



## TP : Serveur NodeJS

### 1. Serveur générique pour site statique (à la mano)

Partant de l'existant, vous devez rendre votre serveur générique, afin d'exposer toutes vos ressources statiques (html, css, js, img, etc...).

Tip : vos ressources se trouvent toutes dans un même dossier. Basez-vous sur l'url demandée dans la requête afin de lire et renvoyer la valeur du fichier correspondant.

### 2. Construction d'un site dynamique avec Express

Dans l'exercice précédent, nous devions construire nous même notre serveur et définir l'intégralité de sa configuration.

Nous allons cette fois, utiliser Express pour construire notre site.

#### Documentation

Avant de démarrer, faites une recherche sur NPM, ExpressJS et Handlebars.

#### Création du serveur

Créer un serveur Express qui répond sur l'url par défaut.

Tip : installez la librairie « express » dans votre projet.

#### Intégrer Handlebars pour le templating

Afin d'afficher le contenu de vos page, utilisez le moteur de templating Handlebars.

Tip : installez « express-handlebars » dans votre projet. Attention, veillez à respecter l'arborescence et les extensions de vos fichiers pour le fonctionnement d'Handlebars.

## Récupérer des données

Afin de retourner des données dans vos pages, vous devez créer une connexion à la BDD classicmodels existante.

Tip : installez et utilisez la librairie mysql.

## Retourner la liste des produits

Créer une page spécifique, permettant de lister l'ensemble des produits de la table « products ».

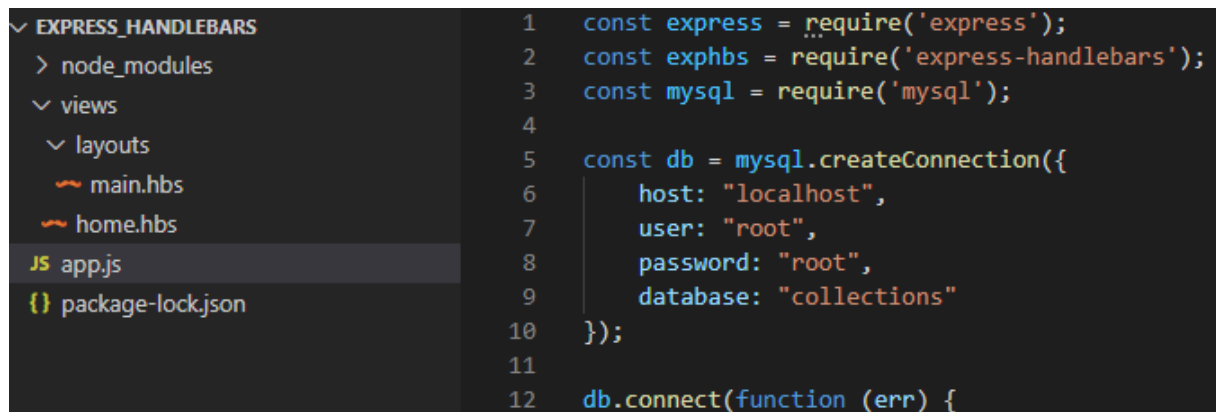
Tip : utilisez Handlebars afin d'intégrer vos données dans une page html existante.

## Intégrer un champ de recherche

Sur la page listant les produits, vous devez intégrer un champ de recherche, permettant de recherche parmi le nom des produits.

Tip : la recherche doit effectuer la même requête que la récupération des produits mais en intégrant un paramètre de recherche supplémentaire.

Final Tip :



```
1  const express = require('express');
2  const exphbs = require('express-handlebars');
3  const mysql = require('mysql');
4
5  const db = mysql.createConnection({
6    host: "localhost",
7    user: "root",
8    password: "root",
9    database: "collections"
10  });
11
12  db.connect(function (err) {
```