Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat

To exit full screen, press Esc

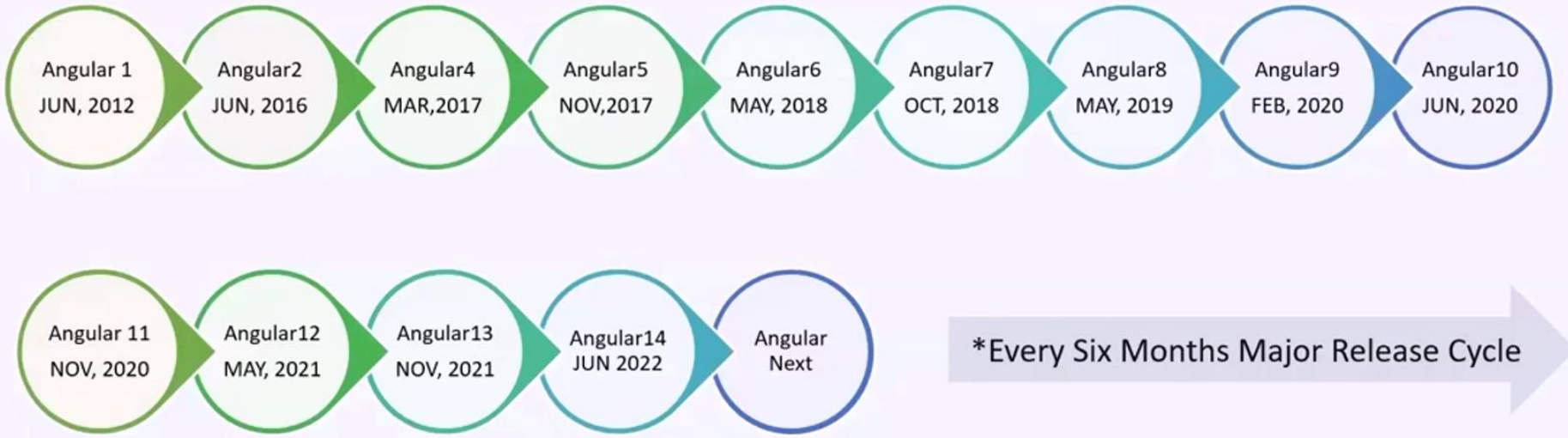
Introduction to Angular

- An open-source framework for building single-page applications (SPA) using web technologies like html, css and js.
- Angular is written in TypeScript and follows TypeScript syntax to write code.
- Empowers developers to build applications for browsers, mobiles, or desktop

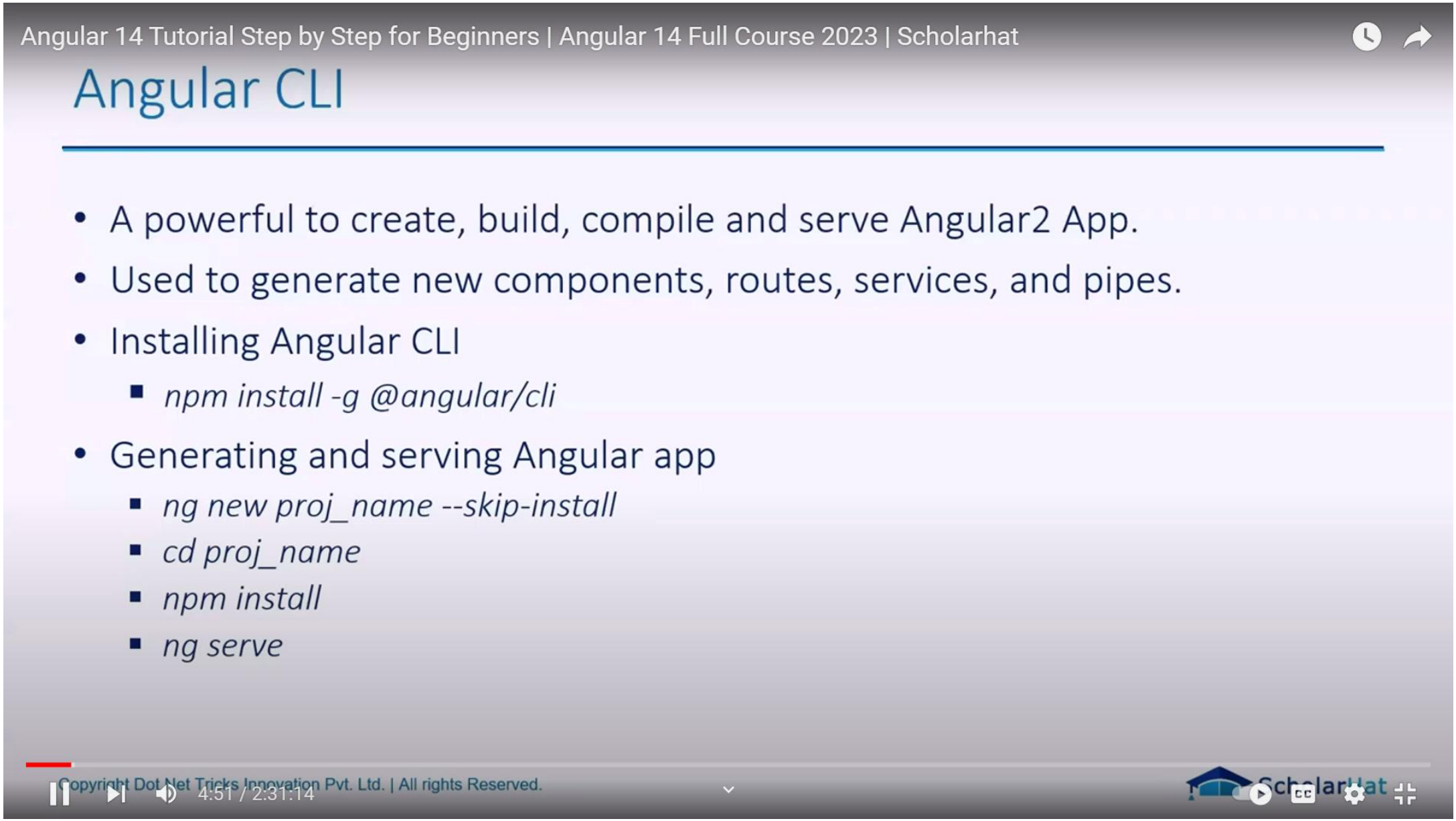
2:00 / 2:31:14

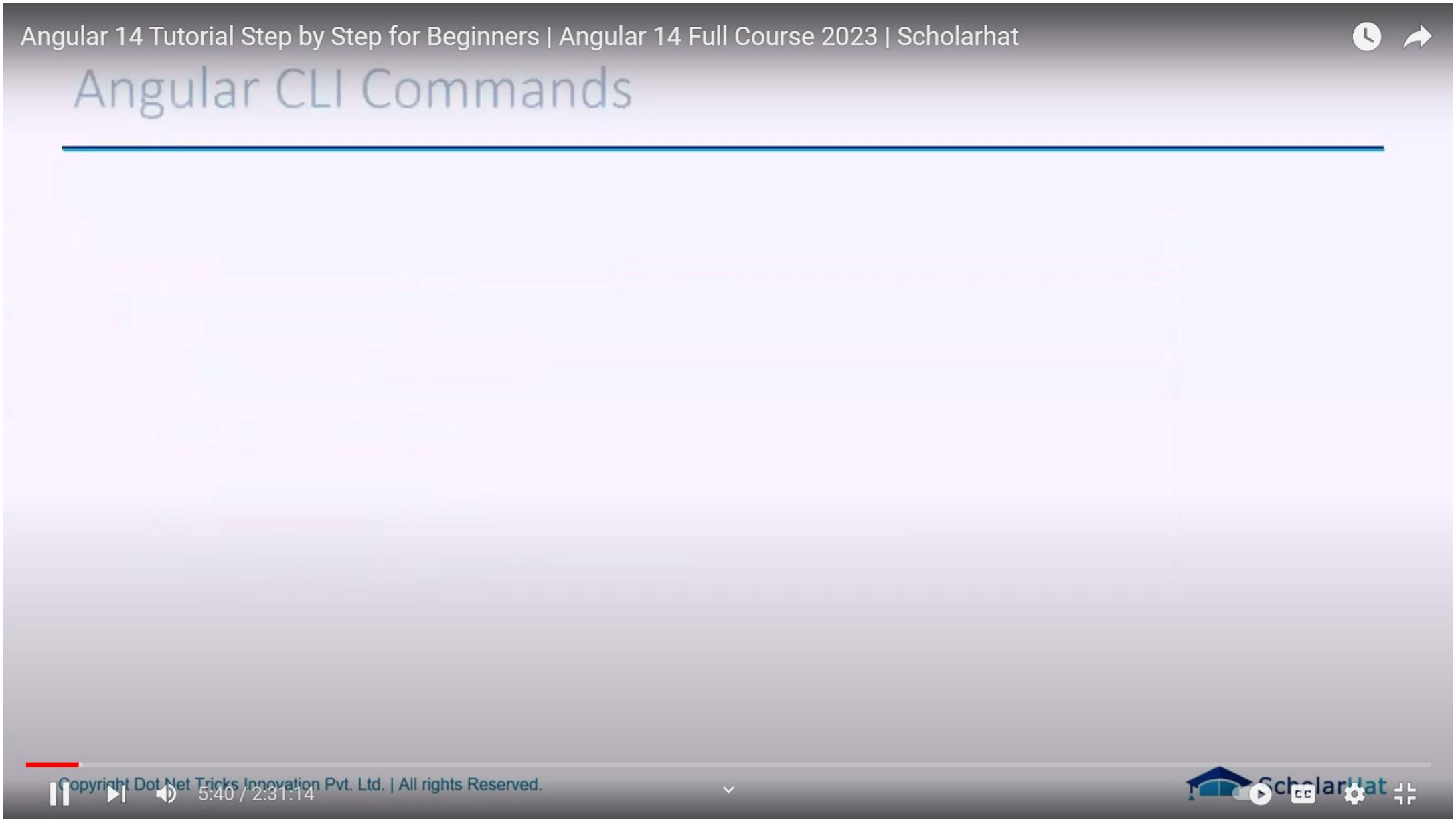
ScholarHat

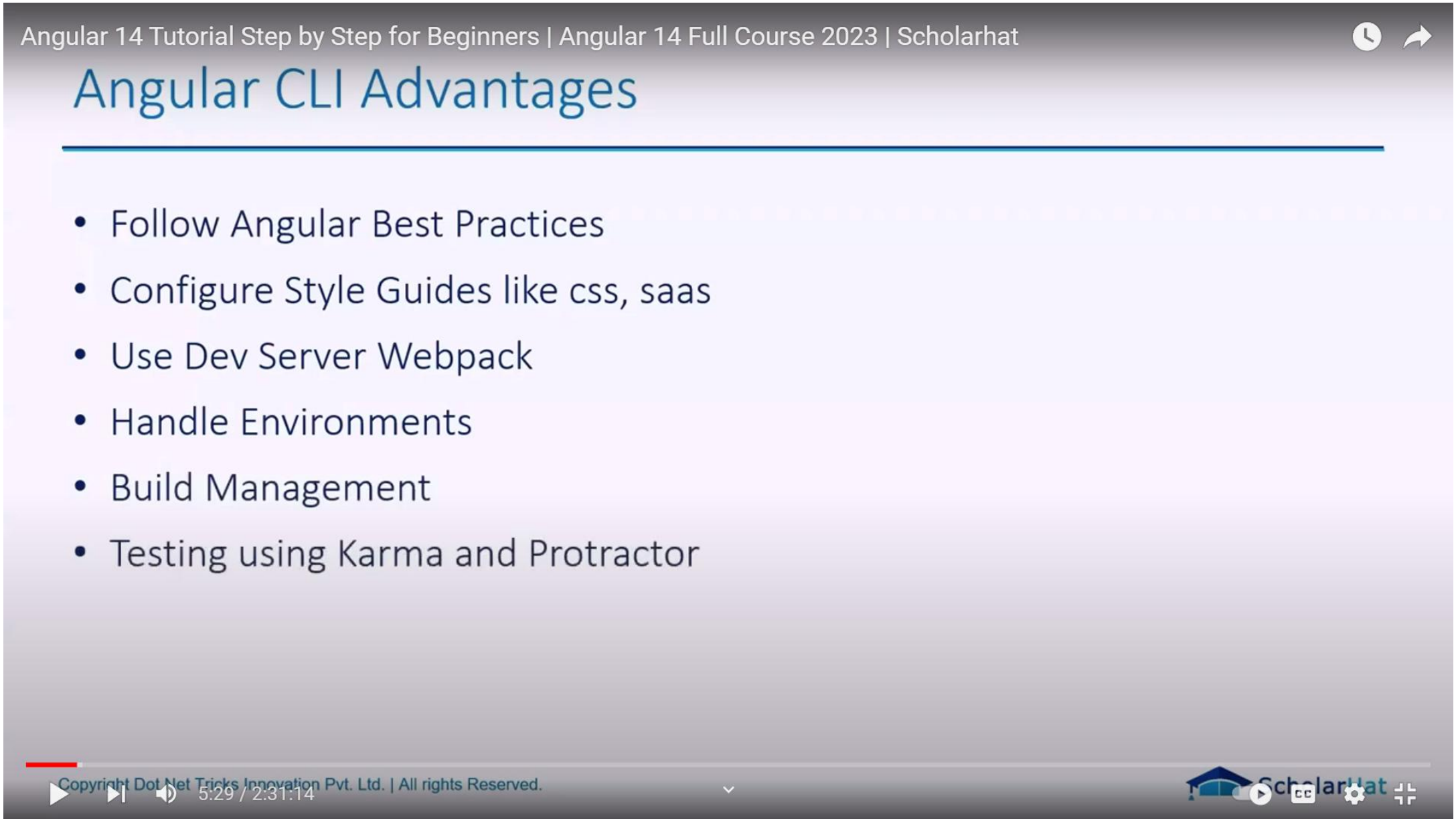
Angular Version History



ScholarHat


Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
Angular CLI
A powerful to create, build, compile and serve Angular2 App.
Used to generate new components, routes, services, and pipes.
Installing Angular CLI
npm install -g @angular/cli
Generating and serving Angular app
ng new proj_name --skip-install
cd proj_name
npm install
ng serve
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
4:51 / 2:31:14
ScholarHat


Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
Angular CLI Commands
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
5:40 / 2:31:14
ScholarHat


Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
Angular CLI Advantages
• Follow Angular Best Practices
• Configure Style Guides like css, saas
• Use Dev Server Webpack
• Handle Environments
• Build Management
• Testing using Karma and Protractor
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
5:29 / 2:31:14
ScholarHat

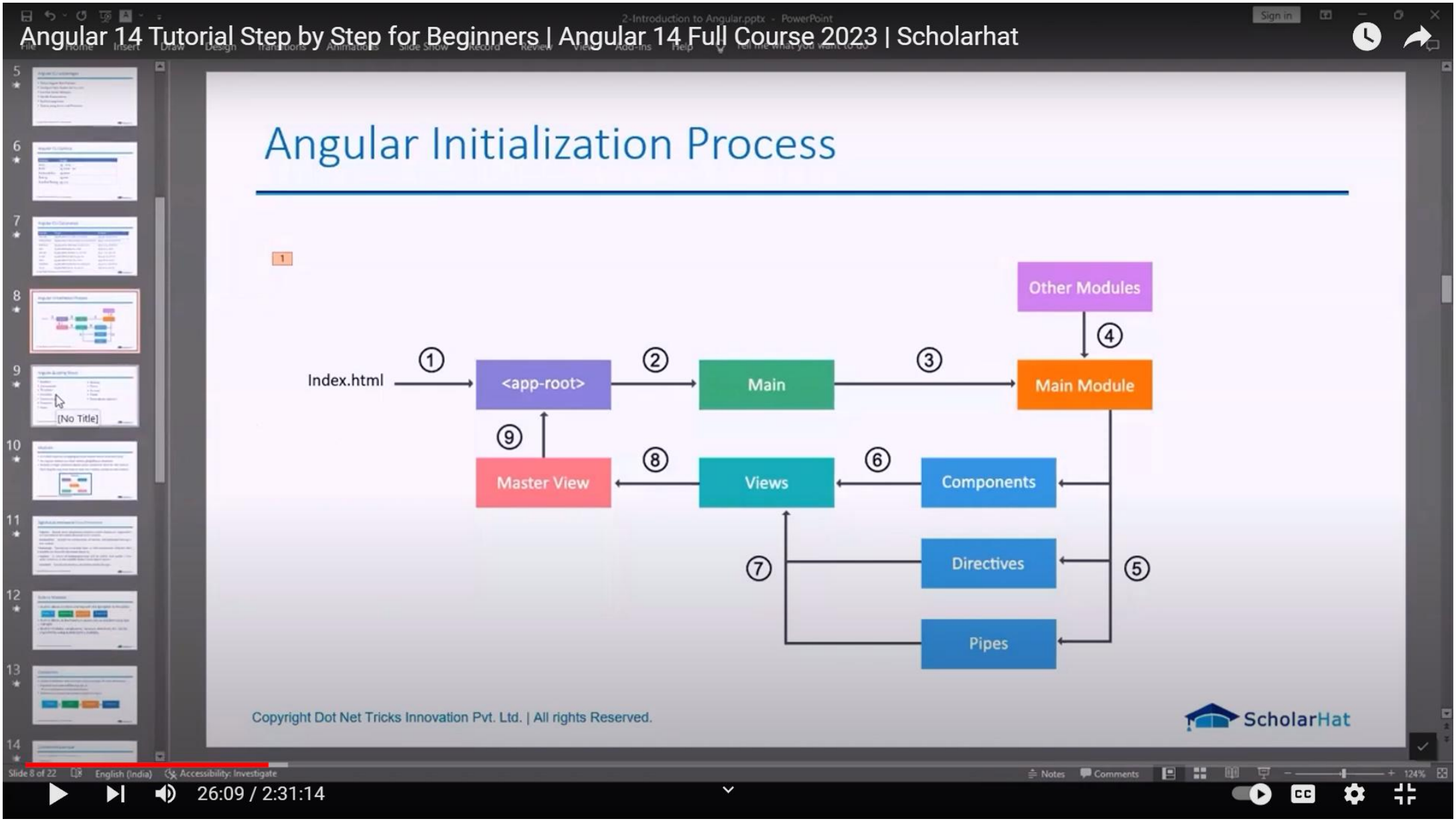
Angular CLI Options

Options	Usage
Help	ng --help
Build	ng build *--env*
Build and Run	ng serve
Testing	ng test
End-End Testing	ng e2e

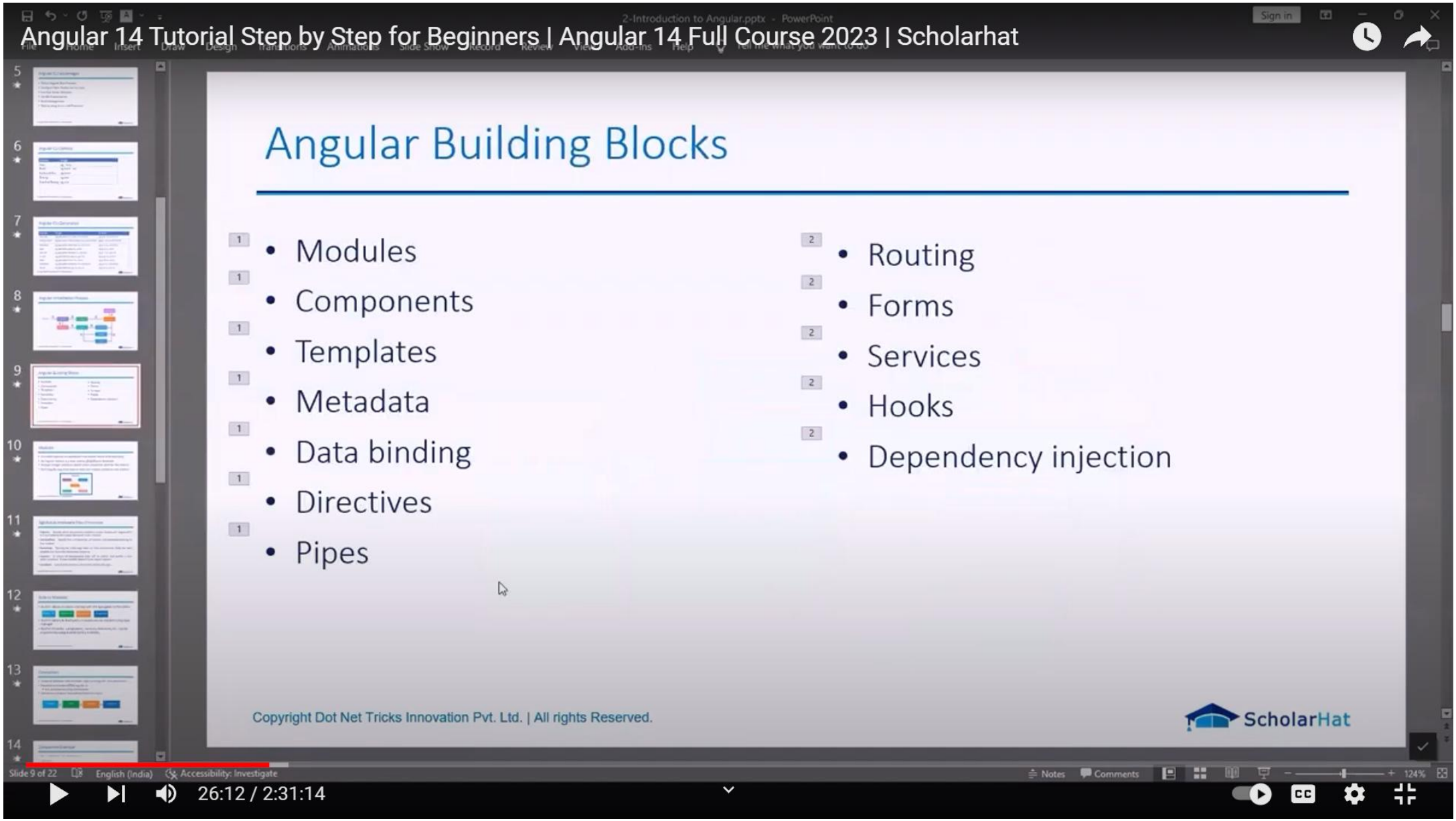
Angular CLI Commands

Scaffold	Usage	In Short
Module	ng generate module *my-module*	ng g m *my-module*
Component	ng generate component *my-component*	ng g c *my-component*
Directive	ng generate directive *my-directive*	ng g d *my-directive*
Pipe	ng generate pipe *my-pipe*	ng g p *my-pipe*
Service	ng generate service *my-service*	ng g s *my-service*
Guard	ng generate guard *my-guard*	ng g g *my-guard*
Class	ng generate class *my-class*	ng g cl *my-class*
Interface	ng generate interface *my-interface*	ng g i *my-interface*
Enum	ng generate enum *my-enum*	ng g e *my-enum*

Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat



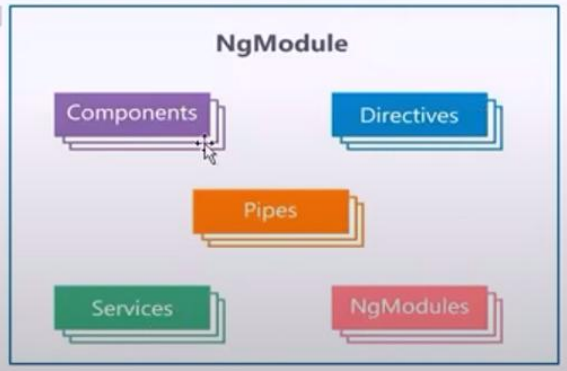
26:09 / 2:31:14


Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
Angular Building Blocks
Modules
Components
Templates
Metadata
Data binding
Directives
Pipes
Routing
Forms
Services
Hooks
Dependency injection
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
26:12 / 2:31:14

Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat

Modules

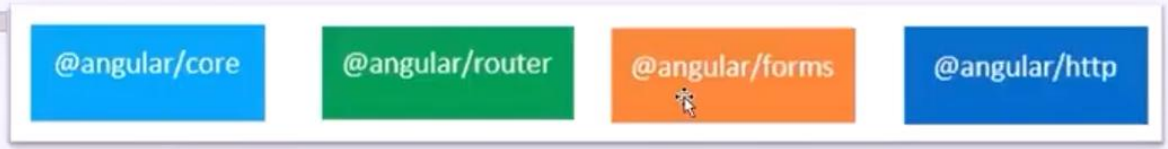
- A module organize an application into unified blocks of functionality
- An Angular module is a class with an *@NgModule* decorator
- Accepts a single metadata object whose properties describe the module
- Each Angular app must have at least one module, known as root module



ScholarHat

Built-In Modules

1
- Built-In library modules starting with the @angular as the prefix.

2


3
- Built-In library & third-party modules can be installed using npm manager.

4
- Built-In modules, components, services, directives, etc. can be imported by using built-In library modules.

