

Test TP

Questions Théoriques : (5 pts)

Q1. Pour pouvoir utiliser Angular, on doit installer préalablement :

- ☐ JVM
- ☐ Node.js
- ☐ Winzip

Q2. Bootstrap est installé par défaut avec Angular ?

- ☐ Vrai
- ☐ Faux

Q3. Quel type d'application peut-on développer avec Angular ?

- ☐ Application embarquée
- ☐ Application mobile
- ☐ Application Web

Q4. Quel langage utilise Angular ?

- ☐ Typescript
- ☐ javascript
- ☐ PHP

Q5. Quelle est la commande CLI qui permet de créer un nouveau component ?

- ☐ ng generate
- ☐ ng new
- ☐ ng build

Q6. Pour lancer l'application angular dans le navigateur, on exécute la commande :

- ☐ ng generate --prod
- ☐ ng serve --open
- ☐ npm install

Q7. Si vous avez une variable `name = 'Salah'` dans votre code TypeScript, quelle syntaxe utilisez-vous pour afficher `Salah` dans le template ?

- ☐ les doubles accolades `{{ }}`
- ☐ les crochets `[]`
- ☐ les parenthèses `()`

Q8. Vous souhaitez qu'un bouton de votre template déclenche une fonction dans votre TypeScript : quelle syntaxe utilisez-vous pour l'y lier ?

- ☐ `(onClick)`
- ☐ `(click)`
- ☐ `[click]`

Q9. Pour qu'un bouton soit désactivé quand la valeur `authenticated` est `false` , quelle syntaxe devez-vous utiliser ?

- ☐ `<button disabled="!authenticated">`
- ☐ `<button (disabled)="!authenticated">`
- ☐ `<button [disabled]="!authenticated">`

Q10. Dans votre template, vous avez le sélecteur suivant :

```
<app-my-component [firstName]="Ali">
```

Quelle ligne de code vous attendez-vous à voir dans le component correspondant ?

- ☐ `const firstName: string;`
- ☐ `@Input() firstName: string;`
- ☐ `firstName() { }`

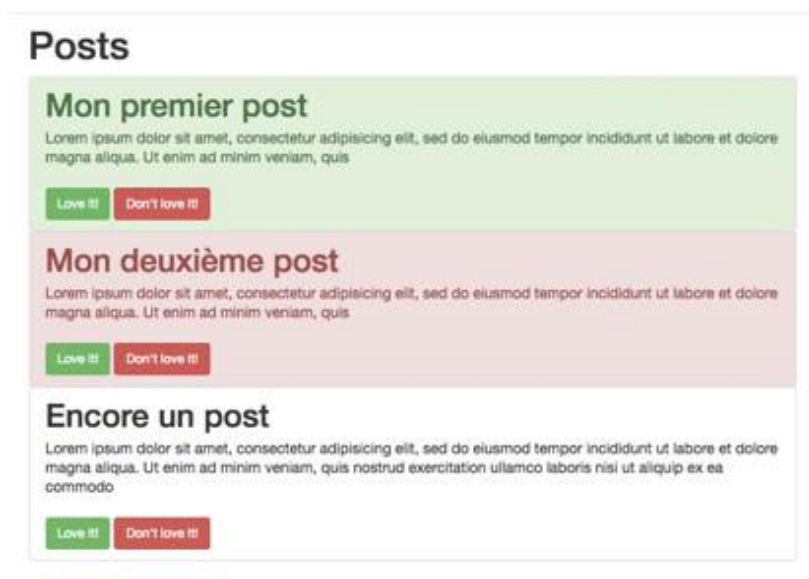
Test TP

Partie Pratique: (15 pts)

Vous allez créer une application simple de type blog. Cette application va afficher les posts du blog, et chaque post aura un bouton permettant de "**love it**" (*en vert*) ou de "**don't love it**" (*en rouge*) permettant d'augmenter et de diminuer le nombre de loveIts. Chaque post aura la forme suivante :

```
post: {  
  title: string,  
  content: string,  
  loveIts: number,  
}
```

- Utiliser Bootstrap. Si vous créez des `list-group-item` dans un `list-group`, vous avez les classes `list-group-item-success` et `list-group-item-danger` pour colorer les items.
- Votre `AppComponent` contiendra un `HeaderComponent` (qui contient le menu) et `PostListComponent`
- Votre `AppComponent` contiendra l'array des posts et il le passera à `PostListComponent`
- Votre `PostListComponent` affichera un `PostListItemComponent` pour chaque post dans l'array qu'il a reçu
- Chaque `PostListItemComponent` affichera le titre et le contenu du post dans le template
- Les `PostListItemComponent` auront des boutons qui permettent d'augmenter et de diminuer le nombre de loveIts.



Exemple

Quelques astuces

- Pensez aux différents types de databinding — comment passer des données d'un component à un autre, comment afficher des données dans le template et comment réagir à un événement venant du template
- Pensez aux directives structurelles comme `*ngFor`, `*ngIf`
- Pensez à colorer les `list-group-item` selon le nombre des loveIts du post :
 - Si `loveIts > 0`, le `list-group-item` sera vert
 - Si `loveIts < 0`, le `list-group-item` sera rouge