

Semestre : 1 ☒ 2 ☐

Session : Principale ☒ Rattrapage ☐

Module : Application côté client

Enseignants : UP Web

Classes :

Documents autorisés : OUI ☐ NON ☒

Calculatrice autorisée : OUI ☐ NON ☒

Date : Heure: 11h

Nombre de pages : 6

Internet autorisée : OUI ☐ NON ☒

Durée : 1h30

Partie 1 : QCM (7 pts)

Choisissez la bonne réponse. Une seule réponse est correcte

- 1- Depuis quel package importer la classe EventEmitter?
 - a. events
 - b. protractor
 - c. @angular/core
 - d. @angular/events

- 2- Soit une propriété "nb" de type number déclarée au niveau du composant et initialisé à 5. Soit le code suivant au niveau du template du même composant: `<input type="number" [value]="nb" name="nb" >` la valeur de nb est : `{{ nb }}`.
En écrivant au niveau de l'input la valeur 10, c'est quoi la valeur de nb affiché au niveau du template?
 - a. 5
 - b. 10
 - c. 105
 - d. rien n'est affiché

- 3- C'est quoi le résultat du code suivant?
`<input type="text" value="Ahmed" name="nom" #p > {{ p }}`
 - a. Undefined
 - b. Ahmed
 - c. [object HTMLInputElement]
 - d. Rien

- 4- C'est quoi le service qui peut être utilisé pour extraire les paramètres de route dans un composant ?
 - a. Router
 - b. Route
 - c. ActivatedRoute
 - d. CurrentRoute

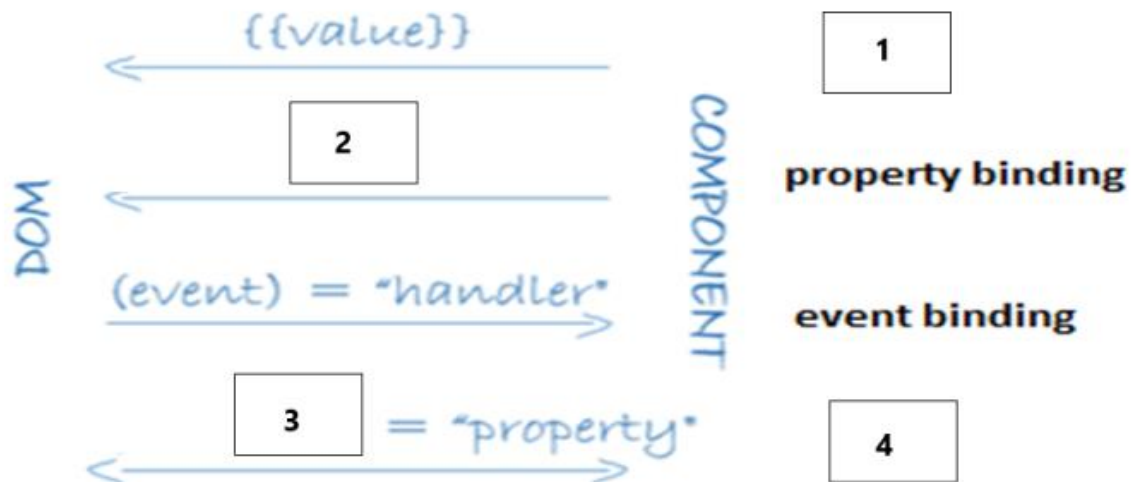
- 5- Le TS utilisé par Angular est-il interprété par les navigateurs ? Choisissez la bonne réponse.

- a. Non il faut compiler d'abord le TP en JS
 - b. Oui les nouveaux navigateurs comprennent TS
 - c. Oui grâce à un plugin Microsoft
 - d. Aucune de ces réponses n'est correcte
- 6- Où pouvons-nous utiliser les directives?
- a. Dans le template du composant
 - b. Dans la classe du composant
 - c. Dans le fichier CSS du composant
 - d. Dans le fichier HTML et TS du composant
- 7- Un service est injecté au niveau de la classe qu'elle va l'utiliser. Que peut être la nature de cette classe ?
- a. Un service
 - b. Un composant
 - c. Réponses A et B
 - d. Aucune de ces réponses