ScholarHat

2-Introduction to Angular.pptx - PowerPoint

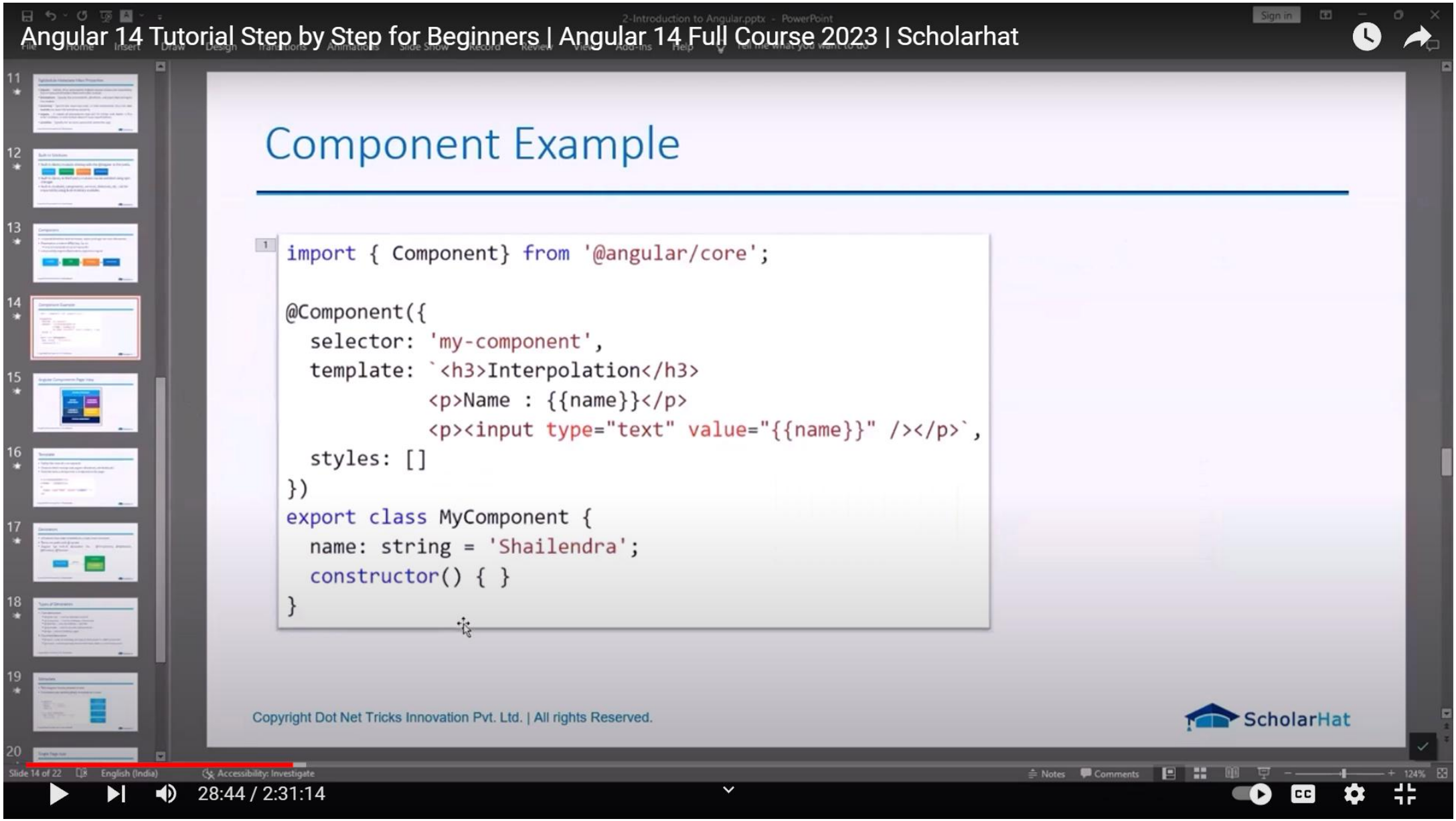
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

Component

- A type of directives with template, styles and logic for user interaction
- Exported as a custom HTML tag like as:
  - <my-component></my-component>
- Initialized by Angular Dependency Injection engine

ScholarHat

Slide 13 of 22 English (India) Accessibility: Investigate Notes Comments 124%

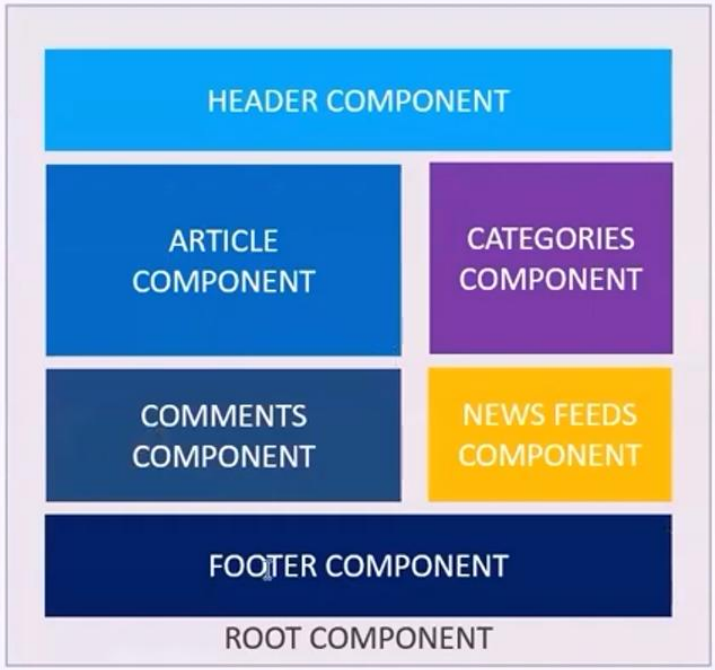

Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
Component Example
import { Component} from '@angular/core';
@Component({
selector: 'my-component',
template: `<h3>Interpolation</h3>
Name : {{name}}
<input type="text" value="{{name}}" /> `,
styles: []
})
export class MyComponent {
name: string = 'Shailendra';
constructor() { }
}
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
28:44 / 2:31:14

2-Introduction to Angular.pptx - PowerPoint

Sign in

File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

Angular Components Page View



ScholarHat

Slide 15 of 22 English (India) Accessibility: Investigate Notes Comments 124%

Template

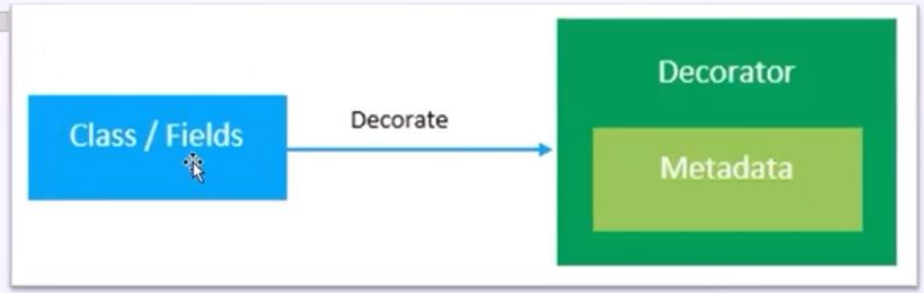
- Define the view of a component.
- Contains Html markup and angular directives, attributes etc.
- Describe how a component is rendered on the page.

```
<h3>Interpolation</h3>
<p>Name : {{name}}</p>
<p>
  <input type="text" value="{{name}}" />
</p>
```

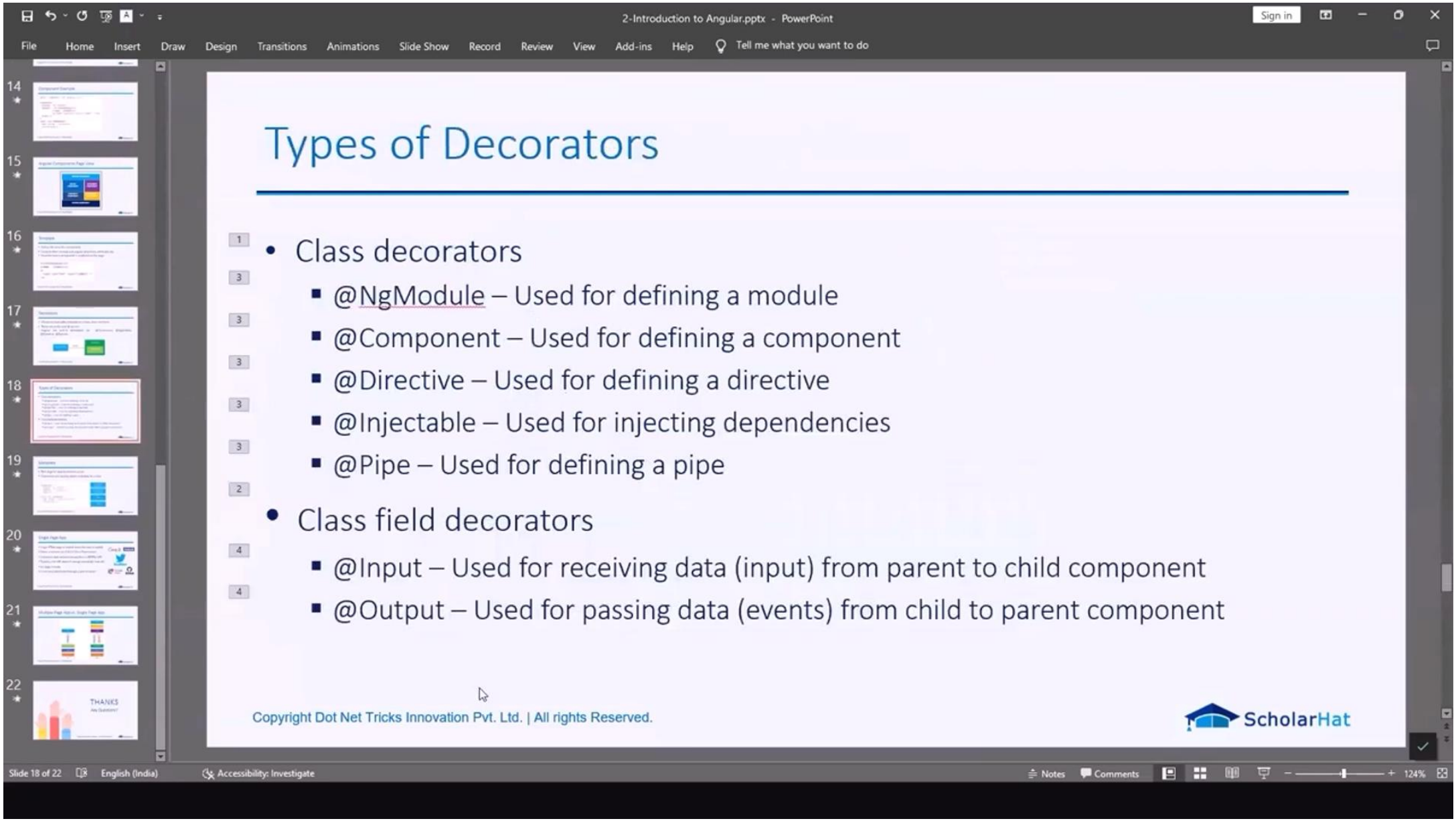

ScholarHat

Decorators

- A function that adds metadata to a class, class members
- These are prefix with @ symbol
- Angular has built-In decorators like - @Component, @NgModule, @Directive, @Pipe etc.



