

Accueil > Cours > Débutez avec Angular > Quiz : Améliorer la structure de votre application avec les services et le routing

## Débutez avec Angular

 10 heures  Moyenne

Mis à jour le 13/06/2022



## Améliorer la structure de votre application avec les services et le routing

Vous n'avez pas validé ce quiz.




Vous n'avez pas atteint le seuil de validation de cet exercice, c'est-à-dire 70%. Ce n'est pas très grave car vous pourrez refaire le quiz dans 24h.

### Compétences évaluées

 Améliorer la structure de votre application avec les services et le routing

#### Question 1

Quel décorateur permet de transformer facilement une classe TypeScript en service ?

-  
- ☐
-  ☐
- ☐

Le décorateur  transforme une classe en service.

## Question 2

Quelle option ci-dessous enregistre FolderService à la racine de mon application ?



typescript

```
1 @Injectable('root')
2 export class FolderService {}
```



typescript

```
1 @Injectable({
2   providedIn: 'root'
3 })
4 export class FolderService {}
```



typescript

```
1 @Injectable({
2   injectIn: 'root'
3 })
4 export class FolderService {}
```



typescript

```
1 @Injectable()
2 export class FolderService<root> {}
```

En passant un objet de configuration `{ providedIn: 'root' }` au décorateur `@Injectable()`, on s'assure d'enregistrer le service à la racine de l'application.

## Question 3

Je veux injecter un service MarbleService dans GamesComponent. Quelle est la bonne syntaxe ?



typescript

```
1 @Component({
2   selector: 'app-games',
3   templateUrl: './games.component.html',
4   styleUrls: ['./games.component.scss']
5 })
6 export class GamesComponent implements OnInit {
7   constructor() {
```

```
8     this.marbleService = MarbleService;
9   }
10
11   ngOnInit() {
12
13   }
14 }
```



typescript

```
1 @Component({
2   selector: 'app-games',
3   templateUrl: './games.component.html',
4   styleUrls: ['./games.component.scss']
5 })
6 export class GamesComponent implements OnInit {
7   constructor(private marbleService = MarbleService) {
8
9   }
10
11   ngOnInit() {
12
13   }
14 }
```



typescript

```
1 @Component({
2   selector: 'app-games',
3   templateUrl: './games.component.html',
4   styleUrls: ['./games.component.scss']
5 })
6 export class GamesComponent implements OnInit {
7   constructor() {
8
9   }
10
11   ngOnInit() {
12     private marbleService = MarbleService;
13   }
14 }
```



typescript

```
1 @Component({
2   selector: 'app-games',
3   templateUrl: './games.component.html',
4   styleUrls: ['./games.component.scss']
5 })
6 export class GamesComponent implements OnInit {
7   constructor(private marbleService: MarbleService) {
8
9   }
10
11   ngOnInit() {
```

```
12  
13     }  
14 }
```

Pour injecter un service dans un component, on passe un argument du type du service au constructor avec un modificateur d'accès ; ex. :

typescript

```
1 constructor(private myService: MyService) {}
```

## Question 4

Mon service `LightsService` contient une méthode `getAllLights()` qui retourne un tableau de type `Light[]` .

Dans `RedLightsComponent`, je veux initialiser ma propriété `lights` à partir de la méthode du service.

Comment je fais ?



typescript

```
1 //...  
2 lights: Light[];  
3  
4 constructor(private LightsService: LightsService) {  
5  
6 }  
7  
8 ngOnInit() {  
9     this.lights = this.lightsService.getAllLights();  
10 }  
11 //...
```



typescript

```
1 //...  
2 lights: Light[];  
3  
4 constructor(private LightsService: LightsService) {  
5     this.lights = this.lightsService.getAllLights();  
6 }  
7  
8 ngOnInit() {  
9  
10 }  
11 //...
```



typescript

```

1 //...
2
3 constructor(lights: Light[],
4             private lightsService: LightsService) {
5 }
6
7 ngOnInit() {
8     this.lights = this.lightsService.getAllLights();
9 }
10 //...

```



typescript

```

1 //...
2 lights: Light[];
3
4 constructor() {
5
6 }
7
8 ngOnInit(private lightsService: LightsService) {
9     this.lights = this.lightsService.getAllLights();
10 }
11 //...

```

Dans le constructor, on injecte le service.

Dans `ngOnInit()` , on initialise la propriété en appelant la méthode.

## Question 5

Voici la déclaration de la méthode `addCowbell()` de `CowbellService` :

typescript

```

1 addCowbell(howMuch: 'some' | 'more', when: number): Cowbell[] {
2     //...
3 }

```

Quelle expression ci-dessous est valable ?



typescript

```
1 const cowbell: Cowbell = this.cowbellService.addCowbell('more', 5);
```



typescript

```
1 const cowbells: Cowbell[] = this.cowbellService.addCowbell('some', '5');
```



typescript

```
1 const cowbells: Cowbell[] = this.cowbellService.addCowbell('more', 5);
```

☐

typescript

```
1 const cowbells: Cowbell[] = this.cowbellService.addCowbell('all', 5);
```

Faites bien attention aux types des arguments et du retour !

