|  |  |
| --- | --- |
|  | **JavaScript côté serveur** |
| **Le framework Express.js** | |

Objectif :

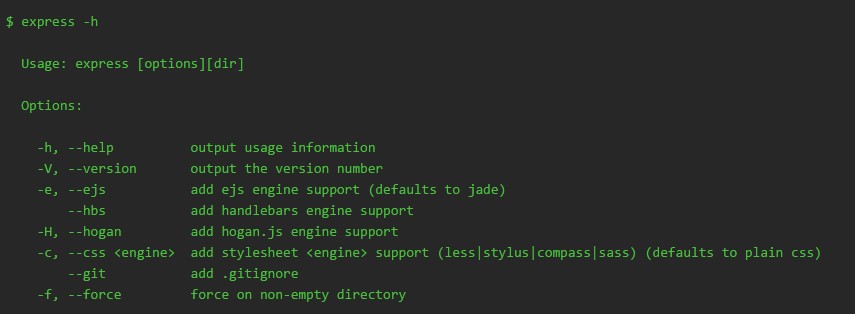
Cet atelier consiste à générer la structure d’un projet Express.js à l’aide ***d’express generator***.

Générateur d’applications Express :

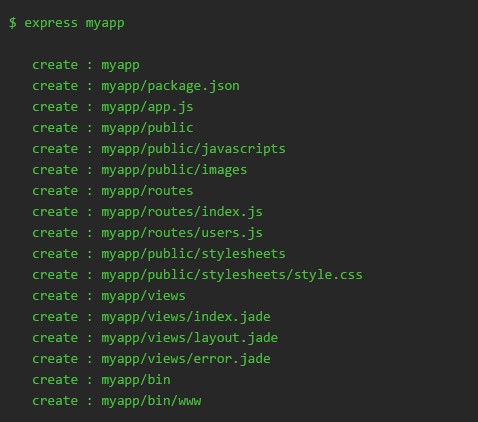
* Utilisez l’outil de générateur d’applications, express, pour créer rapidement un squelette d’application.
* Installez express à l’aide de la commande suivante :



* Affichez les options de commande à l’aide de l’option **-h** :



* Ce qui suit crée une application Express nommée ***myapp*** dans le répertoire de travail en cours :



* Ensuite, installez les dépendances :



* Sous MacOS ou Linux, exécutez l’application à l’aide de la commande suivante :

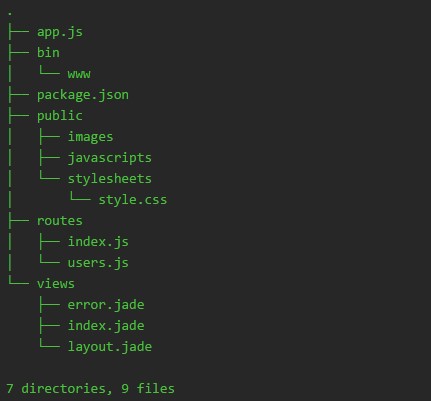


* Sous Windows, utilisez la commande suivante :



* Ensuite, chargez ‘http://localhost:**3000**/’ dans votre navigateur pour accéder à l’application.

L’application générée possède la structure de répertoire suivante :



**NB** : La structure d’application créée par le générateur est l’une des nombreuses manières possibles de structurer les applications Express. Vous avez toute latitude pour l’utiliser ou la modifier en fonction de vos besoins.

Exercice:

Cet exercice va fournir des données JSON aux différentes routes.

1. Générez le squelette d'une application Express en utilisant [express-generator.](http://expressjs.com/starter/generator.html)

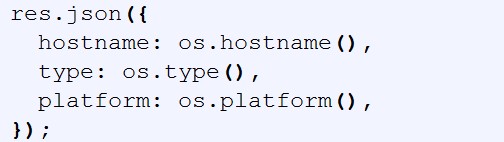
Regardez comment fonctionne les routes déjà existantes (folder */routes*).

Regardez dans *app.js* comment ces routes sont renseignées.

