



WEB開発ワークショップ：タスク管理アプリ制作



概要

このワークショップでは、HTML、CSS、JavaScriptを使って **学習タスク管理アプリ** を段階的に構築します。 逆算学習法を採用し、完成形を先に見てから段階的に実装することで、効率的な学習を実現します。



学習目標

技術的スキル

- HTML**：セマンティックなマークアップとフォーム要素の活用
- CSS**：レイアウト、アニメーション、レスポンシブデザイン
- JavaScript**：DOM操作、イベント処理、動的コンテンツ制御

実用的スキル

- プロジェクトの段階的構築方法
- デバッグとトラブルシューティング
- コードの可読性と保守性の向上



フォルダ構成

```
6_06_再挑戦/  
├── 完成版/  
│   ├── index.html  
│   ├── style.css  
│   └── script.js  
├── 初心者向け/  
│   ├── テンプレート/  
│   │   ├── phase1_html_template.html  
│   │   ├── phase2_css_template.css  
│   │   └── phase3_javascript_template.js  
│   └── ヒント/  
└── # 🏆 最終成果物（デモ用）  
    # 🤖 手取り足取りガイド
```

```
| | | |—— phase1_html_hints.md
| | | |—— phase2_css_hints.md
| | | |—— phase3_javascript_hints.md
| | | |—— 説明資料/
| | | |—— ワークショップ説明資料案2.md
|—— 中級者向け/ # 🎯 穴埋め形式
| |—— 穴埋めテンプレート/
| | |—— index_fillblank.html
| | |—— style_fillblank.css
| | |—— script_fillblank.js
| |—— 解答例/
| | |—— index_answer.html
| | |—— style_answer.css
| | |—— script_answer.js
| |—— 説明資料/
| |—— ワークショップ説明資料案3.md
|—— 上級者向け/ # 🚀 最小限のガイド
| |—— 最小テンプレート/
| | |—— index_minimal.html
| | |—— style_minimal.css
| | |—— script_minimal.js
| |—— 説明資料/
| |—— ワークショップ説明資料案1.md
|—— 講師用資料/ # 📚 指導者向け
|—— ワークショップ企画_nitac_web班.md
|—— ユーザー体験シミュレーション.md
```

クイックスタート

1. 完成版のデモを確認

```
cd 完成版
# ブラウザでindex.htmlを開く
```

2. 自分のレベルに合わせて選択

初心者の方

- 初心者向け/テンプレート/ から開始
- 段階的なガイドに従って実装
- 困ったら ヒント/ フォルダを参照

中級者の方

- [中級者向け/穴埋めテンプレート/](#) から開始
- 穴埋め問題を解きながら実装
- 完成後は [解答例/](#) と比較

上級者の方

- [上級者向け/最小テンプレート/](#) から開始
- 最小限のヒントで自力実装
- 独自の機能追加にチャレンジ

実装する機能

Phase 1: HTML構造

- ☒ セマンティックな文書構造
- ☒ ヘッダーとナビゲーション
- ☒ 進捗表示エリア
- ☒ タスクリストエリア
- ☒ 完了メッセージエリア

Phase 2: CSS装飾

- ☒ モダンなレイアウトデザイン
- ☒ レスポンシブ対応
- ☒ ホバーエフェクト
- ☒ アニメーション効果
- ☒ カラーパレットとタイポグラフィ

Phase 3: JavaScript機能

- ☒ チェックボックスのイベント処理
- ☒ 進捗バーの動的更新
- ☒ 完了状態の視覚的フィードバック
- ☒ 完了メッセージの表示制御
- ☒ リアルタイムパーセンテージ計算



技術仕様

対応ブラウザ

- Chrome 80+
- Firefox 75+
- Safari 13+
- Edge 80+

技術スタック

- **HTML5** : セマンティック要素とアクセシビリティ
- **CSS3** : Flexbox、アニメーション、カスタムプロパティ
- **Vanilla JavaScript** : ES6+、DOM API、イベント処理

パフォーマンス目標

- First Contentful Paint: < 1.5秒
- Largest Contentful Paint: < 2.5秒
- Cumulative Layout Shift: < 0.1



学習のポイント

HTML学習ポイント

1. **セマンティックHTML** : `<header>`, `<section>`, `<article>` の適切な使用
2. **フォーム要素** : `<input>`, `<label>` の関連付け
3. **アクセシビリティ** : ARIA属性とキーボードナビゲーション

CSS学習ポイント

1. **Flexboxレイアウト** : モダンなレイアウト手法
2. **CSSカスタムプロパティ** : 保守しやすいスタイル管理
3. **アニメーション** : `transition` と `transform` の活用
4. **レスポンシブデザイン** : メディアクエリの実践的使用

JavaScript学習ポイント

1. DOM操作 : `querySelector` , `getElementById` の使い分け
2. イベント処理 : イベントリスナーとイベントオブジェクト
3. 状態管理 : アプリケーション状態の追跡と更新
4. 関数設計 : 単一責任の原則と再利用可能な関数

 よくあるトラブルと解決法

HTML関連

問題	原因	解決法
レイアウトが崩れる	タグの閉じ忘れ	HTMLバリデーターで確認
CSSが適用されない	class名の不一致	デベロッパーツールで確認

CSS関連

問題	原因	解決法
アニメーションが動かない	ベンダープレフィックス不足	autoprefixerの使用
レスポンスが効かない	viewport設定忘れ	metaタグの確認

JavaScript関連

問題	原因	解決法
要素が取得できない	DOMContentLoaded前の実行	イベントリスナーで包む
イベントが発火しない	セレクタの間違い	コンソールでデバッグ

 拡張チャレンジ

基本機能完成後の発展的な課題 :

レベル1: 機能拡張

- ☐ タスクの追加・削除機能
- ☐ タスクの編集機能
- ☐ 期限設定機能
- ☐ カテゴリ分類機能

レベル2: UI/UX改善

- ☐ ダークモード対応
- ☐ カスタムテーマ機能
- ☐ アニメーション強化
- ☐ マイクロインタラクション

レベル3: 高度な機能

- ☐ LocalStorageでのデータ永続化
- ☐ ドラッグ&ドロップ並び替え
- ☐ キーボードショートカット
- ☐ PWA対応

レベル4: 外部連携

- ☐ REST APIとの連携
- ☐ リアルタイム同期
- ☐ 通知機能
- ☐ データエクスポート



参考資料

公式ドキュメント

- [MDN Web Docs](#)
- [CSS-Tricks](#)
- [JavaScript.info](#)

ツール・サービス

- [CodePen](#) - オンラインコードエディタ

- [Can I Use](#) - ブラウザ対応状況確認
- [Web.dev](#) - パフォーマンス最適化

サポート

質問・相談

- ワークショップ中：講師またはTA（ティーチングアシスタント）に気軽に声をかけてください
- 後日：GitHub Issuesまたはメールでお問い合わせください

バグ報告

問題を発見した場合は、以下の情報と共にお知らせください：

- 使用ブラウザとバージョン
- 再現手順
- 期待される動作
- 実際の動作

ライセンス

このワークショップ教材は教育目的で作成されています。

NITAC（長岡工業高等専門学校）WEB班の学習活動の一環として制作されました。

Happy Coding! 

[Last updated: 2024年12月](#)