

## 1. 概要

### 目的:

ユーザーがいくつかの質問に回答することで、自分に合った職業を知るための診断Webアプリを制作。  
楽しく直感的に使えるUI(画面デザイン)と、診断結果に基づく簡易レコメンド(推薦)を特徴とする。

### 対象ユーザー:

職業に迷っている学生や転職希望者、自己分析をしたい人など。

## 2. 機能要件

### 2.1 質問機能

- ・複数の選択式質問
- ・各選択肢が職業に対してスコアを加算する
- ・回答選択後「次へ」ボタンで次の質問に遷移
- ・全問回答後に結果ページに遷移

### 2.2 スコア計算機能

- ・各職業に対してスコアを加算
- ・スコアの高い順に並べ、上位3件を抽出

### 2.3 結果表示機能

- ・最適職業1位をメイン表示
- ・上位3位までの職業をリスト表示
- ・スコア付きで表示

### 2.4 再診断・サーバー停止機能

- ・「もう一度診断する」ボタンでトップページへ戻る
- ・「閉じる」ボタンを押すと、アプリが安全に終了するようにサーバーへ終了リクエスト(/shutdown)sを送信

## 3. 非機能要件

### 3.1 デザイン/UI

- ・フォント: Google Fonts「Yomogi」
- ・カラートーン: やわらかいパステル調(背景: 水色、ボタン: ピンク系)

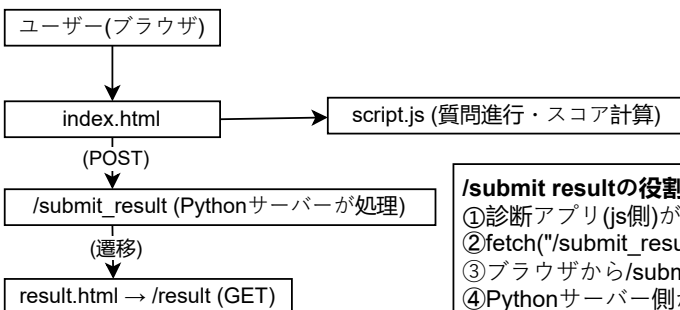
### 3.2 パフォーマンス

- ・ローカル環境で即時レスポンス
- ・軽量のJSON処理とシンプルなUIで高速表示

### 3.3 保守性(変更/追加/修正など)

- ・HTML / CSS / JavaScriptは分離(各要素の役割を明確にし、修正や拡張を容易)
- ・質問はJSONファイルで管理(questions.json)(アプリ本体のコードに手を加えることなく、質問の更新が可能)
- ・サーバーはPythonのhttp.serverベースでシンプル構成(外部ライブラリや複雑なフレームワークを不使用)

## 4. システム構成



### /submit resultの役割

- ①診断アプリ(js側)が診断結果を生成する
- ②fetch("/submit\_result",{method:"POST",body:JSON.stringify(...)})でサーバーに送信
- ③ブラウザから/submit\_resultにPOSTでJSONが送られてくる
- ④Pythonサーバー側がそれを受け取り、resurt\_dataに保存
- ⑤サーバーはクライアントに{"status":"OK"}というJSONレスポンスを返す
- ⑥後で/resultにアクセスすると、保存された診断結果がJSON形式で取得できる

## 5. 使用技術

- ・フロントエンド: HTML, CSS, JavaScript
- ・サーバーは Python 標準ライブラリのhttp.server.SimpleHTTPRequestHandlerとsocketserver.TCPServer
- ・データ形式: JSON

## 全体の構成

### ユーザーのブラウザ

- index.html(質問ページ)
  - script.js(質問の制御・結果計算)
  - style.css(見た目のデザイン)
- result.html(診断結果)
  - とじるボタン(/shutdownにリクエスト→サーバー切断リクエスト)
- questions.json(質問データ)

