



2. Microsoft Learn の「Node.js と Express を使用して Web API を構築する」に取り組み、以下を報告する。

(ア) 作成したものの説明

Github を使って、json データを返すアプリケーションアプリを作った。

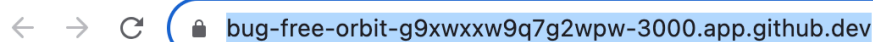
(イ) 自分が理解したこと


今回取り組んだことにより、webAPI の構築するための手順を理解することができた。

(ウ) どんなことに役立つか

ブラウザやコードのクライアントを使用して要求を送信できることがわかったので、社会などに出て新しいサービスを開発するために利用できると思う。

(エ) 作成した WebAPI が動いていることがわかる画面ショットを貼り付けること

A screenshot of a web browser's address bar and content area. The address bar shows a URL: bug-free-orbit-g9xwxw9q7g2wpw-3000.app.github.dev. The content area below the address bar displays the text "Hello World!".

← → ↻  bug-free-orbit-g9xwxw9q7g2wpw-3000.app.github.dev

Hello World!

```
▼ [
  ▼ {
    "id": 1,
    "name": "hammer"
  },
  ▼ {
    "id": 2,
    "name": "screwdriver"
  },
  null,
  ▼ {
    "id": 3,
    "name": "wrench"
  }
]
```

(オ)「知識チェック」の結果について、画面ショットを貼り付けること

それぞれの質問に最も適した回答を選んでください。その後、[回答を確認] を選択します。

1. Express を使用して Web アプリケーションを構築するために必要な手順は、次のどれですか? \*

☐ アプリをインスタンス化し、ルートを構成し、ミドルウェアを設定し、エラー ハンドラーを設定し、サーバーをリスンする

☒ アプリをインスタンス化し、サーバーをリスンする

✓ 正解です。アプリを起動して実行するには、これらの手順だけが必要です。ルートをいくつか構成することを強くお勧めします。

☐ アプリをインスタンス化し、ルートを構成し、サーバーをリスンする

☐ アプリをインスタンス化し、ルートを構成し、ミドルウェアを設定し、サーバーをリスンする

2. Express アプリから JSON 応答を送信する方法として、推奨されるのは次のどれですか? \*

☒ 応答オブジェクトで `json()` ヘルパー メソッドを呼び出す: `res.json({ content: ' ' })`

✓ 正解です。JSON として応答を送信する方法はいくつもありますが、この方法が最も一般的であり、簡単に使用できます。

☐ `res.send({ content: ' ' })` を呼び出す

☐ `res.send(JSON.stringify({ content: ' ' }))` を呼び出す

次のいずれかの方法を使用する: `res.type('json')`、`res.type('application/json')`、`res.contentType('application/json')`、`res.format({ 'application/json': function() {`

☐ `res.send(JSON.stringify({ content: ' ' }));` を呼び出す

☐ 次のいずれかの方法を使用する: `res.type('json')`、`res.type('application/json')`、`res.contentType('application/json')`、`res.format({ 'application/json': function() { res.send({}) } })`

3. JSON データを含む Post 要求を処理するように Express を設定するにはどうすればよいですか? \*

☐ `app.post(<route>, () =>{})` のように `post` メソッドを使用してルートを登録し、`req.body` オブジェクトから読み取る

☐ 本文解析ミドルウェアを構成し、`app.post(<route>, () =>{})` のように `post` メソッドを使用してルートを登録して、`req.data` オブジェクトから読み取る

☒ 先頭で `app.use(bodyParser.json())` を呼び出し、`app.post(<route>, () =>{})` のように `post` メソッドを使用してルートを登録して、`req.body` オブジェクトから読み取る

✓ 正解。この呼び出しでは、受信データを JSON として解釈するように `bodyParser` が構成されます。

☐ 先頭で `app.use(bodyParser.urlencoded({ extended: false }));` を呼び出し、`app.post(<route>, () =>{})` のように `post` メソッドを使用してルートを登録して、`req.body` オブジェクトから読み取る

モジュールが完了しました:

実績のロックを解除