



## 8. Expressの基礎

### Express とは

Express は Node.js で利用できるWebサーバです。MVC アプリケーションフレームワークで構成されており、`npm` で簡単にパッケージをインストールできます。

### Express の基本

Express では `express()` でサーバを作成します。

```
const app = express();
```

### サーバの待機

サーバを起動すると、ポート番号とホストで待機状態になります。デフォルトはポート「3000」、ホスト「localhost」です。

```
app.listen(ポート, ホスト, () => {  
  //処理  
});
```

### ルーティング

Express は、`get()` や `post()` など簡単に HTTPクエストを検知できます。指定のURLでアクセスすると、対応のリクエストメソッドとコールバック関数が実行されます。

### GET リクエスト処理

```
app.get(パス, コールバック関数);
```

## POST リクエスト処理

```
app.post(パス、コールバック関数);
```

## すべてのリクエスト処理

```
app.all(パス、コールバック関数);
```

# Express でサーバ作成

**Express** でWeb表示をしてみましょう。 ファイル構成と利用モジュールは以下のとおりです。

## ファイル構成

```
express
├─ node_modules/
├─ .env
├─ package-lock.json
├─ package.json
└─ server.js
```

## モジュール

- express
- env

# Express のインストール

## npm 初期化

**npm** で初期化します。

```
% npm init -y
```

# パッケージインストール

---

`express`、`dotenv` パッケージをインストールします。

```
% npm install express dotenv
```

## 設定ファイル .env 作成

---

プロジェクト内に `.env` を作成してホストとポート番号を記述します。

```
HOST=localhost  
PORT=3000
```

## サーバ作成

### server.js の作成

---

プロジェクト内に `server.js` を作成します。

### モジュール読み込み

---

`Express` モジュールを読み込みます。

```
const express = require('express');
```

設定ファイルから、変数 `host` と `port` に設定します。

```
const host = process.env.HOST;  
const port = process.env.PORT;
```

### Express 作成

---

`Express` サーバを作成します。

```
const app = express();
```

## GETリクエスト

---

/ で GET リクエストしたときの処理を `get()` で実装します。コールバック関数は、リクエスト&レスポンスのオブジェクトが引数になります。

```
app.get("/", (req, res) => {  
  console.log(req.body);  
  console.log(req.url);  
  console.log(req.query);  
  res.send("Hello Express!");  
});
```

## サーバの待機

---

サーバの待機処理を `listen()` で実装します。

```
app.listen(port, host, () => {  
  console.log(`Server listen: http://${host}:${port}`);  
});
```

## サーバ起動確認

**Express** サーバを起動して Webを表示してみます。

## node でサーバ起動

---

ターミナルで `server.js` を実行します。

```
% node server  
Server listen: http://localhost:3000
```

## ブラウザで確認

---

ブラウザで <http://localhost:3000> にアクセスします。「Hello Express!」が表示されれば成功です。

← → ↺ 🏠 ⓘ localhost:3000

Hello Express!

## GETパラメータの確認

<http://localhost:3000?keyword=apple> のように **GET** パラメータでアクセスしてみます。

← → ↺ 🏠 ⓘ localhost:3000/?keyword=apple

Hello Express!

ターミナルでログで、**GET** パラメータが取得できるか確認してみましょう。

```
undefined
/?keyword=apple
{ keyword: 'apple' }
```

## Express サーバ停止

**Express** サーバ停止したいときは、起動中のターミナルで Ctrl + C を押します。

## ソース

```
const express = require('express');

require('dotenv').config();
const host = process.env.HOST;
const port = process.env.PORT;

const app = express();

app.get("/", (req, res) => {
  console.log(req.body);
  console.log(req.url);
  console.log(req.query);
  res.send("Hello Express!");
});
```

```
app.listen(port, host, () => {  
  console.log(`Server listen: http://${host}:${port}`);  
});
```

## 演習

### 問題1

---

`server.js` を修正して、`/profile` に `GET` アクセスしてレスポンスを返してみましょう。※プログラムを修正したら、サーバの再起動が必要です。

当サイトの教材をはじめとするコンテンツ（テキスト、画像等）の無断転載・無断使用を固く禁じます。これらのコンテンツについて権利者の許可なく複製、転用等する事は法律で禁止されています。尚、当ウェブサイトの内容をWeb、雑誌、書籍等へ転載、掲載する場合は「ロジコヤ」までご連絡ください。