

Objectif :

- Exécuter l'application « coffeeSupplier » en local (testé avec Laragon)
- cette application Node.js Express.js MySQL MVC gère des fournisseurs (CRUD)
- Découvrir le code de cette application Node Express MySQL

### Application « coffeeSupplier »

-Télécharger l'archive coffeeSupplier.zip

C'est une application fournie dans les labs AWS Developer.

« This is a simple CRUD app built with Express. »

-Copier l'archive coffeeSupplier dans le dossier www de Laragon  
et décompresser l'archive

-En Terminal, se déplacer dans le nouveau dossier « coffeeSupplier »

-Ouvrir le dossier avec VisualStudioCode

1 npm install //voir au passage le fichier package.json

2 créer la base de données « coffee »

Les instructions se trouvent dans readme.md

!\ le service MySQL doit être démarré sur le port TCP 3306

3 modifier le fichier ./app/config/config.js

```
let config = {  
  APP_DB_HOST: "localhost",  
  APP_DB_USER: "root",  
  APP_DB_PASSWORD: "",  
  APP_DB_NAME: "coffee"  
}
```

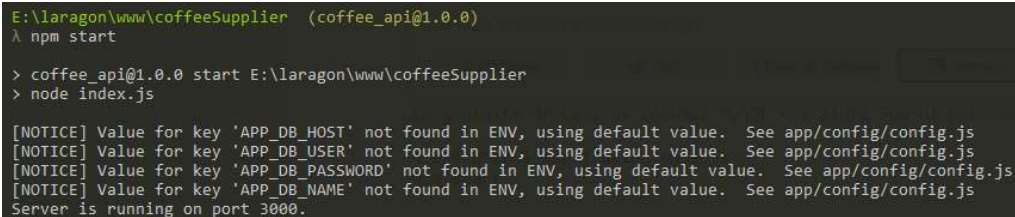
4 modifier la ligne 7 du fichier package.json

```
"start": "node index.js"
```

Ces étapes 3 et 4 permettent la connexion à la base de données MySQL locale.

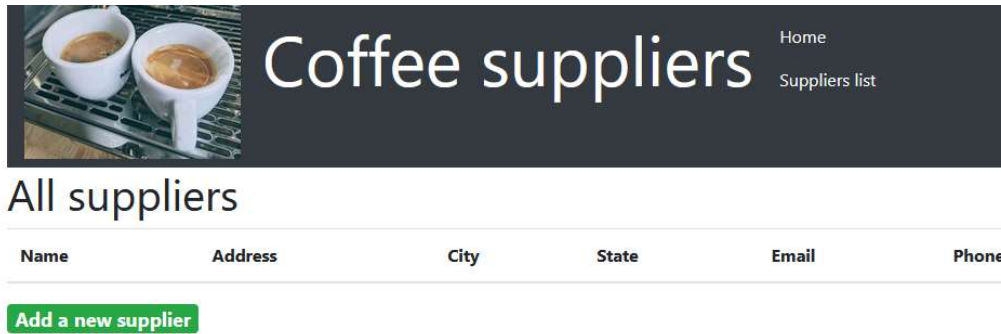
5 exécuter l'application node.js

npm start



```
E:\laragon\www\coffeeSupplier (coffee_api@1.0.0)  
λ npm start  
  
> coffee_api@1.0.0 start E:\laragon\www\coffeeSupplier  
> node index.js  
  
[NOTICE] Value for key 'APP_DB_HOST' not found in ENV, using default value. See app/config/config.js  
[NOTICE] Value for key 'APP_DB_USER' not found in ENV, using default value. See app/config/config.js  
[NOTICE] Value for key 'APP_DB_PASSWORD' not found in ENV, using default value. See app/config/config.js  
[NOTICE] Value for key 'APP_DB_NAME' not found in ENV, using default value. See app/config/config.js  
Server is running on port 3000.
```

Tester l'application dans le navigateur <http://localhost:3000>



- Ajouter un fournisseur.
- Vérifier l'ajout du fournisseur dans la liste.
- Editer le fournisseur, modifier un champ, vérifier.
- Supprimer le fournisseur.
- Tester une URL erronée : <http://localhost:3000/pipo> //erreur 404
- Arrêter le service MySQL et lister les fournisseurs //erreur 500

## Découvrir-Adapter le code de l'application avec de nouveaux besoins

### Nouveau Besoin1 : Afficher le nombre de fournisseurs

//ici on affiche (2) c'est le nombre de « lignes » qu'affiche la liste des fournisseurs



All suppliers (2) ←

Name	Address	City	Stat
malongo	monde	monde	mon
Colombian	69 Turbigo	Paris	Fran

### Nouveau Besoin2 : Disposer du contact (nom prénom) chez le fournisseur

//ici juste connaître les nom prénom de la personne qui représente le fournisseur

//l'entreprise contacte cette personne pour ses commandes de café

### Nouveau Besoin3 : Gérer les cafés (CRUD)

//ici on gère les différents cafés (moka, arabica, grains, moulu...)

//indépendamment des fournisseurs de café

//nouvelle table cafe

//cafe(id, libelle, pays, torrefaction, format)

//proposition de jeu d'essai

Id (autoincrément)	Libelle	Pays	Torrefaction	Format
1	Blend Prerequisite	Brésil	Blonde	Moulu
2	Moka Goba	Ethiopie	Brune	Moulu
3	Arabica Cordoba	Mexique	Blonde	Grain
4	Arabica Cordoba	Mexique	Brune	Grain

//CRUD : il faut pouvoir lister les différents cafés, créer un nouveau, modifier, supprimer

### Nouveau Besoin4 : Associer les cafés aux fournisseurs

//les fournisseurs fournissent des cafés, un fournisseur peut ne pas proposer tous les cafés.

Adapter la base de données //lien fournisseur et café

//pour l'entreprise, le besoin est de trouver un fournisseur pour un café (notamment en rupture de stock)

L'application doit permettre de lister les fournisseurs d'un café sélectionné.