## Question 6

Quelle déclaration de route associe correctement le component `CoffeeComponent` à la route 'coffee' ?

☒

typescript

```
1 { path: coffee, component: CoffeeComponent }
```

☒

typescript

```
1 { path: 'coffee', component: CoffeeComponent }
```

☐

typescript

```
1 { path: coffee, component: 'CoffeeComponent' }
```

☐

typescript

```
1 { route: 'coffee', component: CoffeeComponent }
```

La déclaration de route spécifiée a besoin d'un `path` de type `string` et d'un `component` qui est le type du component correspondant.

## Question 7

Quel lien se verra attribuer la classe 'active-link' lorsque la route qui y correspond sera la route active ?

☒

html

```
1 <a routerLink="/stuff" routerLinkActive="active-link">Stuff</a>
```

☐

html

```
1 <a routerLinkActive="active-link">Places</a>
```

html



```
1 <a routerLink="models" routerLinkActive="active">Models</a>
```



html

```
1 <a [routerLink]="/stuff" [routerLinkActive]="active-link">Stuff</a>
```

La directive `routerLink` expose l'attribut `routerLinkActive` qui prend comme argument le nom de la classe à appliquer sous forme de `string`. Il n'y a donc pas besoin des crochets de la liaison par attribut.

## Question 8

Quelle balise permet de spécifier l'endroit dans le DOM où le composant de la route active sera inséré ?

`<router-link>``<router-template>``<component-outlet>``<router-outlet>`

La balise `<router-outlet>` spécifie l'emplacement du composant à afficher.

## Question 9

Une fois le Router injecté dans GadgetComponent, comment je l'utilise pour naviguer vers la route `new-gadgets` ?



typescript

```
1 this.router.navigateByUrl('new-gadgets');
```



typescript

```
1 this.navigateByUrl('new-gadgets');
```



typescript

```
1 this.router.navigateByUrl('new-gadgets');
```



typescript

```
1 this.navigateByUrl('new-gadgets');
```

La bonne méthode est `navigateByUrl()` .

## Question 10

La route où se trouve mon composant contient un paramètre dynamique `oldAmount` . Je veux récupérer ce paramètre et l'utiliser pour appeler cette méthode de service :

typescript

```
1 calculateNewAmount(oldAmount: number): number {  
2     //...  
3 }
```

Quelle est la bonne solution ?



typescript

```
1 //...  
2 export class AccountantComponent implements OnInit {  
3     oldAmount: number;  
4     newAmount: number;  
5  
6     constructor(private route: ActivatedRoute,  
7                 private amountService: AmountService) {}  
8  
9     ngOnInit() {  
10         this.oldAmount = this.route.params['oldAmount'];  
11         this.newAmount = this.amountService.calculateNewAmount(this.oldAmount);  
12     }  
13 }
```



typescript

```
1 //...  
2 export class AccountantComponent implements OnInit {  
3     oldAmount: number;  
4     newAmount: number;  
5  
6     constructor(private route: ActivatedRoute,  
7                 private amountService: AmountService) {}  
8  
9     ngOnInit() {  
10         this.oldAmount = this.route.params['oldAmount'];  
11         this.newAmount = +this.amountService.calculateNewAmount(this.oldAmount);  
12     }  
13 }
```



typescript

```
1 //...  
2 export class AccountantComponent implements OnInit {  
3     oldAmount: number;  
4     newAmount: number;  
5  
6     constructor(private route: ActivatedRoute,
```



```
7         private amountService: AmountService) {}
8
9     ngOnInit() {
10         this.oldAmount = +this.route.snapshot.params['oldAmount'];
11         this.newAmount = this.amountService.calculateNewAmount(this.oldAmount);
12     }
13 }
```



typescript

```
1 //...
2 export class AccountantComponent implements OnInit {
3     oldAmount: number;
4     newAmount: number;
5
6     constructor(private route: ActivatedRoute,
7                 private amountService: AmountService) {}
8
9     ngOnInit() {
10         this.oldAmount = this.route.params['oldAmount'];
11         this.newAmount = this.amountService.calculateNewAmount(this.+oldAmount);
12     }
13 }
```

Le paramètre se récupère via la clé `params` de l'objet `snapshot` de `ActivatedRoute`. Il faut le **cast** en `number` avant d'appeler la méthode, car les `params` de route sont de type `string`.

[← Activez les routes avec ActivatedRoute](#)[Conclusion →](#)

## Le professeur

**Will Alexander**

Scottish developer, teacher and musician based in Paris.

---

[OPENCLASSROOMS](#)

---

[OPPORTUNITÉS](#)

---

[AIDE](#)

POUR LES ENTREPRISES



EN PLUS



 Français 