1. [Ajoutez une route](http://expressjs.com/starter/basic-routing.html) à votre projet: **/os/**
   1. Créez un fichier **os.js** sous */routes*

Pensez à utiliser le module **os** : voir [documentation](http://www.tutorialspoint.com/nodejs/nodejs_os_module.htm) pour récupérer les données relatives à votre OS.

* 1. La réponse à retourner est sous format JSON



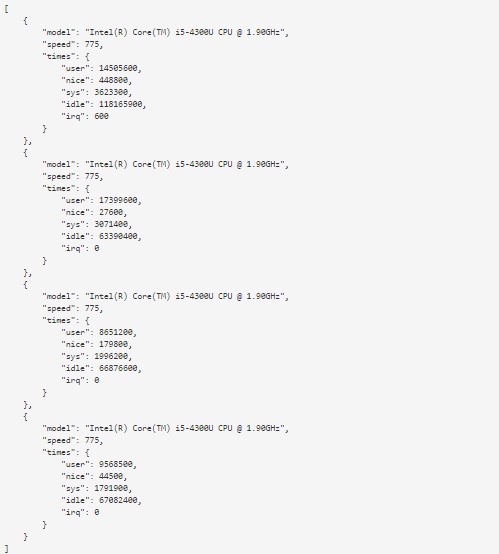
Affiche les informations sur le serveur.

**Réponse**:

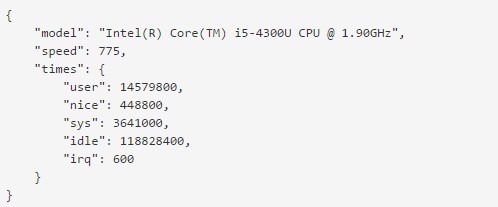


1. Ajoutez la route: **/os/cpus/** : Affiche la liste des processeurs.

**Réponse** :

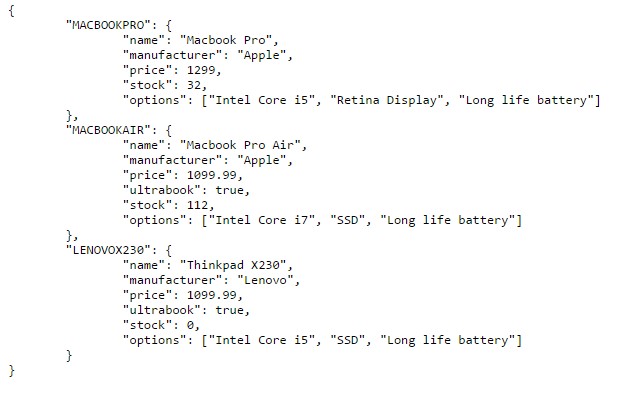


1. Ajoutez la route: **/os/cpus/:id** : Affiche les informations relative à un seul processeur. **Exemple :** /os/cpus/0 **Réponse :**



1. Nous allons maintenant implémenter un autre contrôleur avec Express pour une gestion de stocks : */products/*
2. Créez la route : **/products/** : Affichela liste des produits depuis **products.json** au format JSON.

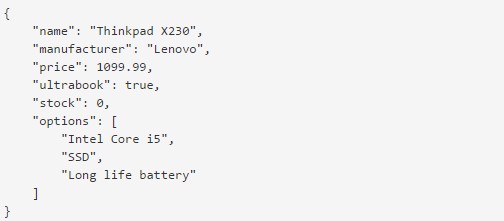
**Réponse :**



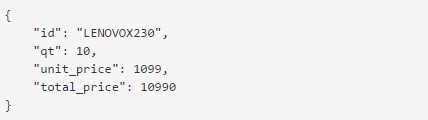
1. Ajoutez la route : **/products/:id** : Affiche les détails du produit associé à id.

**Exemple :** /products/LENOVOX230

**Réponse :**



1. Ajoutez la route : **/products/:id/:qt** : Affiche le prix total pour **qt** fois le prix du produit **id**. **Exemple** : /products/LENOVOX230/10 **Réponse** :



1. Ajoutez la route : **/products/instock/:qt** : Afficher seulement les produits dont le stock est supérieur ou égal à **qt**.

**Exemple** : /products/instock/100 **Réponse** :

