 Se former autrement HONORIS UNITED UNIVERSITIES	EXAMEN : Application Côte Client 1  Semestre : 1 <input type="checkbox"/> 2 <input checked="" type="checkbox"/> Session : Principale <input checked="" type="checkbox"/> Rattrapage <input type="checkbox"/>
Module: Application Côte Client Enseignant(s): UP web Classes : 1CINFO2 Documents autorisés : OUI <input checked="" type="checkbox"/> NON <input type="checkbox"/> Nombre de pages : 4 Calculatrice autorisée : OUI <input type="checkbox"/> NON <input checked="" type="checkbox"/> Internet autorisée : OUI <input type="checkbox"/> NON <input checked="" type="checkbox"/> Date : 20/07/2025      Heure: 18:30h      Durée :1h30	

**A noter avant de démarrer :**

- 1- Seulement l'exécution du code est notée.
- 2- Les questions sont notées séparément. Si jamais vous voulez dépasser une question, vous pouvez faire ce qui vous convient pour pouvoir réaliser les questions d'après.
- 3- Le package « json-server » représente la partie backend de notre application et le fichier « db.json » représente la base de données.

**Enoncé :**

Vous devez réaliser une application Angular 16 permettant à un bibliothécaire de gérer les livres, les auteurs et les emprunts. Le backend est simulé à l'aide de json-server. Le fichier db.json représente la base de données locale.

**Travail demandé :**

- 1- Créer les composants et écrire le code nécessaire pour que :
  - NavBarComponent : Créez un composant avec les liens suivants : Liste des livres, Liste des auteurs, Emprunter un livre. Intégrez ce composant dans AppComponent. (1pts)

**NavBarComponent** contenant trois liens comme indiqué dans la **Figure 1** et insérer le dans le composant AppComponent.

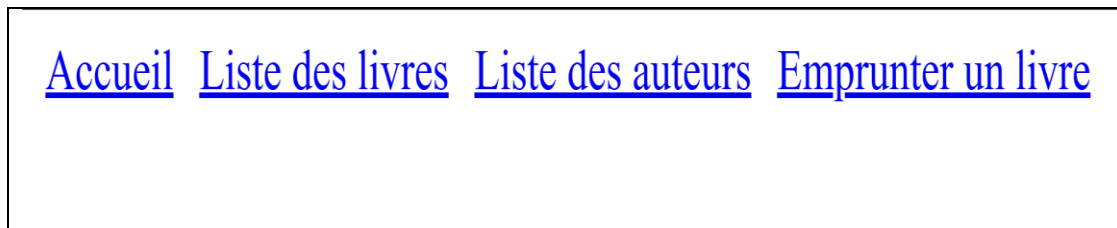


Figure 1

- FooterComponent : Affichez le message « Bibliothèque 2025 » et intégrez-le dans AppComponent. **(1pts)**

**FooterComponent**, insérer-le dans le composant AppComponent affichant le message " Bibliothèque 2025 ", comme indiqué dans la **Figure 2**

- HomeComponent : Affichez un message d'accueil et les statistiques (nombre de livres et d'auteurs) au lancement de l'application. **(1pts)**



Figure 2

Soit les modèles de données « Auteur », « Livre » et « Emprunt » suivants. **(1pts)**

Auteur	Livre	Emprunt
id!: string; nom!: string; nationalite!: string;	id!: string; titre!: string; annee!: number; auteur: Auteur = new Auteur(); disponible: boolean = true;	id!: string; livre: Livre = new Livre(); dateEmprunt!: string; dateRetour!: string;

Alimenter le fichier db.json par le contenu suivant. **(1pts)**

```
{
  "auteurs": [
    { "id": "1", "nom": "Victor Hugo", "nationalite": "Française" },
    { "id": "2", "nom": "Jane Austen", "nationalite": "Anglaise" }
  ],
  "livres": [
    { "id": "1", "titre": "Les Misérables", "annee": 1862, "auteur": { "id": "1", "nom": "Victor Hugo", "nationalite": "Française" }, "disponible": true },
    { "id": "2", "titre": "Orgueil et Préjugés", "annee": 1813, "auteur": { "id": "2", "nom": "Jane Austen", "nationalite": "Anglaise" }, "disponible": true }
  ],
  "emprunts": []
}
```

2- ListeLivresComponent : Affichez tous les livres disponibles. Pour chaque livre, affichez titre, auteur, année, et un bouton "Emprunter"

- Affichage correct de tous les livres disponibles : **(2pts)**
- Affichage des propriétés : titre, auteur, année : **(1pts)**
- Bouton "Emprunter" bien intégré et fonctionnel : **(1pts)**
- Rafraîchissement de la liste après emprunt (livre devient indisponible) : **(2pts)**

Liste des livres			
Titre	Auteur	Année	Action
Les Misérables	Victor Hugo	1862	Emprunter
Orgueil et Préjugés	Jane Austen	1813	Emprunter

**Figure 3**

3- - ListeAuteursComponent : Affichez la liste des auteurs avec leur nom et nationalité. **(1 pt)**

- Affichage des auteurs avec nom et nationalité : **(2pts)**

Liste des auteurs	
Nom	Nationalité
Victor Hugo	Française
Jane Austen	Anglaise


**Figure 4**


- 4- Au niveau du composant **Emprunt** ajouter le formulaire de la **Figure 4(1pts)**

### Emprunter un livre

Prénom :

Nom :

Date d'emprunt :  

Date de retour :  

**Figure 5**

- 5- Sachant que tous les champs sont obligatoires, afficher un message d'erreur pour chaque champ indiquant que ce champ est obligatoire. **(1pts)**
- 6- Le bouton est désactivé tant qu'il existe un champ non rempli. Une fois tous les champs sont remplis le bouton devient actif. **(1pts)**
- 7- En cliquant sur le bouton « valider » : **(3pts)**
- Champs obligatoires : prénom, nom, date d'emprunt, date de retour.
  - Validation : tous les champs sont obligatoires, la date de retour doit être postérieure ou égale à la date d'emprunt.
  - Après validation, ajoutez l'emprunt dans db.json et redirigez vers la liste des livres.
  - UpdateEmpruntComponent : Permet de modifier uniquement la date de retour d'un emprunt.