



TP3 : LIAISON DE DONNEES (DATA BINDING)

Matière : Atelier Framework Coté Client

Enseignante : Sonia Guerbouj

Classes : DSI2, MDW2

Durée : 3h

Objectif

Le but de ce TP est d'assimiler le fonctionnement de la liaison de données dans une application Angular.

1- Composant Angular (Rappel)

Dans une application Angular, un composant est un élément d'un module. Il est créé grâce à la commande `ng generate component nom_composant`.

Chaque composant est décrit par :

- Une classe représentant sa logique métier (fichier `.ts`)
- Un template HTML représentant sa vue.
- Une feuille de style CSS.
- Un fichier de test unitaire (fichier `.spec.ts`)

NB : Il est possible d'ajouter, à la commande `ng generate component`, les options :

`--skipTests=true` : pour ne pas générer les fichiers de test `spec.ts`

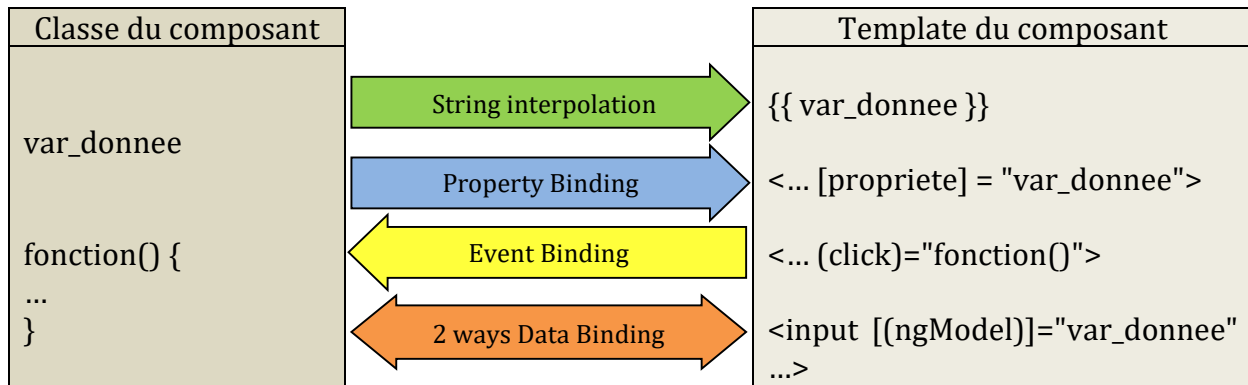
`--inlineStyle=true` : pour ne pas générer le fichier de styles `.css`

2- Liaison de données

La liaison de données ou Data Binding permet d'assurer un lien entre les données de la classe d'un composant et son template associé (vue).

Il y a 4 types de Data Binding :

- Interpolation `{{ }}` : de la classe du composant vers le template.
- Liaison d'une propriété `[]` : de la classe du composant vers le template.
- Liaison d'un évènement `()` : du template vers la classe du composant.
- Liaison bidirectionnelle `[()]` : dans les 2 sens entre la classe et le template.



3- Travail à faire

Ouvrez l'éditeur de code (VS Code) et créez tous les projets Angular sous le dossier « DSI21 ».

Exercice 1

- 1) Ouvrez l'éditeur de commande (cmd) et créez un projet Angular intitulé **TP3**.
- 2) Dans le projet **TP3**, créez un composant **info**.
- 3) Dans le fichier « **app.component.html** », écrivez le code suivant

```
<h1> Data Binding dans Angular</h1>
<app-info></app-info>
```

- 4) Lancer le projet avec la commande `ng serve --open`. Que s'affiche-t-il ?
- 5) Dans « **info.component.ts** », insérez dans la classe le code suivant :

```
export class InfoComponent implements OnInit {
  coordonnees = {
    nom : 'Sonia Guerbouj',
    email : 'guerbouj_sonya@yahoo.fr',
    tel : '22 333 444'
  };
  etat : boolean = true ;

  constructor() { }
  ngOnInit(): void {
  }
}
```

- 6) Dans « **info.component.html** », affichez les coordonnées dans une liste ainsi :

```
<ul>
  <li>Nom : {{ coordonnees.nom }}</li>
  <li>Email : {{ coordonnees.email }}</li>
  <li>Tél : {{ coordonnees.tel }}</li>
</ul>
<button [disabled]="etat">Désactivez ceci</button>
```

- 7) Lancez le serveur `ng serve`. Vérifiez dans un navigateur à l'adresse <http://localhost:4200/>, que les infos sont bien affichées et
- 8) Modifiez l'état du bouton grâce à la variable `etat`, vérifiez dans le navigateur.

Exercice 2

Dans le projet **TP3** de l'exercice 1, modifiez les fichiers du composant « **info** ».

- 1) Dans « **info.component.ts** », insérez dans la classe le code suivant :

```
txt : string = "Event Binding Example" ;
modifText() {
  this.txt = " Exemple de liaison d'évènement" ;
}
```

- 2) Dans « **info.component.html** », insérez ce code après le bouton « Désactivez » :

```
<button (click)="modifText()">Mettre à jour le text </button>
<h4>{{ txt }}</h4>
```

- 3) Testez le bouton. Que se passe-t-il ?

Exercice 3

Dans le projet **TP3** de l'exercice 1, ajoutez un autre composant « **comment** ».

- 1) Ecrivez dans le fichier « **comment.component.ts** » :

```
export class CommentComponent implements OnInit {
  tab : any[] = [] ;
  nouveau = {
    id = 0,
    msg = '
  };

  ajouterCommentaire() {
    this.tab.push({
      id : this.nouveau.id,
      msg : this.nouveau.msg
    }) ;
    this.nouveau = {
      id : 0 ;
      msg : ''
    };
  }

  constructor() { }

  ngOnInit(): void {
  }
}
```

2) Modifiez le fichier « **comment.component.html** » en y insérant :

```
<input type="text" [(ngModel)]="nouveau.msg">
<button (click)="ajouterCommentaire()">Ajouter
commentaire </button>
<ul>
  <li *ngFor="let elem of tab">{{elem.msg}}</li>
</ul>
```

3) Dans le fichier « **app.component.ts** » il faut importer le module FormsModule :

```
import { FormsModule } from '@angular/forms';
...
@NgModule({
  declarations: [
    AppComponent,
    CommentComponent
  ],
  imports: [
    BrowserModule, FormsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
...
```

4) Réaffichez la page Web 127.0.0.1 :4200. Testez en ajoutant des commentaires.