

インターネット基礎理論 10

動的コンテンツ

はじめに

資料置き場

<https://sammyppr.github.io>

に授業資料を置いていきます。復習に使ってください。

欠席した場合などはスライドを確認して追いつくようにしましょう。

スライドあるなら授業受けなくていいや

なんてことは思わないようにお願いします。



今日すること

文書を構造的に記述するHTML

装飾を担当するCSS

について学んできました。

第10回目の今日は、静的・動的コンテンツの区別、及び、動的コンテンツを実現する技術について説明します。JavaScriptについても触れていきます。

何に役立つ？

前回ソースコードの見方を教えたが、
書けなくても
読めるだけで

こんな感じで作られてるんだ

とわかることができます。これからの世の中、社会に出るとホームページの担当になることもあるでしょう。そんな時に、仕組みを理解しておくことで役に立つことがあると思います。

コンテンツ

コンテンツの創造、保護及び活用の促進に関する法律

「コンテンツ」とは、映画、音楽、演劇、文芸、写真、漫画、アニメーション、コンピュータゲームその他の文字、図形、色彩、音声、動作若しくは映像若しくはこれらを組み合わせたもの又はこれらに係る情報を電子計算機を介して提供するためのプログラムであって、人間の創造的活動により生み出されるもののうち、教養又は娯楽の範囲に属するものをいう。

ホームページにおけるコンテンツ

ホームページにおけるコンテンツとはページの情報自体のことを指します。

そして、それらはおおまかに2つに分類することができます。

- 静的コンテンツ
- 動的コンテンツ

静的コンテンツ

静的コンテンツとはこれまで例示して来たように、HTMLの内容が固定されており、状況に応じず普遍的に同じ内容が表示されるものです。

動的コンテンツ

それに対し、HTMLの内容が可変となるものを「動的コンテンツ」と呼びます。

内容が動いている、と考えるとわかりやすいかもしれません。

- フォームのエラーメッセージ
- GoogleMap
- ECサイト

等、多くのものが動的なコンテンツとなっています。

動的コンテンツの作成方法

動的コンテンツには作成方法が大きく分けて2つあります。

- ブラウザ上で実行されるタイプ
- Webサーバ上で実行されるタイプ

ブラウザ上では主に、JavaScriptと呼ばれる言語が利用されます。

Javascript

Javascriptとは

ブラウザ上で実行されるタイプにはJavascript言語があります。

後にJavaという言語について触れますが、全く異なるものですので、これだけは間違えないでください。現在webの仕事をしていても、本当に間違える人が多いです。恥ずかしいので気をつけてください。

言語とは？

そもそも、言語ってなんですか？

XML,HTMLは「構造化言語」でしたね。文章の意味を構造として明示する言語でした。

今回扱うのは「プログラミング言語」のことでコンピュータに対する一連の動作の指示を記述するための人工言語です。

コンピュータが直接理解できるものは「機械語」と呼ばれる言語です。これは16進法等で表記され、人間がみても全くわかりません。(一部の人は理解できますが...)

そのため、機械語とほぼ1対1対応のニーモニックを用いたアセンブリ言語という、多少人間が読みやすくした言語があります。しかし、これも利用しやすいとは言えません。

高級言語

そのため、人間が利用する言語に近い言語として高級言語が開発されました。

これらの言語はCPUは直接理解することができないため、翻訳する必要があります。この翻訳方法に以下の二通りがあります。

- **インタープリタ**: 実行時に翻訳しながら実行する方法
- **コンパイラ**: 開発時に実行可能なプログラムに変換してから実行する方法

インタープリタの方が処理速度が遅いという欠点があります。

JavaScriptに話を戻します。

うちの大学のホームページでJavaScriptが記載されている部分を見てみましょう。

プログラミング言語は英語を基本としているので、英語が分かれば、何となく人にわかりやすく書かれている、ということは伝わるかな...

DOM

HTMLをブラウザは解釈しDOM(Document Object Model)というものを生成し、CSSを解釈しながらボックスモデル等からレイアウトを計算し、それを元に表示をしています。

- DOMとは？仕組みや構造を解説！HTMLやJavaScriptとの関係は？

JavaScriptでできること

JavaScriptではDOMにアクセスし、その要素に追加・修正・削除することができます。

このようにして、Webページの中身を書き換えたりしながらJavaScriptでは動的表現を実現しています。

最近、JavaScriptは早く実行することができるようになっていたため、次のようなホームページも実装することができるようになっています。

- [three.js periodic table](#)

ライブラリ

JavaScriptを1から全部書かなくても、便利なツールをひとまとめにした**ライブラリ**と呼ばれるものが存在しています。

無料で利用できるものも多いため、便利だな、と開発者が思うと一気に世界中に利用が広がります。例えば

- [Lightbox](#)

