 bytes)

CREATE src/app/customer/reg/reg.component.spec.ts (578 bytes)

CREATE src/app/customer/reg/reg.component.ts (190 bytes)

UPDATE src/app/customer/customer.module.ts (266 bytes)

PS C:\Users\BhushanP5\Desktop\LastTry> ng g c customer/login

CREATE src/app/customer/login/login.component.html (20 bytes)

CREATE src/app/customer/login/login.component.spec.ts (592 bytes)

CREATE src/app/customer/login/login.component.ts (198 bytes)

PS C:\Users\BhushanP5\Desktop\LastTry>

Open customer.module.ts

```
import { NgModule } from '@angular/core';
import { CommonModule } from '@angular/common';
import { RegComponent } from './reg/reg.component';
import { LoginComponent } from './login/login.component';
import { ForgetComponent } from './forget/forget.component';



@NgModule({
  declarations: [
```

```
    RegComponent,
    LoginComponent,
    ForgetComponent
  ],
  imports: [
    CommonModule
  ],
  exports:[RegComponent, LoginComponent, ForgetComponent]
})
export class CustomerModule { }
```

open app.module.ts

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { CustomerModule } from './customer/customer.module';


@NgModule({
  declarations: [
    AppComponent,
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    CustomerModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

open index.html

```
<h1>Welcome to Citiustech....</h1>
<app-forget></app-forget>
<app-login></app-login>
<app-reg></app-reg>
```

Directives:

1. Component Directives –

Explain component directive normally first with templateUrl and styleUrl.

Now, do given below changes and explain again

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

@Component({
  selector: 'app-forget',
  // templateUrl: './forget.component.html',
  template:'<h1 style="color: blueviolet;">Forget Password</h1>',
  // styleUrls: ['./forget.component.css']
  styles:['h1 { font-weight: normal; }']
})
export class ForgetComponent {

}
```

2. Structural Directives:

Open Login.component.ts file and add given below code.

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

@Component({
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent {
  customers =['James', 'John', 'Kim','Keanu']
}
```

Open login.component.html and add given below code.

```
<h2>Customers List</h2>
<ul *ngFor="let cust of customers">
  <li>{{cust}}</li>
</ul>
```

Pipe

Open reg.component.ts and add given below code:

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

@Component({
  selector: 'app-reg',
  templateUrl: './reg.component.html',
  styleUrls: ['./reg.component.css']
})
export class RegComponent {
  currentTime = new Date();
  productName ="Iphone 13 Pro Max"

}
```

Open reg.component.html and add given below code.

```
<h2>Product Info {{productName | uppercase}}</h2>
<h2>Purchased on {{currentTime | date:'short'}}</h2>
```