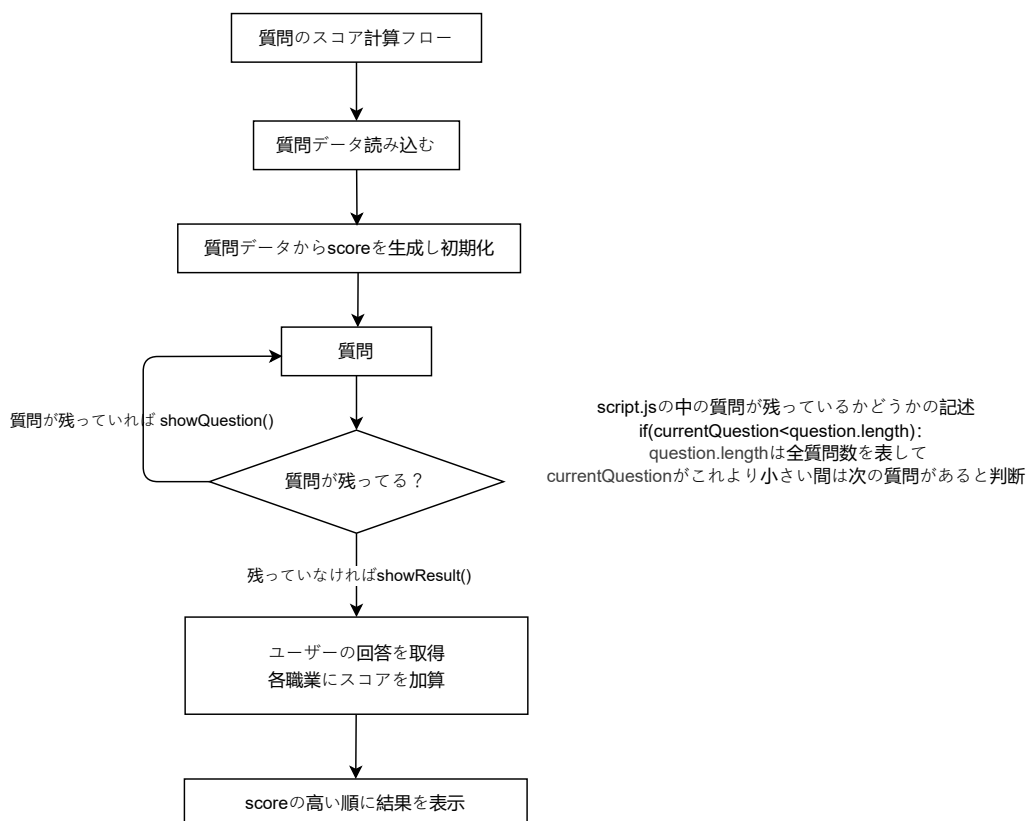
### ローカルサーバー(server.py)

- SimpleHTTPServerでHTML/CSS/JS/JSONを配信
- /shutdown リクエストでサーバー終了

## ファイル構成の役割

- index.html → アプリのメイン画面、質問表示、診断の流れを持つページ。
- style.css → 全体のレイアウト、ボタン・テキストのデザインスタイル設定。
- script.js → 質問の順番管理、ユーザーの選択処理、スコア計算、結果表示処理。(動き・計算・ユーザー操作の処理係)
- questions.json → 診断に使う質問内容と各選択肢の職業スコア影響データ。
- result.html → 結果表示用の別ページ(再診断・サーバー終了ボタン付き)。
- server.py → 簡易HTTPサーバー、HTML / CSS / JS / JSONファイル配信・/shutdown受付・サーバー終了管理。  
(ファイルを渡す & 結果を受け取り返す「司令塔」)

