Directives -Quiz

Consider the following code snippets created for an Angular application.

**Employee.ts:**

export class Employee {

constructor() { }

}

Assume you have defined two variables of class Employee type in our component as given below.

emp1 = new Employee();

emp2 : Employee;

What will happen when the below code is executed?

<div \*ngIf="emp1">

Inside first div element

</div>

<div \*ngIf="emp2">

Inside second div element

</div>

Assumption(s): All the components are registered in the app component of the application.



Only <strong>Inside first div element</strong> is displayed



Only <strong>Inside second div element</strong> is displayed



Both <strong>Inside first div element<br />Inside second div element</strong> are displayed



The code results in an error because \*ngIf directive can be used only once in an Angular 6 component.

**Option Only <strong>Inside first div element</strong> is displayed is Correct**

Q2 of 5outlined\_flag

Consider the following code snippet.

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

@Component({

selector: 'my-app',

templateUrl: 'compute.component.html'

})

export class ComputeComponent {

items = [5,10,15,20];

}

Identify the appropriate code snippets that should be used in 'compute.component.html' so that the output will be as follows:

Odd Number

Even Number

Odd Number

Even Number



<div \*ngFor="let item of items"> <div \*ngIf="item % 2 == 0"> <div>Even Number</div> <div>Odd Number</div> </div></div>



<div \*ngFor="let item of items"> <div \*ngIf="item % 2 == 0"> <div>Even Number</div> </div> <div \*ngIfElse="item % 2 == 1"> <div>Odd Number</div> </div> </div>



<div \*ngFor="let item of items"> <div \*ngIf="item % 2 == 0"> <div>Even Number</div> </div> <div>Odd Number</div> </div>



<div \*ngFor="let item of items"> <div \*ngIf="item % 2 == 0"> <div>Even Number</div> </div> <div \*ngIf="item % 2 == 1"> <div>Odd Number</div> </div> </div>

**Option <div \*ngFor="let item of items"> <div \*ngIf="item % 2 == 0"> <div>Even Number</div> </div> <div \*ngIf="item % 2 == 1"> <div>Odd Number</div> </div> </div> is Correct**

Q3 of 5outlined\_flag

The following error occurred while executing the following code snippet. Identify the appropriate options to fix the errors

Error: **Can't bind to 'ngFor' since it isn't a known property of 'tr'**

<tr \*ngFor="city in cities">

<td>city</td>

</tr>

**Note:** You are expected to display the values instead of index.



Replace the expression as follows: <tr \*ngFor="city of cities"> <td>city</td></tr>



Replace the expression as follows: <tr \*ngFor="let city in cities"> <td>city</td></tr>



Replace the expression as follows: <tr \*ngFor="var city in cities"> <td>city</td></tr>



Replace the expression as follows: <tr \*ngFor="let city of cities"> <td>city</td></tr>

**Option Replace the expression as follows: <tr \*ngFor="let city of cities"> <td>city</td></tr> is Correct**

Q4 of 5outlined\_flag

Which among the following is/are a valid syntax for ngif expression?

1. <div \*ng-If="your-condition">...</div>
2. <div \*ngIf="your-condition">...</div>
3. <div template="ngIf your-condition">...</div>
4. <ng-template [ngIf]="your-condition">  
       <div>...</div>  
   </ng-template>



Only 2 and 3 are valid



Only 2,3 and 4 are valid



Only 2 is valid



Only 1 is valid

**Option Only 2,3 and 4 are valid is Correct**

Q5 of 5outlined\_flag

Potter has written the below code where status has value as false. He and his friend are arguing on the output. Consider the below code and decide who is write among them.

<div [ngIf]="status">

<h2>Hello John </h2>

</div>

<div [hidden] = "!status">

<div>

<h2> Hello Alice</h2>

</div>

</div>

A. John Says output could be “Error [ngIf] should be \*ngIf”

B. Alice says output could be “Hello Alice”



John is Correct



Alice is Correct



John and Alice both are Correct



Both are wrong

**Option John is Correct is Correct**