

angular task 1

```
<h1>control-statement-example works!</h1>
```

```
<p class="mytext">
```

A structural directive that conditionally includes a template based on the value of an expression coerced to Boolean.

When the expression evaluates to true, Angular renders the template provided in a then clause, and when false or null, Angular renders the template provided in an optional else clause.

The default template for the else clause is blank.

```
</p>
```

```
<button (click)="changevisibility()">Change Visibility</button>
```

```
<h4>if condition</h4>
```

```
<span>{{ifcode}}</span>
```

```
@if(show){
```

```
    <h3>Kalaivani G</h3>
```

```
}
```

```
<h4>if else condition</h4>
```

```
<span>{{ifelsecode}}</span>
```

```
@if(show)
```

```
{
```

```
    <h3>Kalaivani G</h3>
```

```
}
```

```
@else
```

```
{
```

```
    <h3>User name is protected</h3>
```

```
}
```

```
<h4>switch statement</h4>
```

```
<span>{{switchcode}}</span>
```

```
<input type="text" [(ngModel)]="switchvalue">
```

```
{{switchvalue}}
```

```
@switch(switchvalue)
```

```
{
```

```
    @case('if'){
```

```
        <p>
```

```
            <span>IF STATEMENT<br></span>
```

```
            if statement is a selection statement that allows more than one possible flow of control
```

```
        </p>
```

```
    }
```

```
    @case('if-else'){
```

```
        <p>
```

```
<span>IF-ELSE STATEMENT<br></span>
```

The if/else statement executes a block of code if a specified condition is true. If the condition is false, another block of code can be executed.

```

    </p>
  }
  @case('switch'){
    <p>

```

SWITCH STATEMENT

The switch statement is the control statement that allows any value to change the control of the execution

```

    </p>
  }
}

```

controlstatement.ts

```

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

```

```

@Component({
  selector: 'app-control-statement-example',
  standalone: true,
  imports: [FormsModule],
  templateUrl: './control-statement-example.component.html',
  styleUrls: ['./control-statement-example.component.css']
})

```

```

export class ControlStatementExampleComponent {
  show:boolean=true;
  switchvalue:any='if';

```

```

ifcode="@if(show){ <h3> Kalaivani G </h3> }";

ifelsecode="@if(show)<h3>Kalaivani G</h3> }  @else{ <h3>User name is protected</h3> }";

switchcode="@switch(statementtype) { @case('if'){<p>hello</p> }  @case('if-else'){<p>hii</p>}
@case('switch'){ <p>bye</p>}}";

changevisibility()

{

    if(this.show==true)

    {

        this.show=false;

    }

    else{

        this.show=true;

    }

}

}

```

deferrable.html

<h1>deferrable view</h1>

<p>Deferrable views can be used in component template to defer the loading of select dependencies within that template.

Those dependencies include components, directives, and pipes, and any associated CSS. </p>

<p>powerful tool that can be used to reduce the initial bundle size of your application or defer heavy components that may not ever be loaded until a later time. This should result in a faster initial load and an improvement in your Core Web Vitals (CWV) results.

Deferring some of your components until later should specifically improve Largest Contentful Paint (LCP) and Time to First Byte (TTFB).</p>

```

@defer {

  <app-control-statement-example />

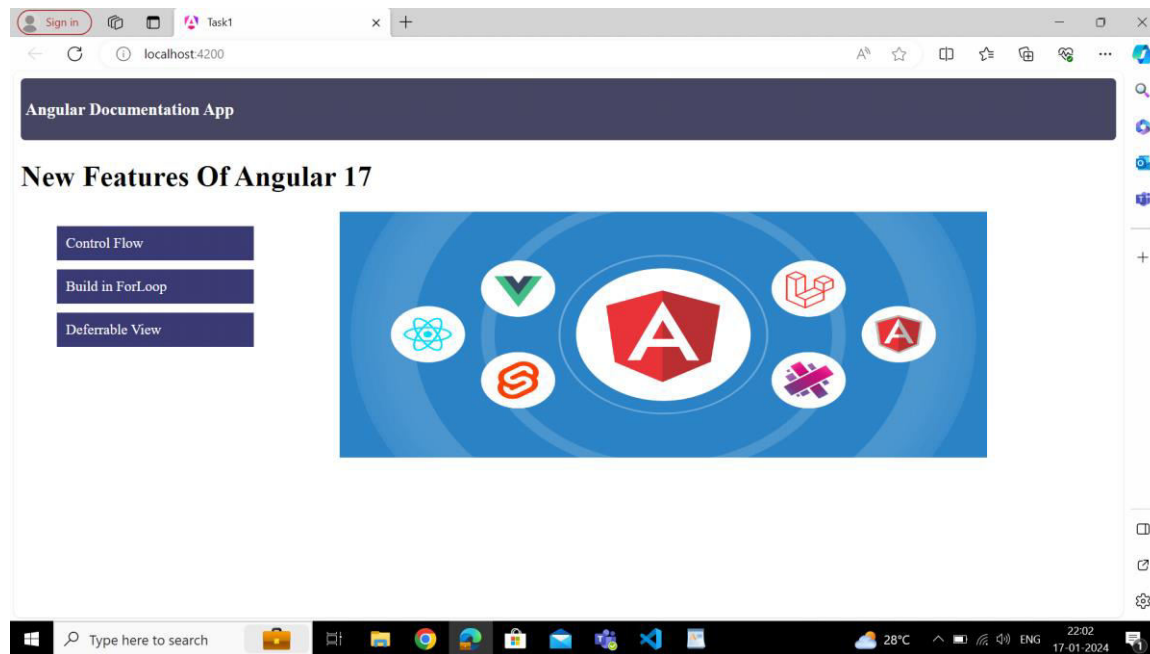
} @placeholder (minimum 5000ms) {

  <span>Control statement block will be displayed in 5000ms</span>

}