## 質問のスコアの計算フロー



①質問数10問  
②職業数48個

となっていますが内容も数もJSONファイルの書き換えで簡単にできます

## 職業一覧 + 必要スキル（今回入っている職業）

### クリエイティブ系

- ・デザイナー → デザインセンス、Photoshop/Illustrator、色彩感覚、プレゼン力
- ・小説家 → 創作力、文章力、構成力、読解力
- ・ゲームクリエイター → ゲーム企画力、プログラミング、グラフィック知識、ユーザー理解
- ・イラストレーター → デッサン力、デジタルツール、色彩感覚、独自のスタイル
- ・漫画家 → 作画力、ストーリー構成、キャラクターデザイン、締切管理
- ・動画クリエイター → 動画編集スキル、企画力、撮影技術、SNS活用
- ・Webデザイナー → HTML/CSS、UI/UX知識、デザインツール、SEO理解
- ・YouTuber → 企画力、動画編集、トーク力、マーケティング理解
- ・プログラマー → 高いゲームスキル、反射神経、戦略思考、忍耐力
- ・俳優 → 表現力、演技力、台本理解、柔軟な対応力

### 技術・分析系

- ・エンジニア → プログラミング、問題解決力、論理的思考、チーム開発経験
- ・データサイエンティスト → 統計知識、データ分析、Python/R、論理的思考
- ・AIエンジニア → 機械学習知識、Python、数学知識、最新論文理解
- ・セキュリティエンジニア → ネットワーク知識、セキュリティ対策、脆弱性診断、監視スキル
- ・プログラマー → コーディング力、デバッグ力、アルゴリズム知識、継続力
- ・システムエンジニア → 要件定義、設計力、システム構築、顧客対応力
- ・データアナリスト → データ処理スキル、Excel、SQL、可視化力
- ・研究者 → 探究心、論文執筆、分析スキル、プレゼン力

### 人と関わる系

- ・教師 → 教育力、コミュニケーション、忍耐力、指導力
- ・看護師 → 医療知識、対人スキル、判断力、チームワーク
- ・営業 → コミュニケーション力、提案力、交渉力、忍耐力
- ・コンサルタント → 課題解決力、分析力、プレゼン力、提案力
- ・カスタマーサポート → 共感力、傾聴力、課題解決力、忍耐力
- ・SNSマーケター → トレンド把握、SNS運用、企画力、分析力
- ・イベントプランナー → 企画力、調整力、交渉力、当日対応力
- ・ツアーガイド → 地域知識、説明力、語学力（場合による）、臨機応変さ
- ・広報 → 発信力、文章力、企画力、メディア対応力
- ・人事 → 採用スキル、面接力、評価スキル、調整力
- ・カウンセラー → 傾聴力、共感力、心理知識、倫理観
- ・保育士 → 子供対応力、遊び企画、安全管理、保護者対応
- ・福祉士 → 福祉知識、相談力、記録管理、忍耐力

### ビジネス・企画系

- ・インフルエンサー → SNS運用、発信力、ブランディング、分析力
- ・起業家 → ビジネス企画、資金調達、リーダーシップ、柔軟な対応力
- ・ベンチャー企業経営者 → 経営力、資金管理、チームマネジメント、企画力
- ・プロジェクトマネージャー → 計画力、進行管理、リーダーシップ、課題解決力
- ・マーケター → 市場分析、企画力、広告戦略、データ分析

### 公共・作業系

- ・公務員 → 法律知識、事務処理力、協調性、安定志向
- ・軽作業 → 作業スピード、正確性、忍耐力、体力
- ・農家 → 農業知識、体力、季節管理、計画力
- ・漁師 → 漁業知識、体力、チームワーク、危機対応力
- ・林業作業員 → 森林知識、体力、危険管理、環境保護意識
- ・地域おこし協力隊 → 地域理解、企画力、コミュニケーション、柔軟さ
- ・環境保護活動家 → 環境知識、行動力、広報力、情熱

### スポーツ・アウトドア系

- ・トレーナー → 体力、トレーニング知識、モチベーション管理、指導力
- ・シェフ → 調理技術、創作力、衛生管理、チームワーク
- ・アウトドアインストラクター → アウトドア技術、安全管理、指導力、環境知識
- ・スポーツトレーナー → スポーツ知識、体力、リハビリ知識、メンタルサポート力
- ・登山ガイド → 登山技術、安全管理、応急処置力、説明力