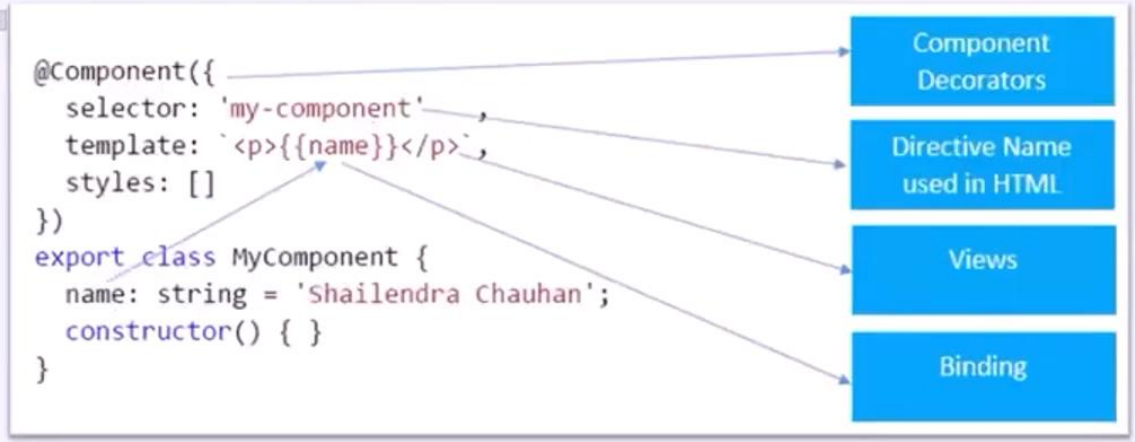
ScholarHat


2-Introduction to Angular.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help
Tell me what you want to do
Types of Decorators
Class decorators
@NgModule – Used for defining a module
@Component – Used for defining a component
@Directive – Used for defining a directive
@Injectable – Used for injecting dependencies
@Pipe – Used for defining a pipe
Class field decorators
@Input – Used for receiving data (input) from parent to child component
@Output – Used for passing data (events) from child to parent component
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 18 of 22
English (India)
Accessibility: Investigate
Notes
Comments
124%

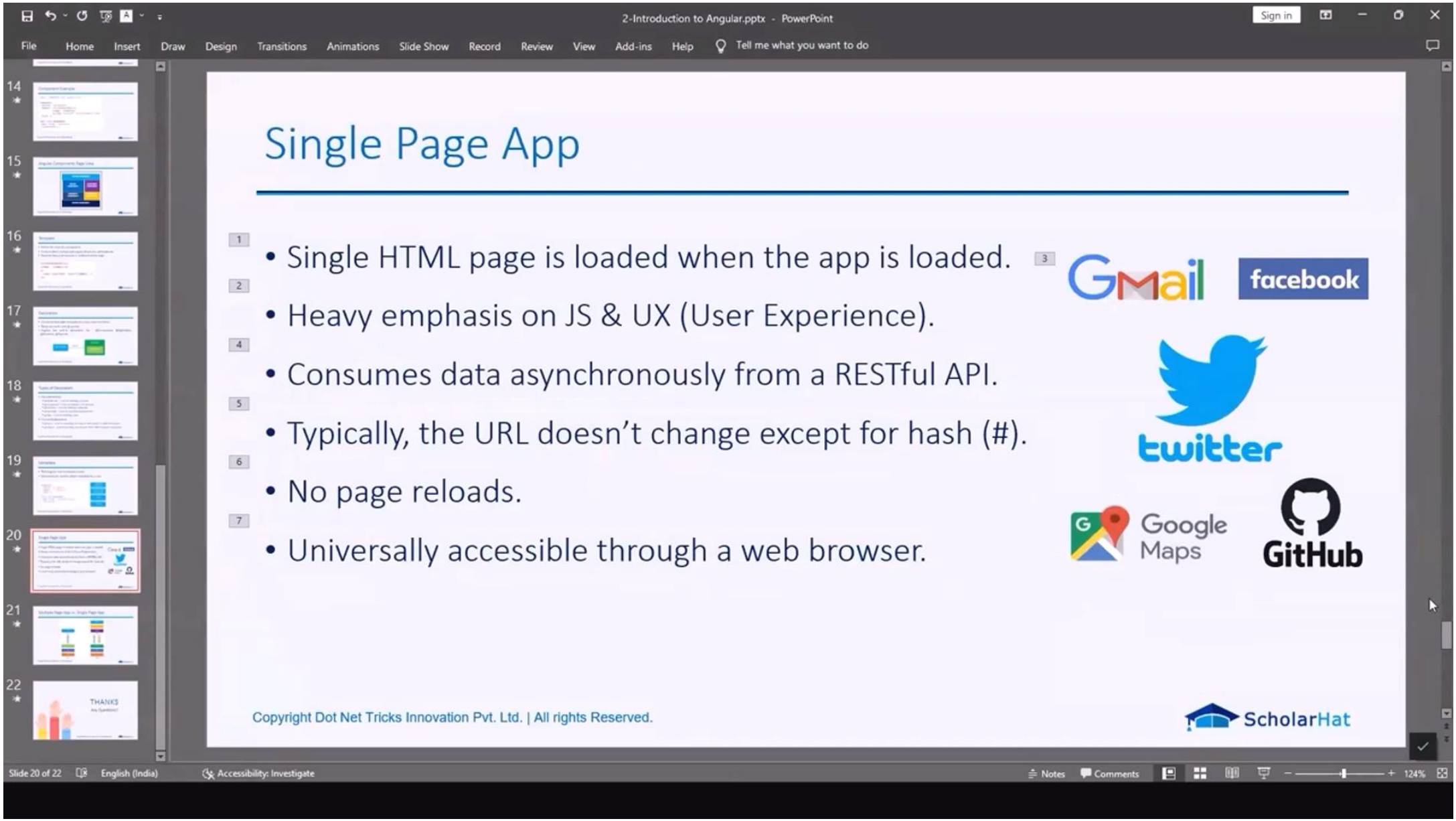
Metadata

- Tells Angular how to process a class
- Decorators are used to attach metadata to a class

```
@Component({
  selector: 'my-component',
  template: `<p>{{name}}</p>`,
  styles: []
})
export class MyComponent {
  name: string = 'Shailendra Chauhan';
  constructor() { }
}
```



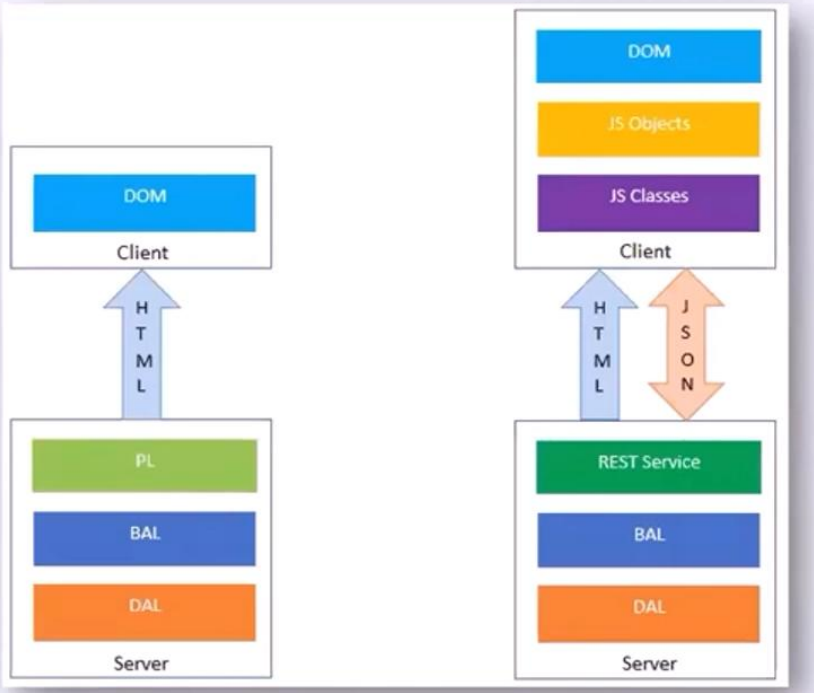
ScholarHat


2-Introduction to Angular.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do
Single Page App
1
• Single HTML page is loaded when the app is loaded.
2
• Heavy emphasis on JS & UX (User Experience).
3
4
• Consumes data asynchronously from a RESTful API.
5
• Typically, the URL doesn't change except for hash (#).
6
• No page reloads.
7
• Universally accessible through a web browser.
Gmail
facebook
twitter
Google Maps
GitHub
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 20 of 22
English (India)
Accessibility: Investigate
Notes
Comments
124%

2-Introduction to Angular.pptx - PowerPoint

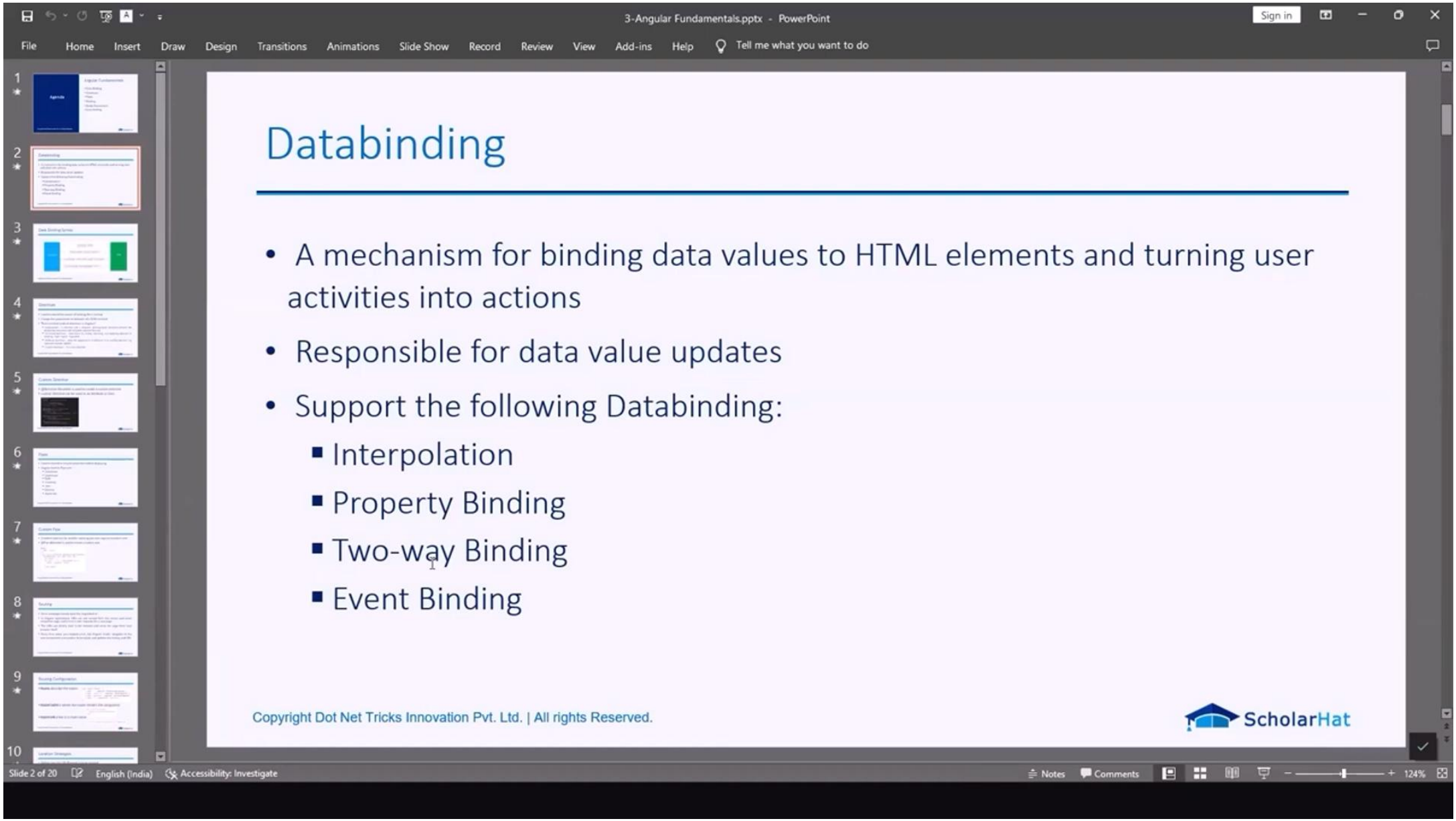
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