```

output screenshots :



Sign in

ControlStatement

localhost:4200/Controlstatement

Control-Statement-Example Works!

A structural directive that conditionally includes a template based on the value of an expression coerced to Boolean. When the expression evaluates to true, Angular renders the template provided in a then clause, and when false or null, Angular renders the template provided in an optional else clause. The default template for the else clause is blank.

Change Visibility

If Condition

```
@if(show){ <h3> Kalaivani G </h3> }
```

If Else Condition

```
@if(show)<h3>Kalaivani G</h3> } @else{ <h3>User name is protected</h3> }
```

User name is protected

Switch Statement

```
@switch(statementtype) { @case('if'){<p>hello</p> } @case('if-else'){<p>hii</p> } @case('switch'){ <p>bye</p> }} if
```

IF STATEMENT

if statement is a selection statement that allows more than one possible flow of control

28°C

22:03

17-01-2024

Sign in

ControlStatement

localhost:4200/Controlstatement

Control-Statement-Example Works!

A structural directive that conditionally includes a template based on the value of an expression coerced to Boolean. When the expression evaluates to true, Angular renders the template provided in a then clause, and when false or null, Angular renders the template provided in an optional else clause. The default template for the else clause is blank.

Change Visibility

If Condition

```
@if(show){ <h3> Kalaivani G </h3> }
```

Kalaivani G

If Else Condition

```
@if(show)<h3>Kalaivani G</h3> } @else{ <h3>User name is protected</h3> }
```

Kalaivani G

Switch Statement

```
@switch(statementtype) { @case('if'){<p>hello</p> } @case('if-else'){<p>hii</p> } @case('switch'){ <p>bye</p> }} if
```

IF STATEMENT

if statement is a selection statement that allows more than one possible flow of control

28°C

22:03

17-01-2024

Sign in

ControlStatement

+

localhost:4200/Controlstatement

Control-Statement-Example Works!

A structural directive that conditionally includes a template based on the value of an expression coerced to Boolean. When the expression evaluates to true, Angular renders the template provided in a then clause, and when false or null, Angular renders the template provided in an optional else clause. The default template for the else clause is blank.

Change Visibility

If Condition

@if(show){ <h3> Kalaivani G </h3> }

Kalaivani G

If Else Condition

@if(show)<h3>Kalaivani G</h3> } @else{ <h3>User name is protected</h3> }

Kalaivani G

Switch Statement

@switch(statementtype) { @case('if'){<p>hello</p> } @case('if-else'){<p>hii</p> } @case('switch'){ <p>bye</p> }} if-else

IF-ELSE STATEMENT

The if/else statement executes a block of code if a specified condition is true. If the condition is false, another block of code can be executed.

28°C 22:03 17-01-2024

Sign in

Deferrable

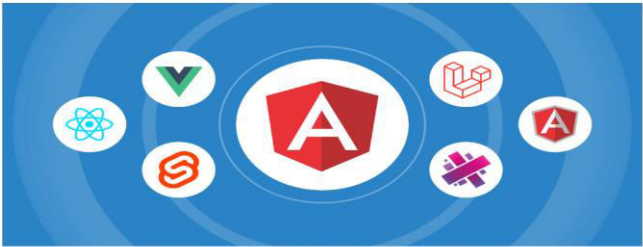
+

localhost:4200/Deferrables

Control Flow

Build in ForLoop

Deferrable View



Deferrable View

Deferrable views can be used in component template to defer the loading of select dependencies within that template. Those dependencies include components, directives, and pipes, and any associated CSS.

powerful tool that can be used to reduce the initial bundle size of your application or defer heavy components that may not ever be loaded until a later time. This should result in a faster initial load and an improvement in your Core Web Vitals (CWV) results. Deferring some of your components until later should specifically improve Largest Contentful Paint (LCP) and Time to First Byte (TTFB).

Control statement block will be displayed in 5000ms

28°C 22:03 17-01-2024