

非同期通信 (Ajax)

非同期通信とは

送信者のデータ送信タイミングと受信者のデータ受信タイミングを合わせずに通信を行う通信方式ことになります。

同期通信は、データ通信のリクエストを出してからレスポンスが来るまでほかの処理を行わずにレスポンスを待ち続けるが、非同期通信ではレスポンスを待っている間にほかの処理を行える。ほかの処理を行っている際に、レスポンスを受信すると受信処理を実行する。

非同期通信は、近年利用されているAjaxの通信方式としても利用されています。

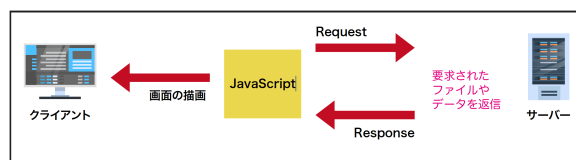
Ajaxとは

Ajax (Asynchronous JavaScript And XML) は、ブラウザ内で非同期通信を行いながらインターフェースの構築を行うプログラミング手法になります。

XMLHttpRequest として知られる技術 (Web サーバーとの非同期通信) を用いて、Web ページ全体を再読み込みせずに、ページの表示などを更新することができます。

Web システムやWeb アプリ、SPA (Single Page Application) など、最近のWeb サイトで広く利用されている技術となります。

最近では、XMLHttpRequest からFetch API を利用した非同期通信の実装が主流となっております。



AJAX - 開発者ガイド | MDN

Asynchronous JavaScript and XML (AJAX) はそれ自体がある種の技術というわけではありません

<https://developer.mozilla.org/ja/docs/Web/Guide/AJAX>

Fetch API

Fetch API を利用すると、リクエストやレスポンスといったHTTP のパイプラインを構成する要素を操作できるようになります。

また、**fetch()** メソッドを利用することで、非同期通信を簡単にわかりやすく実装できるようになります。

フェッチ API - Web API | MDN

フェッチ API は (ネットワーク越しの通信を含む) リソース取得のためのインターフェイスを提供し

mdn web docs

フェッチ API の使用 - Web API | MDN


フェッチ API は、リクエストやレスポンスといったプロトコルを操作する要素にアクセスするための JavaScript インターフェイ


mdn web docs

fetch() メソッドは、**Promise** を返すためリクエスト/ レスポンスの結果を待ってから、then() メソッドで次の処理を実行させることができます。

Promise - JavaScript | MDN

Promise オブジェクトは、非同期処理の完了 (もしくは失敗) の結果およびその結果の値を表します。

 https://developer.mozilla.org/ja/docs/Web/JavaScript/Reference/Global_Objects/Promise

 mdn web docs

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8">
    <title>Web演習 2 - サンプル 13</title>
  </head>
  <body>
    <h1>Ajax</h1>
    <div>
      <p>都道府県の選択</p>
      <select id="pref"></select>
    </div>

    <div>
      <p>市区町村一覧</p>
      <div id="city"></div>
    </div>
    <script src="sample13.js"></script>
  </body>
</html>
```

```
{
  // RESAS API関連
  const APIKey = "FP93jNInkB3RkU194NzBYKMqoZLD3eORqDSZqd6s";
  const APIEndpoint = "https://opendata.resas-portal.go.jp";
  const prefAPI = "api/v1/prefectures"; // 都道府県一覧API
  const citiesAPI = "api/v1/cities"; // 市区町村一覧API

  // Element
  const selectPref = document.querySelector( "#pref" );

  /**
   * 都道府県セレクトメニューの作成
   * 都道府県一覧API
   */
  fetch(
    `${ APIEndpoint }/${ prefAPI }`, // 非同期通信をおこなうURL
    {
      // リクエストヘッダーの設定
      headers: {
        'X-API-KEY': APIKey, // RESAS APIはheader情報としてAPI KEYを送
      }
    }
  )
  .then( ( response ) => { return response.json() } )
  .then(
    ( data )=>{
      console.log( data );
      const prefs = data.result;
      prefs.forEach(( pref ) => {
        console.log( pref );

        // elementの作成
        const item = document.createElement( "option" );
        // 作成したelementのプロパティ設定 (value属性と表示)
        item.value = pref.prefCode;
        item.innerText = pref.prefName;
        // selectに作成したelementを追加
        selectPref.append( item );
      }); // endforeach
    }
  );
}
```