Multiple Page App vs. Single Page App

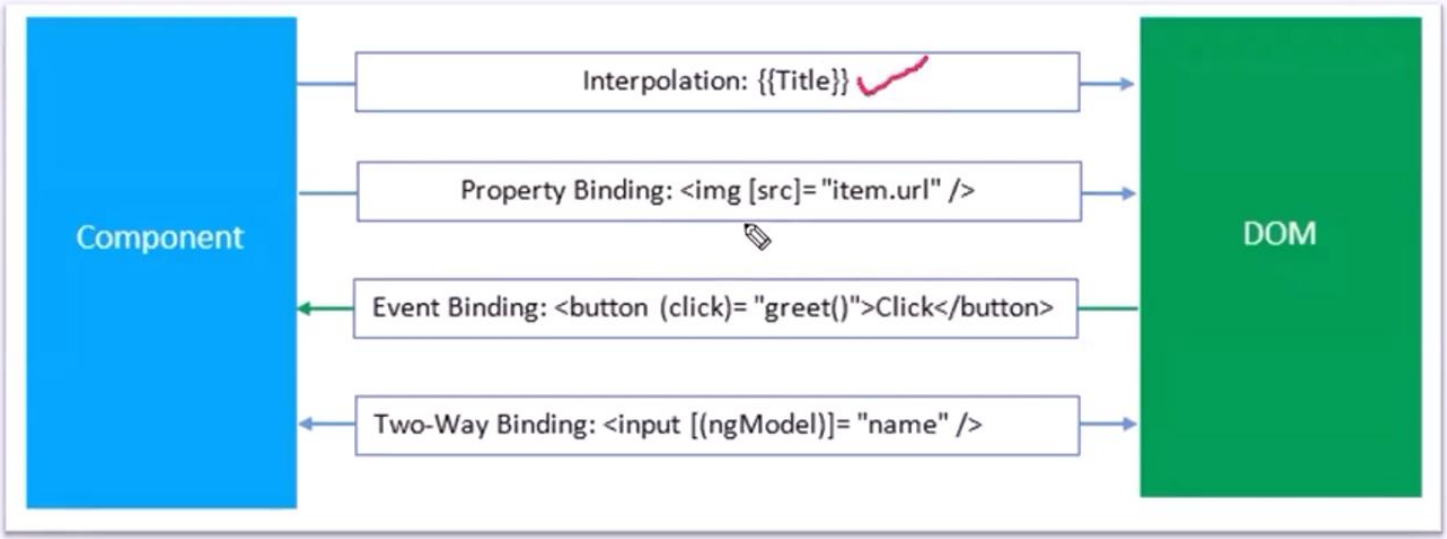


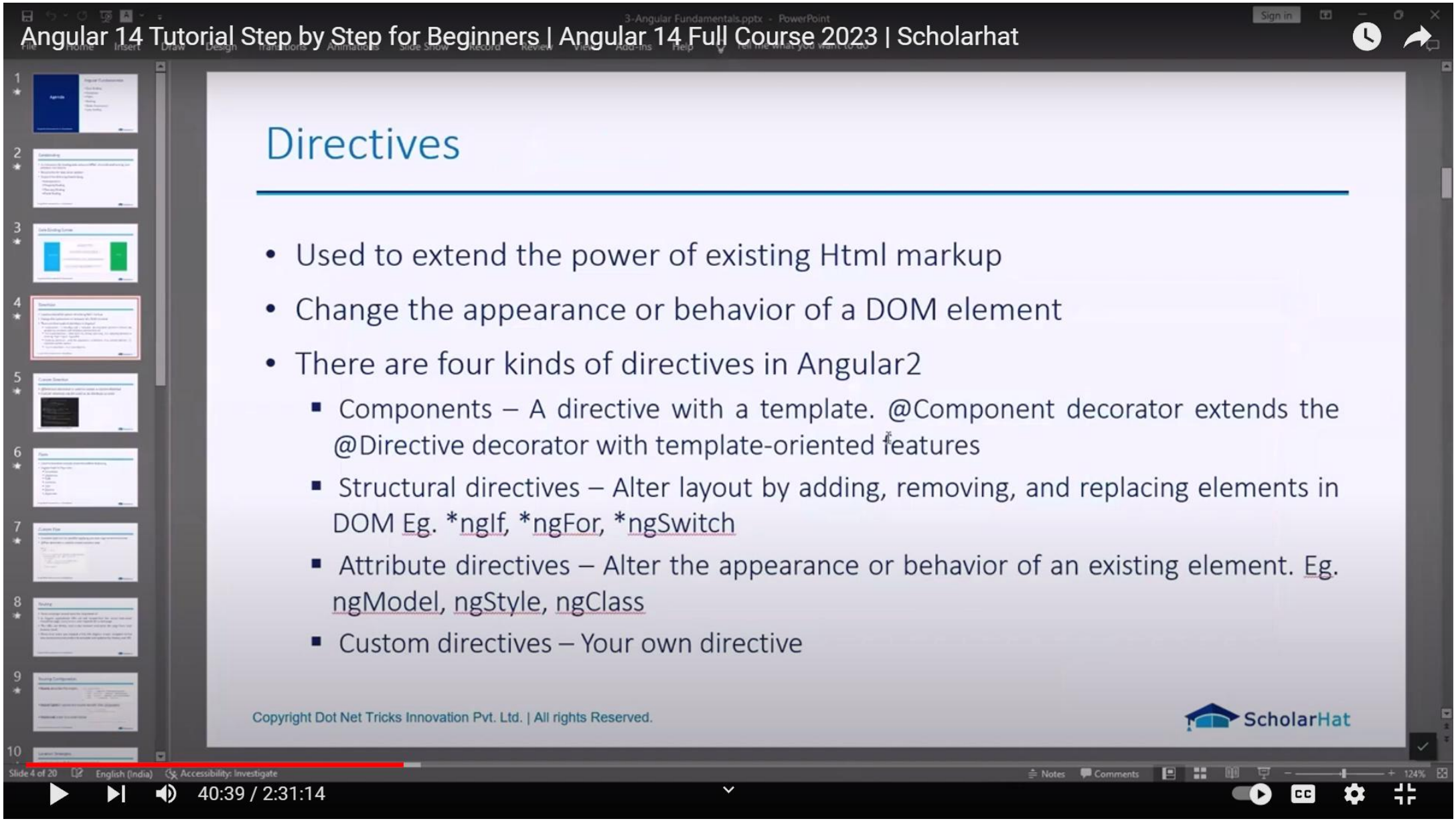
ScholarHat

Slide 21 of 22 English (India) Accessibility: Investigate Notes Comments 124%


3-Angular Fundamentals.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help
Tell me what you want to do
Databinding
• A mechanism for binding data values to HTML elements and turning user activities into actions
• Responsible for data value updates
• Support the following Databinding:
▪ Interpolation
▪ Property Binding
▪ Two-way Binding
▪ Event Binding
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 2 of 20
English (India)
Accessibility: Investigate
Notes
Comments
124%

Data Binding Syntax



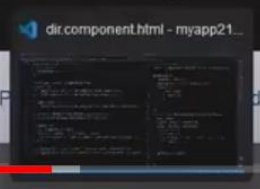

Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
Directives
Used to extend the power of existing Html markup
Change the appearance or behavior of a DOM element
There are four kinds of directives in Angular2
Components – A directive with a template. @Component decorator extends the @Directive decorator with template-oriented features
Structural directives – Alter layout by adding, removing, and replacing elements in DOM Eg. *ngIf, *ngFor, *ngSwitch
Attribute directives – Alter the appearance or behavior of an existing element. Eg. ngModel, ngStyle, ngClass
Custom directives – Your own directive
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
40:39 / 2:31:14

Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat

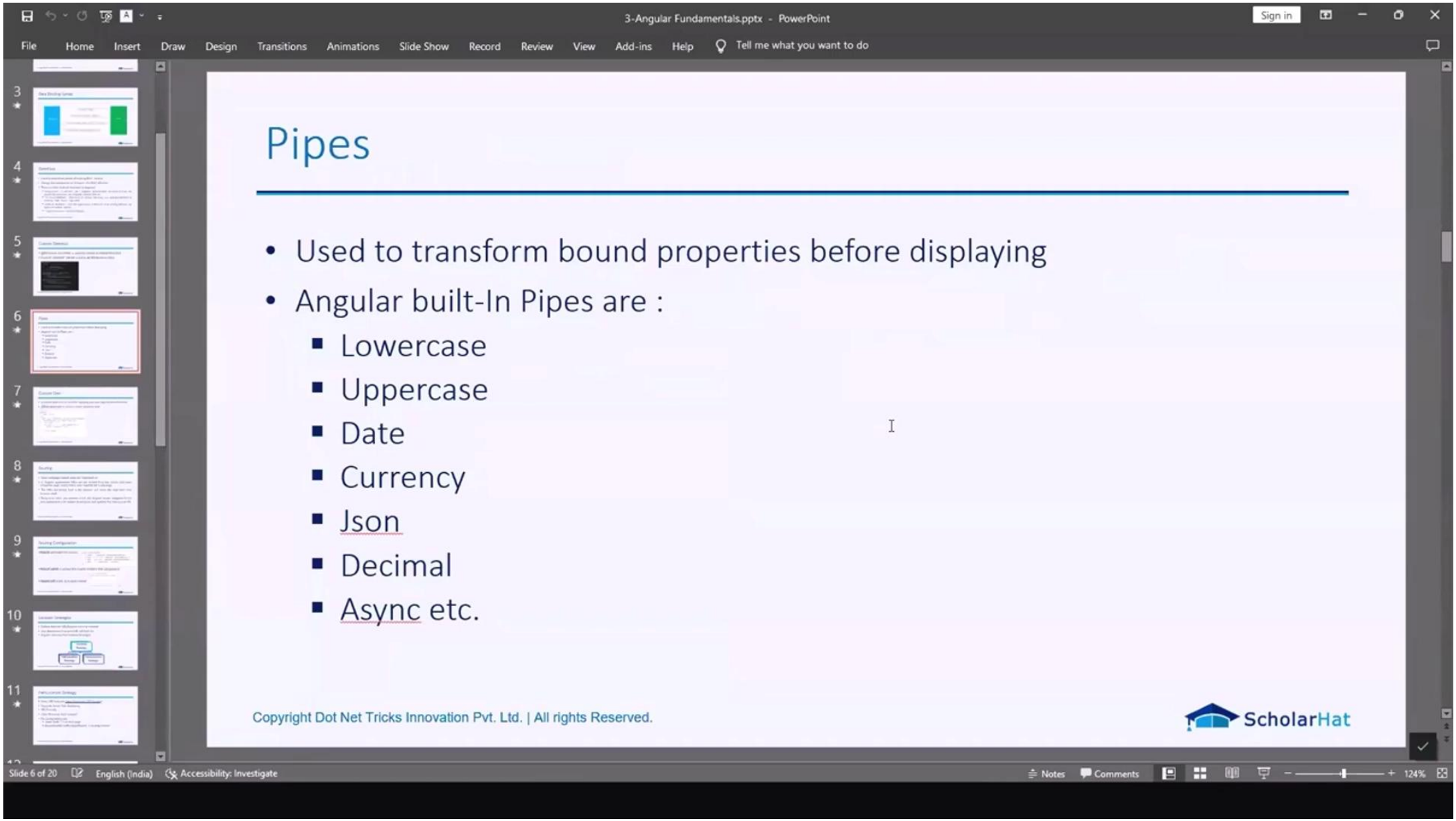
Custom Directive

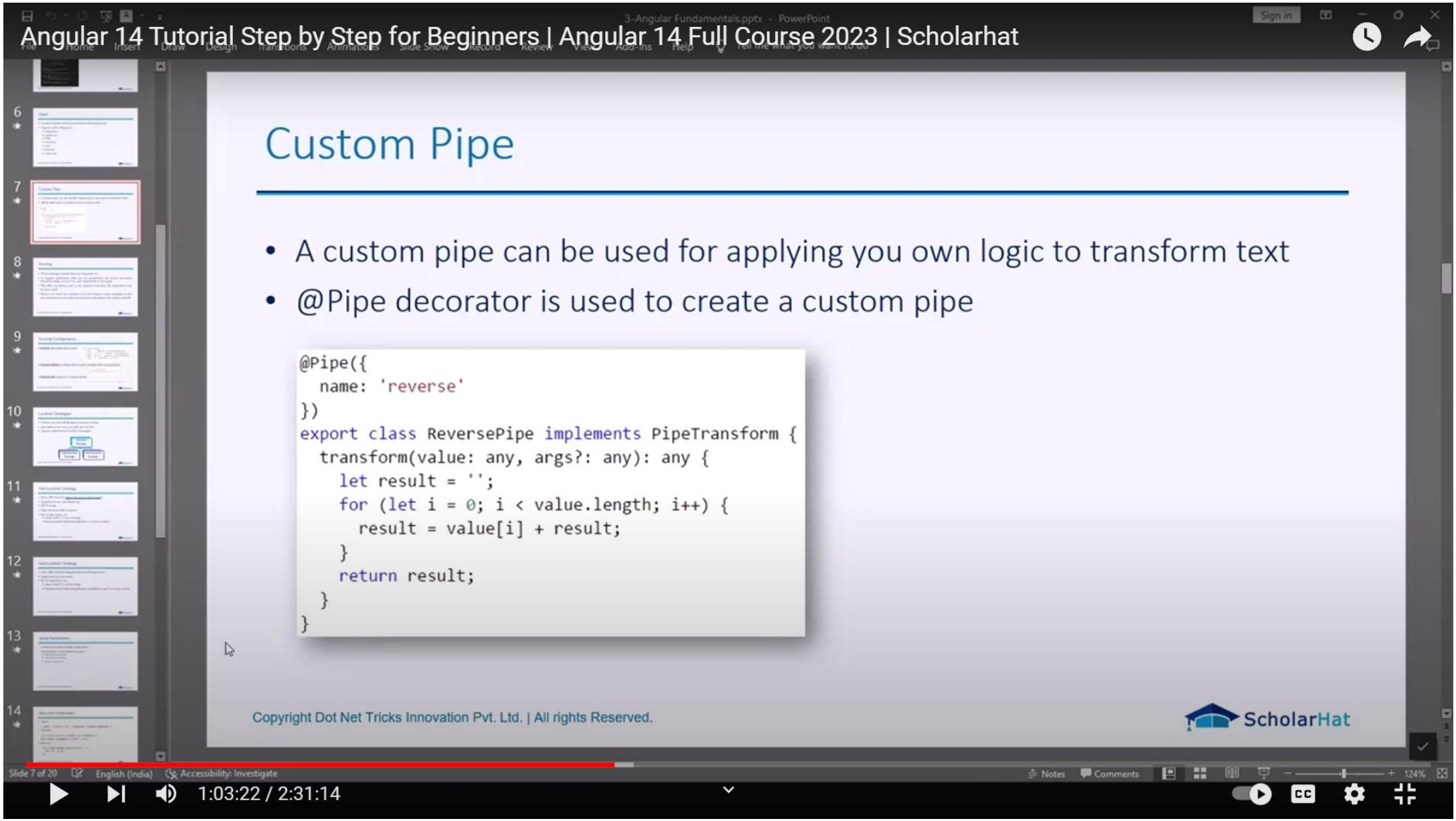
- @Directive decorator is used to create a custom directive
- Custom directive can be used as an attribute or class

