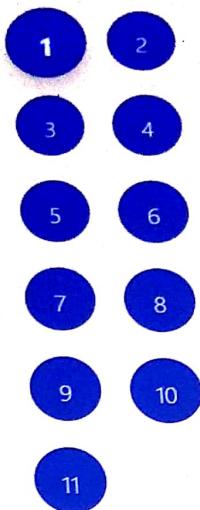


## Angular Basics



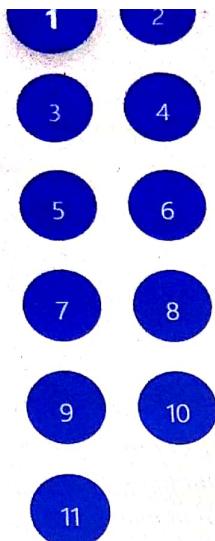
Consider the below code:

```
app.component.ts:  
export class AppComponent {  
    empList: any[] = [ {  
        "empId": "1",  
        "empName" : "Roger"  
    },  
    {  
        "empId": "2",  
        "empName" : "Jimmy"  
    } ];  
}
```

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

`<div *ngFor = 'let emp of empList'>`  
`<ul>`  
`<li>{{emp.empId}}</li>`  
`<li>{{emp.empName}}</li>`  
`</ul>`  
`</div>`

`<div *ngFor = 'let emp as empList'>`  
`<ul>`  
`<li>{{emp.empId}}</li>`  
`<li>{{emp.empName}}</li>`  
`</ul>`  
`</div>`



```
<div ngFor = 'let emp of empList'>
  <ul>
    <li>{{emp.empId}}</li>
    <li>{{emp.empName}}</li>
  </ul>
</div>
```

Reset

Save

## Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

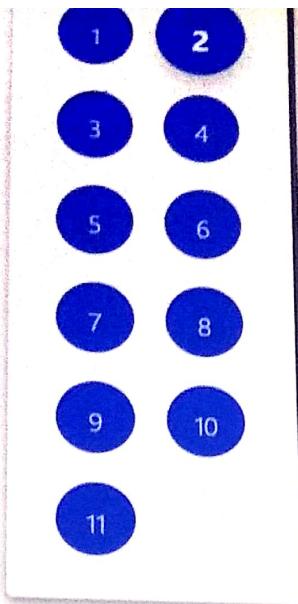
## Question 2

If `app.component.ts` file contains the following array

```
empArr=[  
    {empId:100,empName:'Emp1',age:30},  
    {empId:200,empName:'Emp2',age:32},  
    {empId:300,empName:'Emp3',age:34},  
    {empId:400,empName:'Emp4',age:37},  
];
```

Output:

100	30
200	32
300	34
400	Emp4



i;

### Output:

100  
200  
300  
400

30  
32  
34  
Emp4

Which of the below code snippet written in app.component.html will give output?

a]

```
<table border="1px solid">  
<tr *ngFor="let emp of empArr">
```

b]

```
<table border="1px solid">  
<tr *ngIf="emp.age>35" *ngFor="let emp of empArr">
```

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

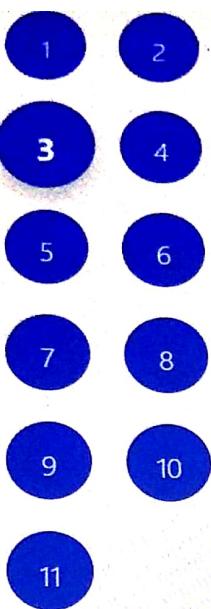
</table>

</table>

- Only [b]
- Both [a] and [b]
- neither [a] nor [b]
- Only [a]

Reset

Save



```
<!-- app.component.html -->  
{'the united kingdom' | slice:5:13 | titlecase } }
```



- Nited Ki
- Nited Kingdom
- nited kingdom
- nited ki

Reset

Save

#### Question 4

Anna, a web developer, created the below form in her application. What will be the output displayed at the browser console when the below form is rendered initially? Assume that all necessary imports have been done.

31%

69%

- 1
- 2
- 3
- 4**
- 5
- 6
- 7
- 8
- 9
- 10
- 11

server.component.ts:

```
export class ServerComponent {  
  
    getValue(event){  
        console.log(event.target.value); //line1  
    }  
}
```

server.component.html:

```
<form>  
    <input type="text" value="user1" (change)="getValue($event)">  
</form>
```

You're be

- 1
- 2
- 3
- 4**
- 5
- 6
- 7
- 8
- 9
- 10
- 11

```
<form>
  <input type="text" value="user1" (change)="getValue($event)">
</form>
```

→

- user1
- Returns a runtime error as "value is not a property of unknown"
- {value : user1}
- No value will be displayed in the console

## Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

Tom, a front-end developer wants to update his Angular 7 project to the latest Angular version. Suggest him the possible way.

