

Webプログラミング入門

【狙い】

- Webアプリケーション開発に必要なプログラミング技術を体系的に学ぶ。
- チーム開発を通じて、プロジェクト管理の実践的なスキルを磨く。

【到着目標】

- JavaScriptにより動的なWebアプリケーションを構築できるようになる。
- プロジェクトの計画、実装、発表までの一連の作業を統一して実行できる能力を身につける。

【授業方針】

- 実践重視：コードを書いて動かす体験を重視し、手を動かす中で理解を深める。
- 段階的学習：基礎から応用へ段階的に進め、無理のない学習を提供する。
- コラボレーション：ペアワークやチーム開発を取り入れ、実務に近い環境を体験させる。

授業計画

第1回：Webプログラミングの概要

- ・ Webの仕組み（HTTP、URL）
- ・ クライアントとサーバーの役割
- ・ モダンなWeb技術の紹介 ・ 開発環境の説明

第2回：JavaScriptの基本とWeb API入門

- ・ JavaScriptの基本：データ型、変数、関数の基本
- ・ Web APIの紹介
- ・ Javascriptを使って動的に画面更新

第3回：DOM操作とイベント処理

- ・ DOMの操作（取得、変更、追加）
- ・ 複数DOM要素とDOM操作の効率化
- ・ イベントリスナーを利用した動的UIの構築

第4回：フォームとデータ処理

- ・ 入力フィールドの操作
- ・ バリデーションと送信イベントの処理

第5回：React.jsの基礎

- ・ Reactプロジェクトのセットアップ、・ JSXの基本
- ・ コンポーネントの作成と利用

第6回：React.jsでの状態管理

- ・ 状態管理の基本（useStateの使用）
- ・ 状態を利用したUIの動的更新

第7回：React.jsでのイベント処理

- ・ イベントハンドラーの設定、・ フォームの入力と状態管理

第8回：React.jsとWeb APIの連携

- ・ useEffectフックを使ったデータフェッチ
- ・ 外部Web APIから取得したデータの表示

第9回：コンポーネント設計と再利用

- ・ コンポーネントの分割
- ・ 再利用可能なコンポーネントの作成

第10回：レスポンシブデザイン

- ・ CSSモジュールの利用
- ・ ブートストラップやTailwind CSSを使ったデザイン

第11回：Gitとバージョン管理

- ・ Gitの基本操作
- ・ GitHub/GitLabでの共同作業

第12回：プロジェクト計画

- ・ 要件定義、スケジュール管理、タスク分担

第13回：プロトタイプ作成

- ・ Reactを用いた簡易Webアプリケーションの開発

第14回：プロジェクトの発表準備

- ・ イベントハンドラーの設定、・ フォームの入力と状態管理

第15回：プロジェクト発表と振り返り

- ・ 学生プロジェクトの発表（内容はYoutubeに公開）
- ・ フィードバックセッション

見取り図

第1フェーズ: Javascript基礎知識の習得 (第1回～第4回)

目的: Webプログラミングの基本概念を理解し、JavaScriptの基礎を習得。

第1回: Webプログラミングの概要

第2回: JavaScriptの基本とWeb API入門

第3回: DOM操作とイベント処理

第4回: フォームとデータ処理

第2フェーズ: React.jsの習得 (第5回～第9回)

目的: モダンなフロントエンド技術React.jsを習得し、API連携を実現。

第5回: React.jsの基礎

第6回: React.jsでの状態管理

第7回: React.jsでのイベント処理

第8回: React.jsとWeb APIの連携

第9回: コンポーネント設計と再利用

第3フェーズ: 実践力の強化 (第10回～第15回)

目的: 学んだ技術を統合し、実践的なプロジェクトを経験。

第10回: レスポンシブデザイン

第11回: Gitとバージョン管理

第12回: プロジェクト計画

第13回: プロトタイプ作成

第14回: プロジェクトの発表準備

第15回: プロジェクト発表と振り返り