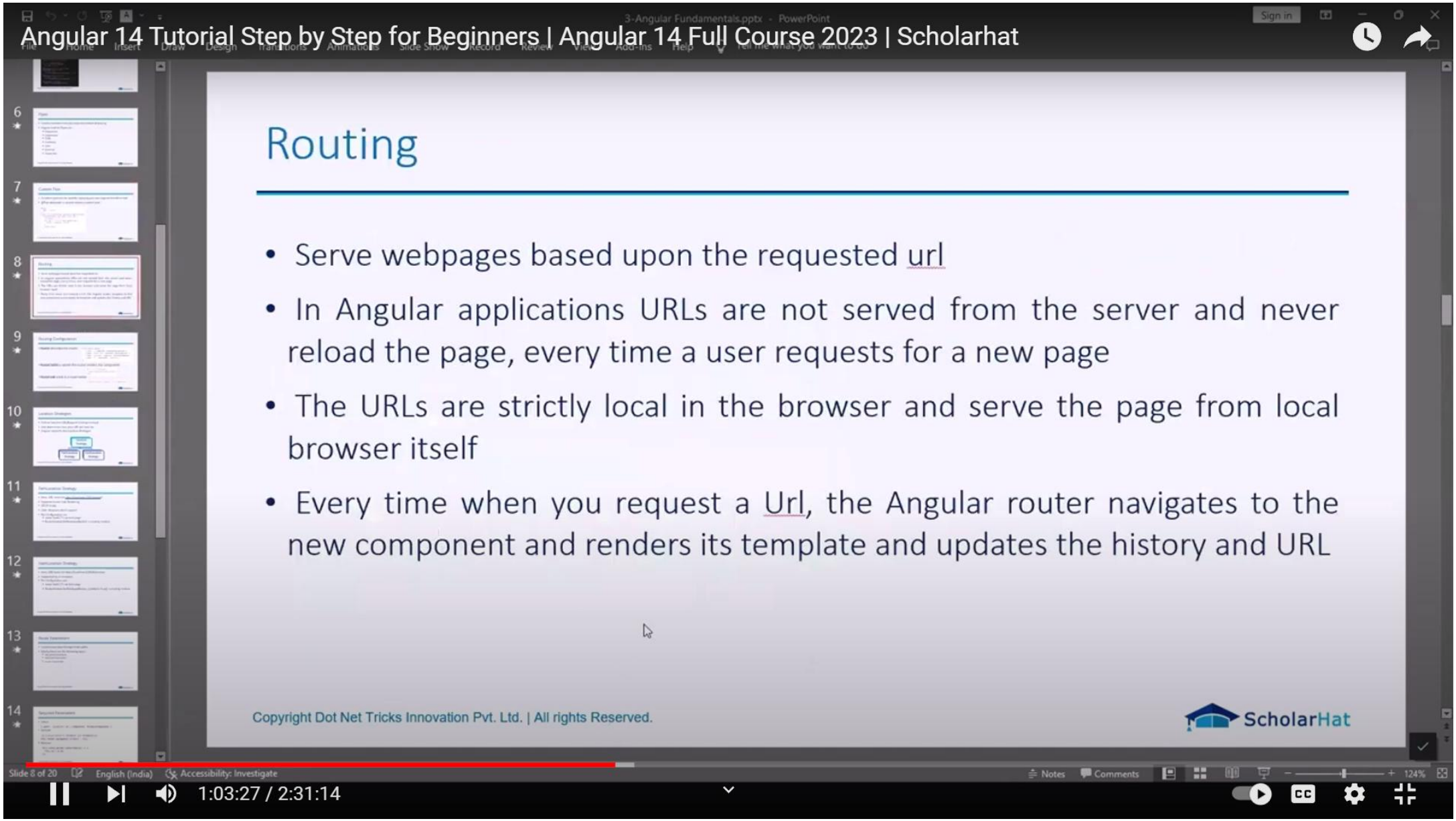
```
@Directive({
  selector: '[appHighlight],.appHighlight'
})
export class HighlightDirective {
  constructor(private el: ElementRef) {
    this.color = 'black';
  }
  @HostBinding('style.color') color: string;
  @HostListener('mouseover') f1() {
    this.color = 'red';
    this.el.nativeElement.style.backgroundColor = 'yellow';
  }
  @HostListener('mouseleave') f2() {
    this.color = 'black';
    this.el.nativeElement.style.backgroundColor = '';
  }
  @HostListener('click') f3() {
    alert(this.el.nativeElement.innerHTML);
  }
}
```

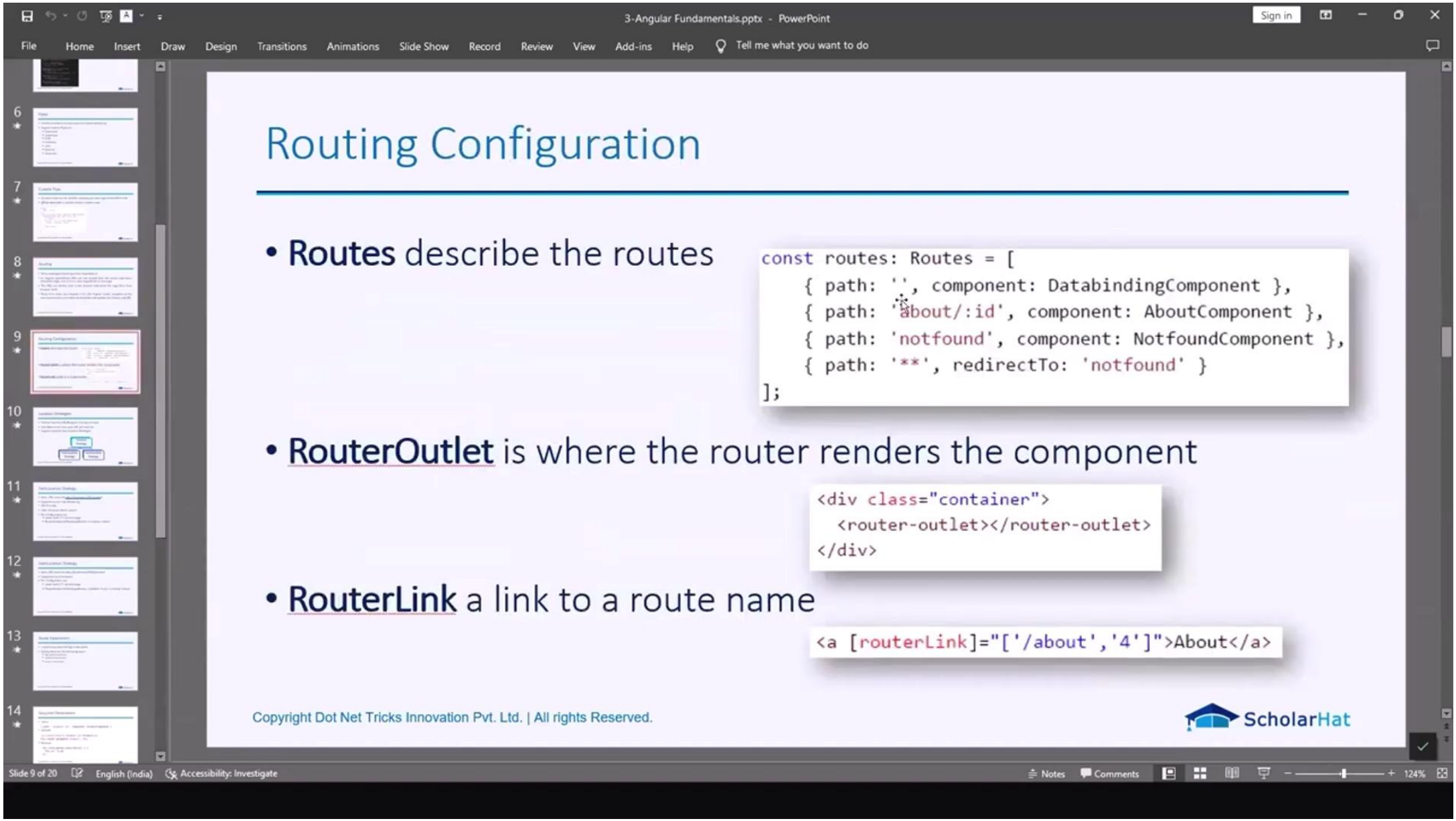


ScholarHat


3-Angular Fundamentals.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do
Pipes
• Used to transform bound properties before displaying
• Angular built-In Pipes are :
▪ Lowercase
▪ Uppercase
▪ Date
▪ Currency
▪ Json
▪ Decimal
▪ Async etc.
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 6 of 20 English (India) Accessibility: Investigate
Notes Comments
124%

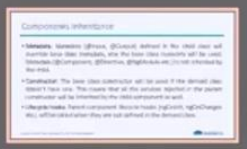
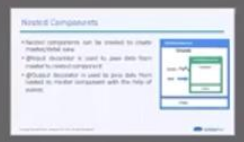

Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
3-Angular Fundamentals.pptx - PowerPoint
Sign in
Custom Pipe
A custom pipe can be used for applying you own logic to transform text
@Pipe decorator is used to create a custom pipe
@Pipe({
name: 'reverse'
})
export class ReversePipe implements PipeTransform {
transform(value: any, args?: any): any {
let result = '';
for (let i = 0; i < value.length; i++) {
result = value[i] + result;
}
return result;
}
}
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 7 of 20
English (India)
Accessibility: Investigate
Notes
Comments
124%
1:03:22 / 2:31:14


Angular 14 Tutorial Step by Step for Beginners | Angular 14 Full Course 2023 | Scholarhat
3-Angular Fundamentals.pptx - PowerPoint
Sign in
Routing
Serve webpages based upon the requested url
In Angular applications URLs are not served from the server and never reload the page, every time a user requests for a new page
The URLs are strictly local in the browser and serve the page from local browser itself
Every time when you request a Url, the Angular router navigates to the new component and renders its template and updates the history and URL
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 8 of 20
English (India)
Accessibility: Investigate
Notes
Comments
124%
1:03:27 / 2:31:14

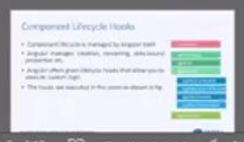

3-Angular Fundamentals.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do
Routing Configuration
• Routes describe the routes
const routes: Routes = [
{ path: '', component: DatabindingComponent },
{ path: 'about/:id', component: AboutComponent },
{ path: 'notfound', component: NotfoundComponent },
{ path: '**', redirectTo: 'notfound' }
];
• RouterOutlet is where the router renders the component

<router-outlet></router-outlet>

• RouterLink a link to a route name
<a [routerLink]="['/about','4']">About
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 9 of 20 English (India) Accessibility: Investigate
Notes Comments
124%

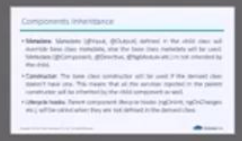
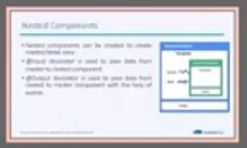




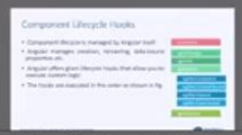
Components Inheritance

- **Metadata**: Metadata (@Input, @Output) defined in the child class will override base class metadata, else the base class metadata will be used. Metadata (@Component, @Directive, @NgModule etc.) is not inherited by the child.
- **Constructor**: The base class constructor will be used if the derived class doesn't have one. This means that all the services injected in the parent constructor will be inherited by the child component as well.
- **Lifecycle hooks**: Parent component lifecycle hooks (ngOnInit, ngOnChanges etc.) will be called when they are not defined in the derived class.

