

環境構築

じゃーGulpを使ってみよう！と言いたいところなのですが使う為には環境構築をする必要があります。

MacとWindowsではそれぞれ手順が違うので別々に書いていきます。

共通としてはNode.jsをインストールしてもらいます。

Node.jsとは

簡単に言うとJavaScriptは本来フロント側で使用する言語であって、PHPやRubyといったサーバーサイドの実装はできません。

ですが、Node.jsを使用すればサーバーサイドをJavaScriptで動かすことができるようになります。（本気を出せばJavaScriptだけでアプリを作ることも可能）

ここの部分をもう少し踏み込んだ話をする、ChromeにはJavaScriptを処理する為のエンジン（V8エンジン）が搭載されています。

それが搭載されているのでブラウザ上でJavaScriptが処理できています。

それと同じエンジンをサーバー側にいれることでJavaScriptでもサーバーサイドの実装が可能となります。

Node.jsについてもっと詳しく知りたい方は下記のリンクで詳しく解説しているので読んでみてください。

※公式ドキュメントは必ず見るようにしましょう！

[Node.jsについて](#)

[Node.js公式ドキュメント](#)

なぜフロントでもNode.jsを必要とするのか

上の説明だとNode.jsはサーバーサイドの実装を行う為に使うものという認識を持つと思います。

フロントとしての大きな役割はパッケージ管理を行うために使用します。

GulpなどもNode.jsでパッケージ化されています。

それを管理するためにNode.jsをインストールする必要があります。

このパッケージ管理する為のツールのことを**Node Package Manager**と言い、略称は**npm**と言います。

[npm公式ドキュメント](#)

Node.jsをインストールするバージョン

Node.jsも日々更新されているので複数のバージョンが存在しています。

最新版はv14.2.0（2021/04）

Node.jsは奇数番号が開発版、偶数が安定版と言われてます。

ただ、安定版の最新のものをいれるのはリスクがあります。

いくらNode.jsが最新でもパッケージ側がそれに対応しているかは不明です。

場合によってはバージョンが合わないことによる動作不良が起こる場合があります。

なので安定版で推奨されるものをインストールしましょう。

14系も推奨される日が来るので偶に調べるようにしましょう。

現在（2021/04）の推奨版は**v12.16.2**なのでこれをインストールしてください。

最新版をインストールしてもかまいませんがそれで引き起こされるエラーは対応しかねますので自己責任でお願いします。

ここまでできたら[Gulpの導入](#)から作業を進めてください。

Macの環境構築

まずはHomebrewをインストールをしてください。

インストールの前に`brew -v`をターミナルで実行してバージョン確認を行なってください。

Homebrew `0.0.0`と表示される場合はすでにインストールされています。もし表示されない場合は下記リンクに沿ってインストールしてください。

Xcode Command Line Toolsのインストールが必要な場合があります。インストールに時間がかかるので終わるまで落ち着いて待ちましょう。

読み進めていくとインストールできたかを確認と言った小見出しがあると思います。そこまでできたら作業終了して次に進んでください。

MacにHomebrewをインストールする方法と基本的な使い方

次にnodebrewとNode.jsのインストールを行なってください。

Homebrewをインストールすることに少し触れていますが、そこは無視してください。

ここで出てくるnodebrewはNode.jsのバージョンを管理する為のものです。

`npm -v`でバージョン確認ができればそれ以降の作業はしなくて大丈夫です。(nodebrew helpでどんなコマンドがあるのかだけは確認しておいてください。)

`nodenv global v12.16.2`を実行しておいてください。

これはすべてのディレクトリで指定したバージョンを反映させる為のコマンドです。

今回は1つしかインストールしてませんが今後別バージョンをインストールして気づかない内にバージョンが変わらないようにする為です。

`nodenv local 0.0.0`作業ディレクトで実行するとディレクトリ単位でバージョンを指定でき、グローバル設定よりも優先されます。

今回は必要ないですが今後必要になる時が来るかもしれませんので覚えておいてください。

Node.js を Mac にインストール

Windowsの環境構築

まずは`node -v`と`npm -v`を実行してインストールされているか確認してください、

`VO.O.O.O`、`O.O.O.O`と表示されたらインストールされていますが、もし表示されない場合は下記リンクに沿ってインストールしてください。

記事と違うのはインストールファイル画面です。

記事通りにインストールすると最新版がダウンロードされてしまう為です。

開くとリンクが複数あるので`node-v12.16.2-x64.msi`または`node-v12.16.2-x86.msi`をダウンロードしてインストールしてください。