71%

- ng update @angular/cli @angular/core
- npm update
- ng angular --update
- npm update ng

Reset

Save

## Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

Option B: <nav [ngClass]=[“navbar, navbar-inverse”]>...</nav>

Option C: <nav [ngClass]={‘navbar’:true, ‘navbar-inverse’:true}>...</nav>

Option D: <nav [ngClass]={“navbar:true, navbar-inverse:true”}>...</nav>

- Only option B
- Only options B and D
- Only options A and C
- Only option A

Reset

Save

### Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

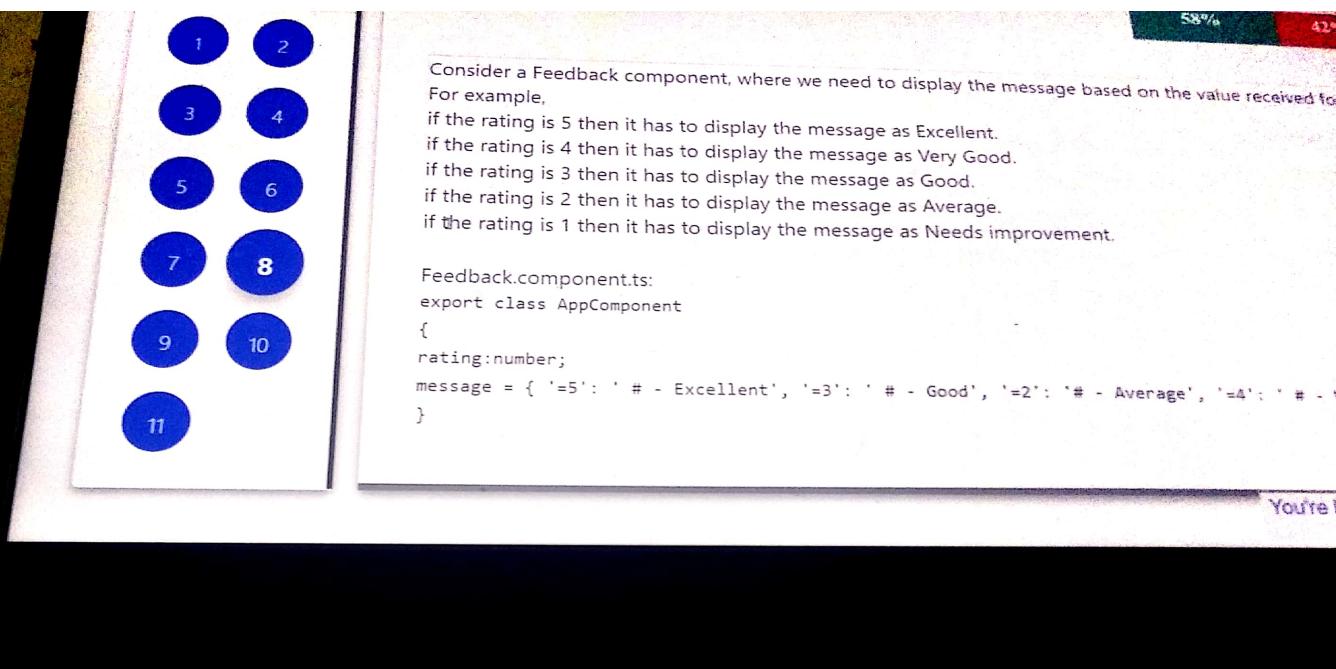
Which of the following are false w.r.t to change detection algorithm

- Zones detects all asynchronous actions at run time
- zones informs Angular about the changes in the application
- "The change detection algorithm runs at the component where change is detected and the root component"
- The change detection algorithm is generated when every component is initialized

Reset

Save

You're



Consider a Feedback component, where we need to display the message based on the value received for example,  
if the rating is 5 then it has to display the message as Excellent.  
if the rating is 4 then it has to display the message as Very Good.  
if the rating is 3 then it has to display the message as Good.  
if the rating is 2 then it has to display the message as Average.  
if the rating is 1 then it has to display the message as Needs improvement.

Feedback.component.ts:

```
export class AppComponent
{
  rating:number;
  message = { '=5': '# - Excellent', '=3': '# - Good', '=2': '# - Average', '=4': '# - V' }
}
```

You're t

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

What should be the template written to achieve this requirement?

- `<p> {{rating | message}} </p>`
- `<p> {{rating | i18nPlural:message }} </p>`
- `<p> {{message | i18nPlural:rating }} </p>`
- `<p> {{rating | i18nPlural }} </p>`

Reset

Save

## Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

### Question 9

What will be the output displayed on the browser while serving the application with the following code? Assume that all necessary imports are made.