Partie 2 : Questions/Réponses (13 pts)

Question 1 : (3pts)

1. Complétez les numéros mentionnés sur la figure suivante :



- 1 :
- 2 :
- 3 :
- 4 :

2. Complétez les parties manquantes des codes ci-dessous

a.

```
@Component({
  selector: 'my-selector',
  template: `<h1>{{name}}</h1>
  Name : <input [(ngModel)]="name">`
})
export class TestComponent { 1 }
```

1 :

b.

```
@Component({
  selector: 'my-selector',
  template: `<div>
    <button (click)='clicked()'>Toggle</button>
    <div *ngIf="show"> can you see me ? </div></div>`
})
export class TestComponent { 2 }
```

2 :

.....

c.

```
@Component({
  selector: 'my-selector',
  templateUrl: './test.component.html'
})
export class TestComponent {
  @Input() count : number = 0 ;
  @Output() change 3 ;
  Increment(){
    this.count++ ;
    this.change 4 ; }
}
```

3 :

4 :

Question 2 : (1.5pts)

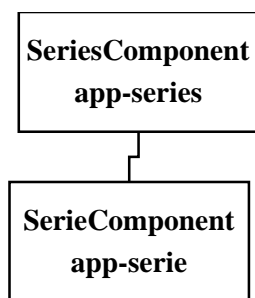
Pour le bon fonctionnement des différentes routes, corrigez le code ci-dessous :

```
const routes: Routes = [ {path: '**', component: ErrorsComponent}, {path: 'livres',  
component: LivreComponent}, {path: article, component: ArticleComponent}, ];
```

.....
.....
.....

Question 3 : (3pts)

Soit les deux composants SeriesComponent et SerieComponent définis comme suit :



Le composant SeriesComponent possède une propriété appelée **state** de type booléenne. Le composant SerieComponent récupère la valeur de cette propriété dans une propriété appelée **myState** depuis le composant SeriesComponent et l'applique pour afficher ou cacher un message

- a- Ecrivez la déclaration de la propriété **myState** dans le composant SerieComponent

.....
.....

- b- Ajoutez le code nécessaire permettant d'afficher ou de cacher un message du template du composant SerieComponent par la valeur récupérée depuis le composant SeriesComponent

```
<div>  
Mon message  
//template du compnant SerieComponent  
</div>
```

.....
.....

.....

Question 4 : (2.5pts)

Complétez le code HTML suivant pour afficher le bon message au bon moment, sachant que

- Le champ « username » est obligatoire et contient minimum 8 caractères.
- Le bouton « envoyer » est désactivé tant que le champ username est invalide.
- Un message personnalisé selon l'erreur saisi par l'utilisateur sera affiché.

<code><form</code> 1 <code>></code>	
<code><table></code>	
<code><tr><td> Username :</td></code>	<code><td><input type="text" name="username" </code> 2 <code>></code>
<code></td></code>	3
<code></tr></code>	
<code><tr></code>	
<code><td>Message :</td></code>	<code><td><textarea></textarea></td></tr></code>
<code><tr></code>	
<code><td></td></code>	<code><td><button </code> 4 <code>>Envoyer</button></td></tr></code>
<code></table></code>	

- (1) :
- (2) :
- (3) :
- (4) :

Question 5 : (3pts)

Soit un service `SerieService` contenant une méthode `getAllSeries()`. Un composant `SeriesComponent` utilise ce service pour récupérer la liste des séries.

- a-** Ecrivez la ligne de code permettant d'injecter le service `SerieService` au niveau du composant `SeriesComponent`.

.....

.....

- b-** Au niveau de la méthode `ngOnInit()` du composant `SeriesComponent`, écrivez le code nécessaire permettant de récupérer la liste des séries dans une propriété

appelée `listSeries`. Si une erreur survient, le message d'erreur doit s'afficher au niveau de la console.

```
ngOnInit(){
```

```
.....  
.....  
.....}
```

- c- Le composant `SerieComponent` affiche les informations d'une série. Sachant ceci, écrivez le code HTML du composant `SeriesComponent` permettant d'afficher la liste des séries contenus dans la propriété `listSeries` à l'aide du composant `SerieComponent`.

```
.....  
.....  
.....
```

Bon Travail ☺