どちらをダウンロードするかは使用しているPCがx86(32ビット)とx64(64ビット)、どちらのCPUを搭載しているのかで変わります。

それはこちらでは確認できないのでご自身で確認してもらってどちらかをダウンロードしてください。

確認する方法についてはリンクを貼っておきますので参考にしてください。

インストール後バージョン確認が取れたら作業終了です。

[Node.jsの開発環境を用意しよう！](#)

[Node.js v12.16.2](#)

[【Windows 10版】自分のPCがx86かx64かを確認する方法](#)

ここまでできたら[Gulpの導入](#)から作業を進めてください。

Gulpの導入

ここからはMac、Windows共通です。

ターミナル（コマンドプロンプト）を開き下記のコマンドでフォルダーを作成してください。

```
mkdir gulp_practice
```

そして、今作成したフォルダーの中まで移動してください。

移動ができれば下記のコマンドを実行してください。

```
npm init -y
```

`package.json`と言うファイルが作成され、同じ構造になっていれば問題ないです。

nameは`npm init -y`を実行したフォルダー名に依存するのでここだけ違っていても問題ないです。

```
{
  "name": "gulp",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

`package.json`はインストールしたパッケージを記載、開発時に実行するスクリプトを記載するものです。

案件に入って作業を行う前にどんなスクリプトがあるのか、どんなパッケージがあるのかなどはここを見て確認してください。

次に`npm install gulp`を実行してください。

実行後`package.json`の`dependencies`の部分に...

```
"dependencies": {
  "gulp": "^4.0.0"
}
```

このようになっていれば無事インストールできています。

この時に`package-lock.json`、`node_modules`と言うものが作られます。

`package-lock.json`の中身を開いて見てください。

いろいろ書いてあって訳わかんないと思います。

その中でgulpと検索して見てバージョンを見てください。

`package.json`と同じバージョンが記載されています。

これはなんなのかと言いますと、先ほど実行した`npm install gulp`ですがバージョンの指定はしていません。

バージョンを指定していないので最新のバージョンがインストールされています。

複数人で開発する時、個々にそうするとバージョンの統一性がなくなってしまいます。

`package-lock.json`を用意しておくとしインストール時、このファイルに記載されているバージョンにしたがってインストールしますのでバージョンの統一性が生まれます。

また、1度削除したパッケージでもここに記載されていればインストールした時に以前と同じバージョンをインストールできます。

`package-lock.json`はバージョン情報を厳密に管理するものと覚えておいてください。

`node_modules`はnodeが機械的に用意したもの、インストールしたパッケージのファイルなどが入っています。

ただかなり膨大で人が把握仕切れない部分なので気にしなくて大丈夫です。

次に`npm install gulp-sass`を実行してください。

`dependencies`の部分に`gulp-sass`とバージョンが記載されていたらインストール成功です。

ここまでできたら`node_modules`を削除してください。

`npm install`を実行してください。

そうすると`node_modules`が再び作られます。

これを実行すると`package-lock.json`を見てインストールを行い、`node_modules`を作成します。

複数人で開発している時に他の方が新しいパッケージを入れた時に`npm install`を実行する必要があります。

追加しても気づかない人がほとんどなので追加した時は共有するのがルールになっています。

ここまでできたら環境構築は終わりです。

補足

`npm install` グローバルインストールとローカルインストールがあります。

- グローバルインストール

`npm install -g gulp`

- ローカルインストール

`npm install -D gulp`

今回実行したコマンドもローカルインストールです。

グローバルインストールをしてしまうと使用しているPC全体に影響が出てしまいます。

そうすると必要のないパッケージの影響を受けて開発に支障をきたす場合もあるので基本的にはグローバルインストールはしないようにしてください。

課題

以下の項目を調べてClassroomに提出してください。

念の為Mac、Windowsのどちらで環境構築をしたのか記載しておいてください。

- Node.jsとは
- Node.jsをインストールするバージョンで気を付けるポイント
- Node.jsをインストールするバージョンで気を付けるポイント
- Node Package Managerとは
- nodebrewとは（Mac使用者のみ）
- package.jsonとは
- package-lock.jsonとは
- `npm install` パッケージ名を実行する際にしなければならないことが3つあります。公式ドキュメントで実際にパッケージ名を調べて確認するなどして上げてください。
- 今回作業してできたファイルの中で1つGit管理の不要なものがあります。それは何か、なぜそれが不要なのか理由も書いてください。