64%

home.component.ts:

```
-----  
@Component({  
    selector: 'app-home',  
    template: '',  
    styleUrls: ['./home.component.css']  
})
```

app.component.html:

```
-----  
<h1>App Home</h1>  
<app-home></app-home>
```

You're

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

```
app.component.html:  
-----  


# App Home</h1> <app-home></app-home> app.module.ts: ----- @NgModule({ declarations: [ AppComponent, HomeComponent ], imports: [ BrowserModule ], bootstrap: [AppComponent] })


```

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

App Home

- App Home
- No output will be displayed
- Returns a run time error

App Home <app-home></app-home>

-

Consider the below code in ContactDetailsComponent component:  
(Assume all the necessary imports are done)

```
/* contact-details.component.ts */
@Component({
  selector: 'contact-details',
  templateUrl: './contact-details.component.html'
})
export class ContactDetailsComponent {
```

Which of the following is the correct way to render ContactDetailsComponent in other component?

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

## Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

Which of the following is the correct way to render ContactDetailsComponent in other components ?

- <ContactDetailsComponent></ContactDetailsComponent>
- <contact-details></contact-details>
- <div contact-details></div>  
    ↳
- <app-contact-details></app-contact-details>

Reset

Save

You're b

## Angular Basics

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11

### Question 11

Can we have an interface with optional and default properties in Typescript?



- Yes, both optional and default values can be passed
- No, only default values can be passed
- No, only optional values can be passed
- All properties of interface should only contain mandatory parameters

Reset

Save

You're b

Angular  
Advanced

- 1
- 2
- 3
- 4

Question 1

75%

Julie wants to disable the button of the form when the form is invalid. Which of the following options can she use assuming the formGroup value is registerForm

- <button [disabled]="!registerForm.valid">Submit</button>
- <button [disabled]="registerForm.valid">Submit</button>
- <button [disabled]="registerForm.invalid">Submit</button>
- <button [disabled]="!registerForm.invalid">Submit</button>

Angular  
Advanced

- 1
- 2
- 3
- 4

2%

## Question 2

John, a web developer, has created forms with validations in Angular. What will be the result if he enters value 'Max' in the username field? Assume that all necessary imports have been made.

app.component.html:

```
<form [formGroup]="signupform">
  <label for="user">Username: </label> <br/>
  <input type="text" id="user" formControlName="username">
  <label for="pass">Password: </label> <br/>
  <input type="password" id="pass" formControlName="password">
  <button type="submit">Sign up</button>
</form>
```

app.component.ts:

```
export class AppComponent implements OnInit {
```

ADVANCED

- 1
- 2
- 3
- 4

```
app.component.ts:
```

```
export class AppComponent implements OnInit {  
    signupform: FormGroup;  
    hiddennames=['Max','Ray']  
  
    constructor(private formBuilder:FormBuilder) { }  
  
    ngOnInit(){  
        this.signupform = this.formBuilder.group({  
            username: [null, this.allowednames],  
            password: [null],  
        })  
    }  
}
```

## Angular Advanced

- 1
- 2
- 3
- 4

```
allowednames(name){  
    if(this.hiddennames.indexOf(name.value)!=-1){  
        return {name:true}  
    }  
    return null;  
}  
}
```

- Validator function allowednames will return null
- Validator function allowednames will return {Max:true}
- Returns a compilation error
- Returns a run-time error "cannot read property hiddennames of undefined"

**Advanced**

- 1
- 2
- 3
- 4

How many instances of service class will be created if you inject a service in three different components using its component's decorator?

- 5
- 1
- 4
- 3

Reset

Save

39%

## Angular Advanced

- 1
- 2
- 3
- 4

Observe the below code. What should be written at line 10 to fetch the values of name and age from the url 'http://localhost:4200/user/david/27'. Assume that all the imports are done.

app.routing.module:

```
const routes: Routes = [  
  { path: 'user/:name/:age', component: HomeComponent }  
];
```

home.component.ts:

```
export class HomeComponent implements OnInit {  
  user:{name:string,age:number}  
  
  constructor(private route: ActivatedRoute) {}  
  
  ngOnInit() {  
    // line 10  
  }  
}
```

### Advanced

- 1
- 2
- 3
- 4

```
        this.user={  
            name: this.route.snapshot.params['name'],  
            age: this.route.snapshot.params['age']  
        }.
```

```
        this.user={  
            name: this.route.snapshot.param['name'],  
            age: this.route.snapshot.param['age']  
        }.
```

- 1
- 2
- 3
- 4

```
this.user={  
    name: this.route.params['name'],  
    age: this.route.params['age']}
```

O }



O Returns a run time error

Reset

Save