ScholarHat

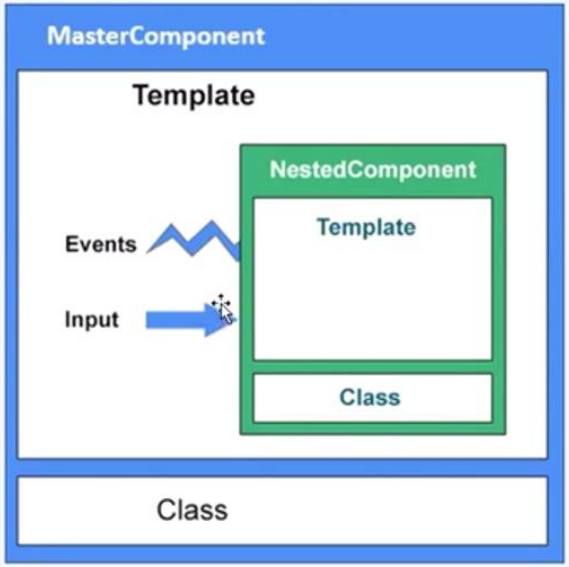






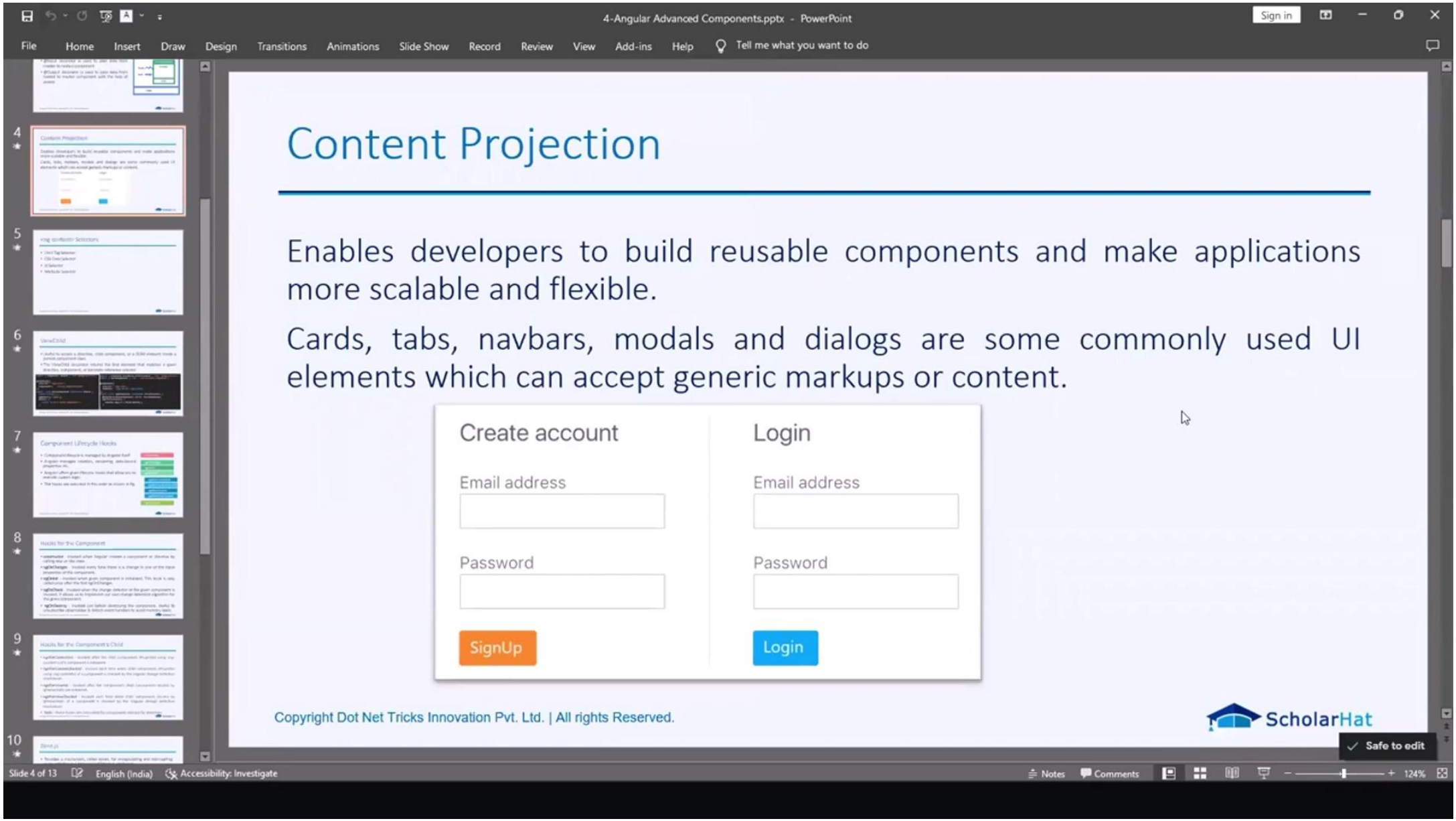


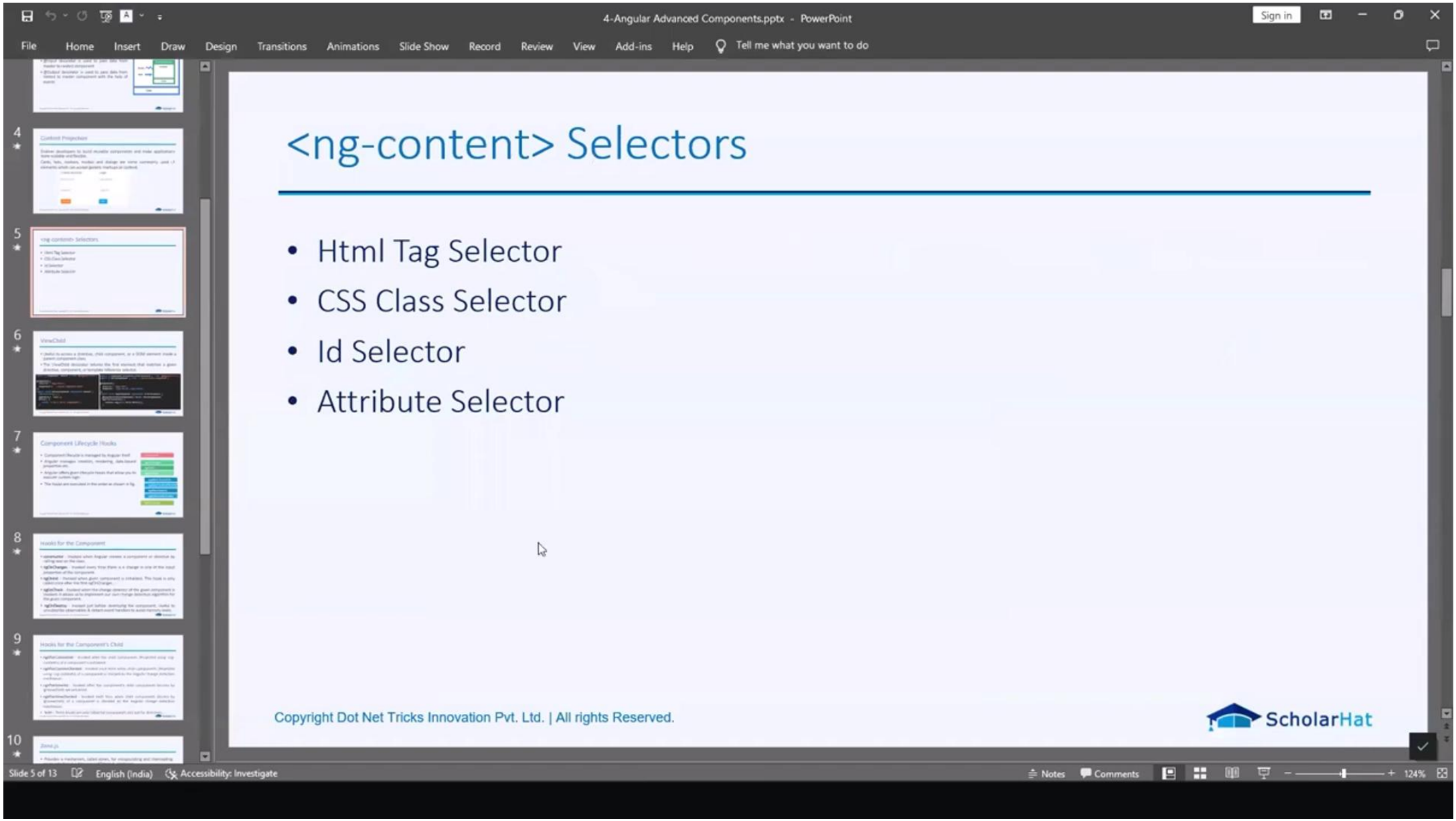
Nested Components

- Nested components can be created to create master/detail view
- @Input decorator is used to pass data from master to nested component
- @Output decorator is used to pass data from nested to master component with the help of events



ScholarHat


4-Angular Advanced Components.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do
Content Projection
Enables developers to build reusable components and make applications more scalable and flexible.
Cards, tabs, navbars, modals and dialogs are some commonly used UI elements which can accept generic markups or content.
Create account
Email address
Password
SignUp
Login
Email address
Password
Login
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Safe to edit
Slide 4 of 13
English (India)
Accessibility: Investigate
Notes
Comments
124%


4-Angular Advanced Components.pptx - PowerPoint
Sign in
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help
Tell me what you want to do
<ng-content> Selectors
• Html Tag Selector
• CSS Class Selector
• Id Selector
• Attribute Selector
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 5 of 13
English (India)
Accessibility: Investigate
Notes
Comments
124%

ViewChild

- Useful to access a directive, child component, or a DOM element inside a parent component class.
- The ViewChild decorator returns the first element that matches a given directive, component, or template reference selector.

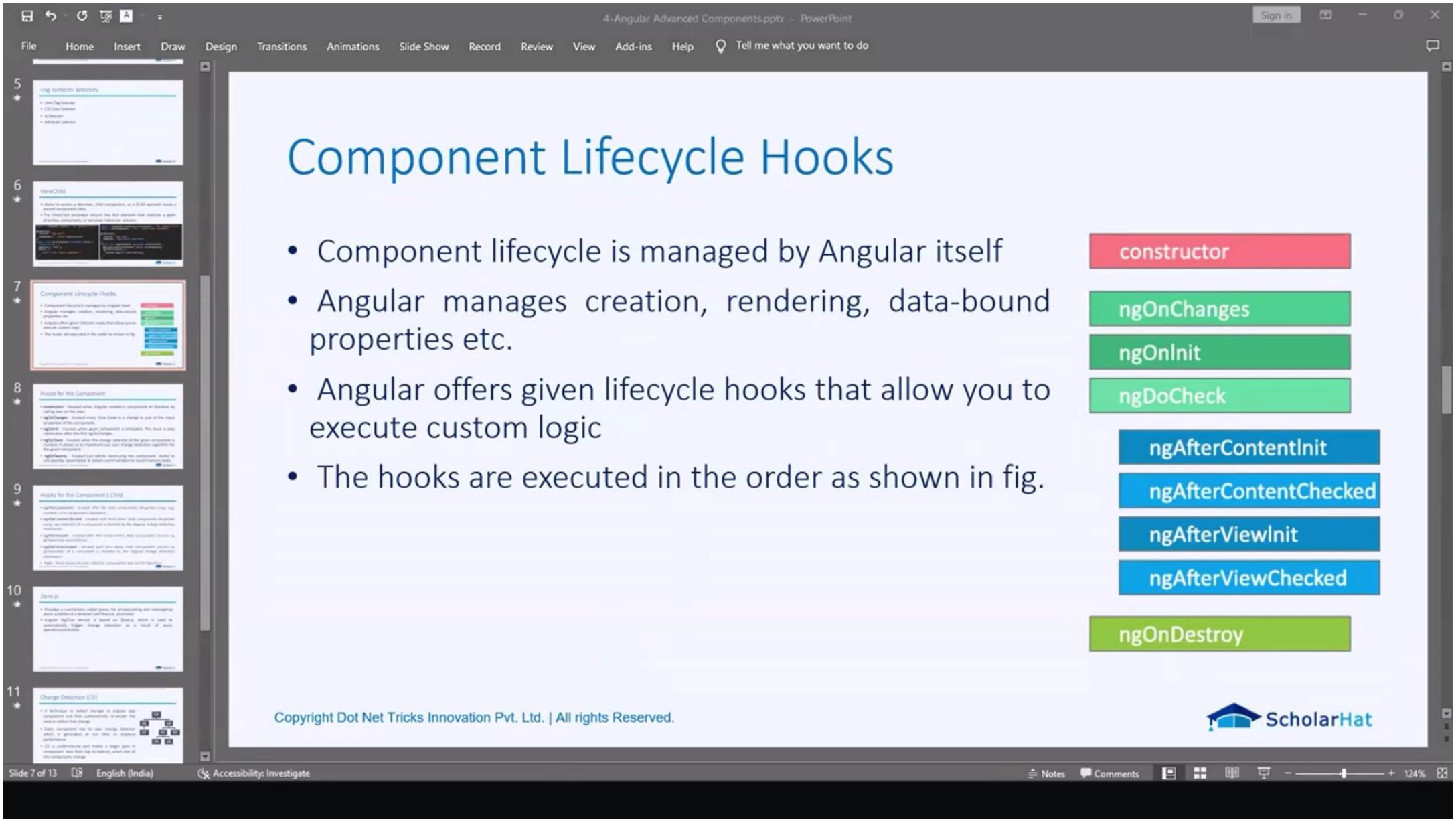
```
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-child',
  templateUrl: './child.component.html'
})
export class ChildComponent implements OnInit {
  constructor() { }
  ngOnInit(): void {}
  Hello() {
    return 'I am a child component!';
  }
}
```

```
import { Component,ViewChild,AfterViewInit } from '@angular/core';
import { ChildComponent } from './child/child.component';

@Component({
  selector: 'app-root',
  template: `<app-child></app-child>`
})
export class AppComponent implements AfterViewInit {
  @ViewChild(ChildComponent) child: ChildComponent;
  ngAfterViewInit() {
    console.log(this.child.Hello());
  }
}
```