Webではこういうものを組み合わせながら開発していきます。

WebAssembly

もっと高速に処理したいというニーズのもとWebAssemblyと呼ばれる技術も実用段階に入っています。

- ビデオやオーディオの編集
- Webブラウザでのゲーム
- 科学分野での視覚化とシミュレーション
- コンピュータゲームのエミュレーター

こういうことをするには、WebAssemblyが向いています。デザインツールとして有名なFigmaにても利用されています。

フロントエンド・バックエンド

動的コンテンツ

繰り返しになりますが

- ブラウザ上で実行されるタイプ
- Webサーバ上で実行されるタイプ

と説明しましたが、

Webの開発では

- フロントエンド WEBサービスやアプリケーションの見た目に関する部分の開発
- バックエンド WEBサービスやアプリケーションの機能面やデータなどに関する部分の開発

に分けることができます。

ロードマップ

それぞれに必要な技術・知識をまとめたロードマップというものが公開されています。ちょっと見てみましょう。

- [Frontend Developer](#)
- [Backend Developer](#)

バックエンドでやっていること

動的コンテンツでは、ユーザにあった情報などをリクエストに応じて、その場で情報を組み合わせてクライアントにデータを渡します。

- Webサーバ
- プログラミング言語
- データベースサーバ

この三つが協調して作業することが多いです。

バックエンドで利用される言語

- 【2024年】バックエンド開発におすすめ言語の人気ランキング

あれ？1位にJavaScriptが...

バックエンド言語としてのJavaScript

JavaScriptはもともとフロントエンド用言語として利用されてきましたが、インタプリタ型ではあるものの、そこそこ処理速度が上がったため、node.js(サーバサイドでJavaScriptを実行する環境)を用いて、バックエンドでも利用されています。

同じ言語でフロントエンド・バックエンド両方とも記述できることが、メリットとなります。

PHP

- **PHP** PHP: Hypertext Preprocessor

と呼ばれる、言語があります。HTMLと親和性が高く、.htmlを.phpと変更しても動作するほ
どです。数回後に紹介するwordpressでも利用されています。こんな感じ...

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>Hello World</title>
</head>
<body>
  <?php
    echo "Hello World";
  ?>
</body>
</html>
```

Java

JavaScriptとは全く異なる言語となります。

気をつけましょう。

1995年に発表され、もともとWebブラウザ上で動かすアプレットというものを動かすための言語で、環境に依存せず動くという特徴がありました。

2000年になるとWeb上のアプレット用の言語としての利用はされなくなり、企業システム向けのプログラミング言語として利用されていきます。

そして、Webサーバ用の開発言語としても利用されています。

データベースサーバ

情報を保存するにはテキストファイルを利用しても良いのですが、何十万件というデータを扱うためにはそれ専用に(DB)と呼ばれるアプリケーションを別途利用します。

代表的なものは

- MySQL
- PostgreSQL
- SQLite
- Oracle Database

等があります。

SQL

多くのアプリケーションの名前に「SQL」が入ってます。

SQLとはデータベースにおいて、データの操作や定義を行うためのデータベース言語(問合せ言語)があります。

情報の捜査における基本機能

情報操作するときの基本的な機能として

CRUD

と言われる概念があります。これらは

- *C* Create 生成
- *R* Read 読取
- *U* Update 更新
- *D* Delete 削除

の頭文字をとったものです。

Ajax

従来のWebアプリケーションでは、サーバにリクエストを送信後、レスポンスを新たにWebページとして受け取ることで画面遷移が発生していました。

ですが、Ajaxという技術により、画面遷移を伴わないWebアプリケーションが実現可能となります。

例えば、Googleマップで移動する度に新しいページがロードされたら面倒くさいですね。このAjaxを実現するためには

1. JavaScriptがサーバにリクエストする
2. 結果が戻ってきたら、JavaScriptが動的にページの一部を書き換える

という仕組みが取られています。

動的コンテンツまとめ

通常の場合

1. パラメータ付きのデータをサーバにリクエスト
2. サーバ側で必要に応じてデータベースなどにアクセスし、必要なデータをまとめる
3. クライアント側に送る
4. 結果を表示する

Ajaxの場合

1. ユーザの挙動から新たなデータをJavaScriptでリクエスト
2. サーバ側で、必要に応じてデータベースなどにアクセスし、必要なデータをまとめる
3. クライアント側に送る
4. 情報に応じてDOMを操作し表示を変更

これまでのまとめ

ここまで、

- コンピュータの基礎 (ハードウェア・ソフトウェア)
- インターネットの仕組み(サーバ・クライアント・DNS・IP)
- html,CSS
- JavaScript,DB

等を学んできました。一応、これらのことがわかれば、どのようにインターネットが技術的に成り立っているかの基礎は網羅したつもりです。

次回からは、もう少し違った角度からインターネットを見てみましょう。

時間余ったら

JavaScriptの実際についてやってみましょう。

- 変数(数値・文字列)・配列
- 四則演算等, =, 代入演算子
- 真偽値
- 制御文 if, for, while, switch
- 関数

小レポート

動的コンテンツがどんな仕組みで実現されているか簡単にまとめよ。
(manaba)