

Atelier n°10 :

Le service HttpClient

Objectifs :

- Installer json-server
- Manipuler les méthodes de HttpClient : GET, POST, PUT, DELETE

Travail demandé :

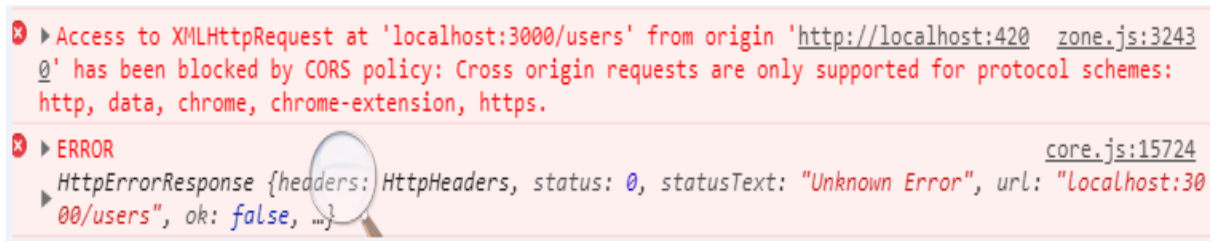
- 1- Installez json-server en tapant la commande : **npm install -g json-server**
- 2- Au niveau du terminal du projet courant, tapez la commande :
json-server --watch db.json
⇒ Un fichier db.json est généré automatiquement sous votre projet.
- 3- Mettez à jours ce fichier en ajoutant un attributs **invoices** avec des données de votre choix.
- 4- Testez maintenant l'url localhost:3000/invoices, vous devez avoir la liste des invoices contenu dans le fichier db.json
- 5- Créez un modèle de données appelé **Invoice (si ceci n'est pas déjà fait)** comme suit :

```
export class Invoice {  
  idInvoice : number;  
  discountAmount: number;  
  billAmount: number;  
  dateBill : string;  
  Status: Boolean;  
}
```

- 6- Créez le service InvoiceService et implémentez dedans les différentes méthodes de CRUD sur le modèle Invoice.
- 7- Le lien MainInvoice qui se trouve au niveau du header pointe sur le composant MainInvoiceComponent déjà créée.
- 8- Injectez le service InvoiceService dans les composants appropriés que vous devez créées (si ceci n'est pas déjà fait) afin de consommer les méthodes de ce service et développez les interfaces nécessaires.
- 9- Après chaque action de suppression, ajout et modification réaffichez la liste des invoices.

Configuration de proxy

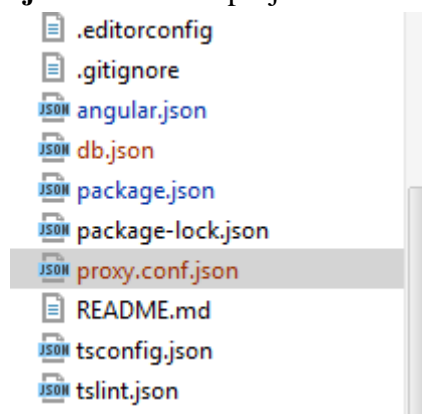
Problème :



Solution :

Cette solution est pour résoudre le problème de CORS. Bien sûr les url et les exemples mentionnés changent en fonction de votre besoin et le serveur que vous utilisez.

- 1- Créez le fichier **proxy.conf.json** sous votre projet en dehors du src.



- 2- Mettez dedans la configuration suivante :

```
{
  "/api/*": {
    "target": "http://localhost:3000",
    "secure": false,
    "logLevel": "debug",
    "changeOrigin": true,
    "pathRewrite": {"^/api" : ""}
  }
}
```

NB: la configuration ci dessus est pour la résolution de l'erreur rencontrée lors de l'utilisation du serveur "json-server". Cette configuration change en fonction de l'adresse de votre api.

- 3- Au niveau du fichier angular.json, ajoutez la ligne suivante :

```

"serve": {
  "builder": "@angular-devkit/build-angular:dev-server",
  "options": {
    "proxyConfig": "proxy.conf.json"
  },
  "configurations": {
    "production": {
      "browserTarget": "app4TWIN2:build:production"
    },
    "development": {
      "browserTarget": "app4TWIN2:build:development"
    }
  },
  "defaultConfiguration": "development"
},
"extract-i18n": {
  "builder": "@angular-devkit/build-angular:extract-i18n",

```

- 4- Au niveau de votre service, l'url ne doit plus pointer directement sur l'url (exemple : "localhost:3000/invoices"), mais elle doit pointer sur le préfixe choisi au niveau du fichier proxy.conf.json, ici c'est **/api** suivi par le path souhaité.
Exemple: L'url devient: **invoicesUrl:string="/api/invoices"**;

```

invoicesUrl : string = "/api/invoices";
getInvoices(): Observable<Invoice[]>{
  return this.http.get<Invoice[]>(this.invoicesUrl);
}

```

- 5- Redémarrez votre application.