ScholarHat


4-Angular Advanced Components.pptx - PowerPoint
File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do
Component Lifecycle Hooks
• Component lifecycle is managed by Angular itself
• Angular manages creation, rendering, data-bound properties etc.
• Angular offers given lifecycle hooks that allow you to execute custom logic
• The hooks are executed in the order as shown in fig.
constructor
ngOnChanges
ngOnInit
ngDoCheck
ngAfterContentInit
ngAfterContentChecked
ngAfterViewInit
ngAfterViewChecked
ngOnDestroy
Copyright Dot Net Tricks Innovation Pvt. Ltd. | All rights Reserved.
ScholarHat
Slide 7 of 13 English (India) Accessibility: Investigate Notes Comments 124%

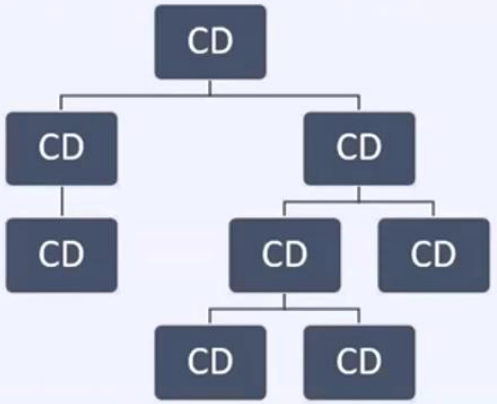
Zone.js

- Provides a mechanism, called zones, for encapsulating and intercepting async activities in a browser (setTimeout, promises)
- Angular *NgZone* service is based on Zone.js, which is used to automatically trigger change detection as a result of async operations/activities

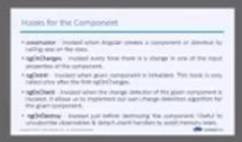
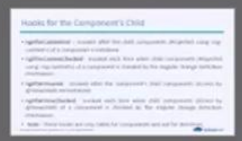
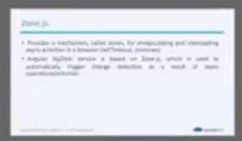
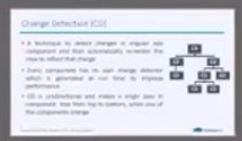
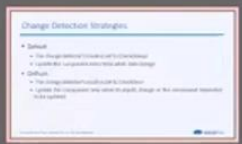
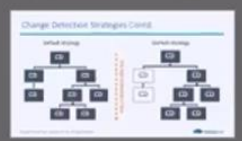
ScholarHat

Change Detection (CD)

- A technique to detect changes in angular app component and then automatically re-render the view to reflect that change
- Every component has its own change detector which is generated at run time to improve performance
- CD is unidirectional and makes a single pass in component tree from top to bottom, when one of the components change



ScholarHat

Change Detection Strategies

- Default
  - The change detector's mode is set to *CheckAlways*
  - Update the component every time when data change
- OnPush
  - The change detector's mode is set to *CheckOnce*
  - Update the component only when its inputs change or the component requested to be updated

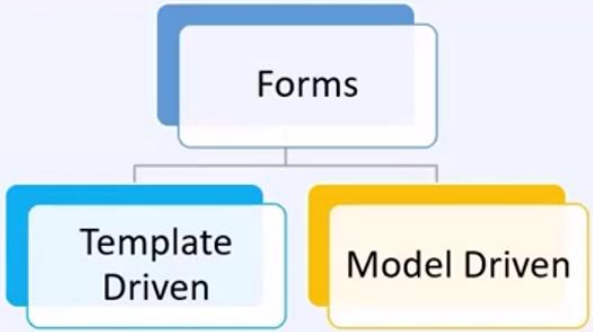
ScholarHat

5-Angular Forms.pptx - PowerPoint

File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

Angular Forms

- HTML forms are an essential part of a web application
- Provides two ways to create form – Template Driven and Model-Driven



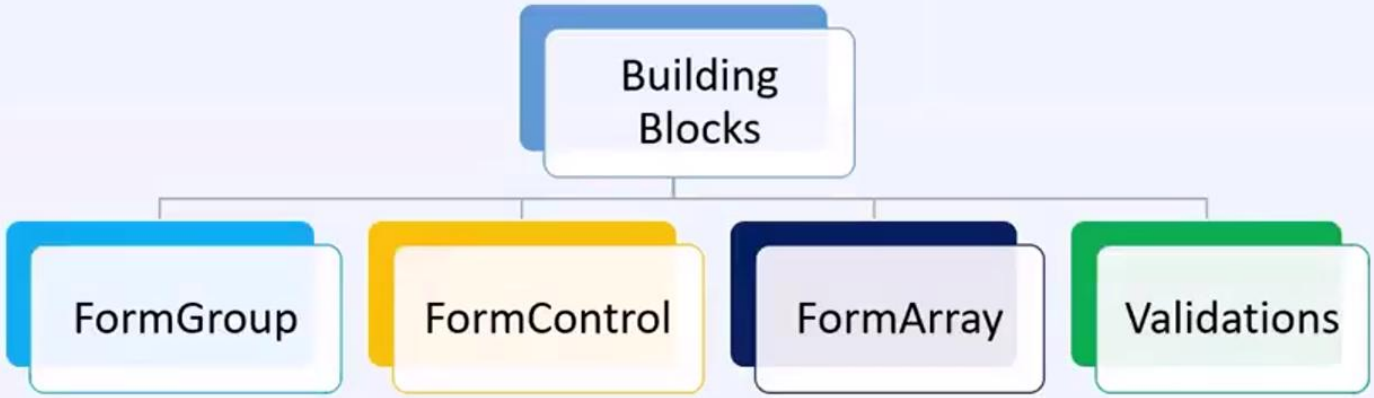
ScholarHat

Slide 2 of 10 English (India) Accessibility: Investigate Notes Comments 124%

5-Angular Forms.pptx - PowerPoint

File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

Angular Form Building Blocks



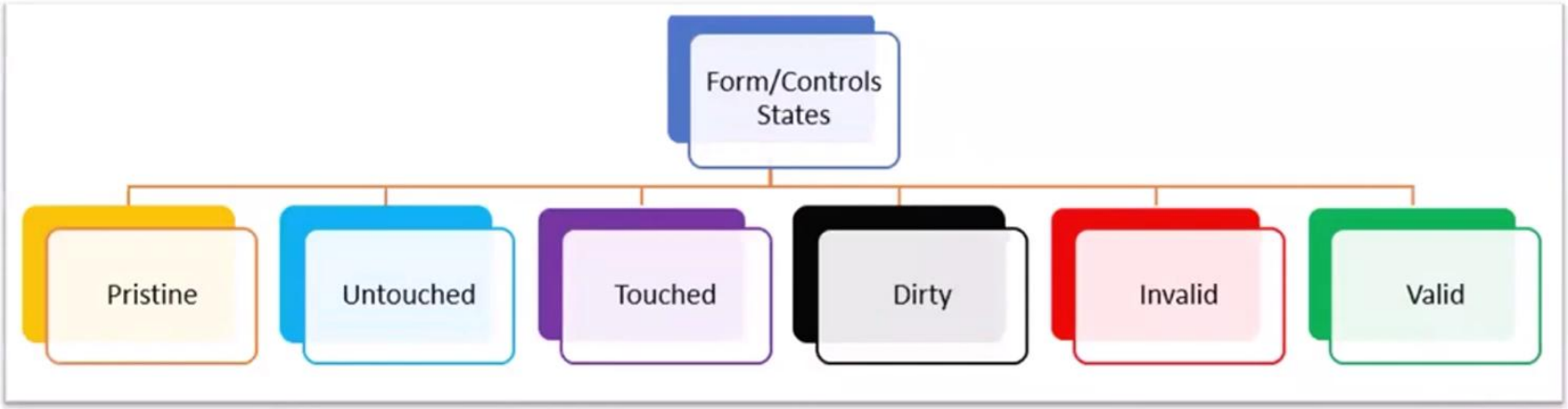
ScholarHat

Slide 3 of 10 English (India) Accessibility: Investigate Notes Comments 124%

5-Angular Forms.pptx - PowerPoint

File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

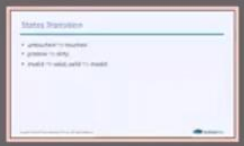
Angular Form and Form Controls States



ScholarHat

Slide 4 of 10 English (India) Accessibility: Investigate Notes Comments 124%




States Transition

- untouched => touched
- pristine => dirty
- invalid => valid, valid => invalid

ScholarHat

Template Driven Form

- Form is setup and configured in HTML Code
- Easy and suits for simple form
- Use Directives and Reference Name
  - ngForm
  - ngModel
  - #refName

Form Group

```
<form #myForm="ngForm" (ngSubmit)="save()">
        <input id="name" type="text"
               [ (ngModel) ]="user.Name"
               name="Name"
               #Name="ngModel"/>
</form>
```

Form Control

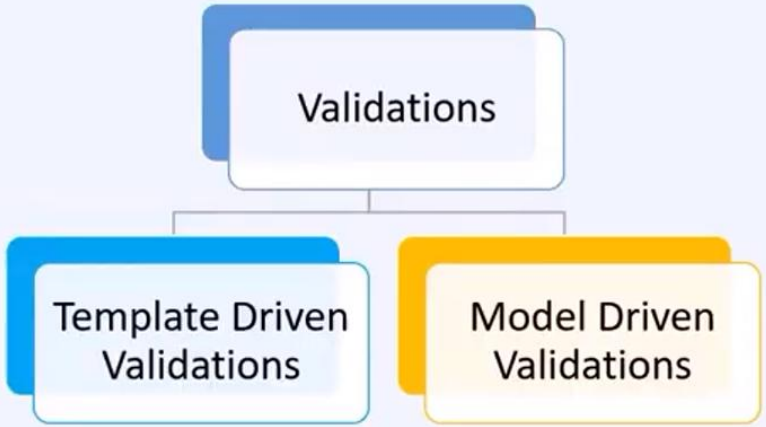
ScholarHat

5-Angular Forms.pptx - PowerPoint

File Home Insert Draw Design Transitions Animations Slide Show Record Review View Add-ins Help Tell me what you want to do

Angular Form Validations

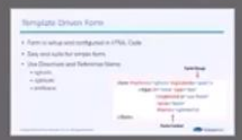
- Validate user input for accuracy and completeness
- Performs on the client-side.

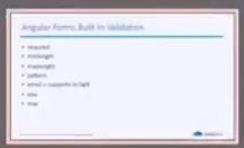
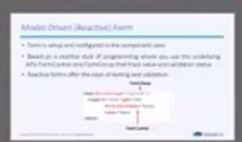
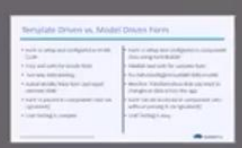


ScholarHat

Slide 7 of 10 English (India) Accessibility: Investigate Notes Comments 124%





Angular Forms Built-In Validation

- required
- minlength
- maxlength
- pattern
- email – supports in Ng4
- min
- max

ScholarHat