



# 主要プログラミング言語 22選まとめ

by セイト先生のWeb/ITエンジニア転職ラボ

# 01. Python

## オススメ度

初心者：★★★★

経験者：★★★★

## 主な利用シーン

Webのバックエンド開発、データ分析、AI分野（機械学習など）

## 特徴

1. 幅広い層から人気の言語。
2. Web開発のフレームワークやデータ分析やAI系のライブラリが豊富で幅広い開発が可能。
3. 直感的に書ける言語で、初心者しやすい。
4. 国内外問わず求人数は増えている（ただし分野がWeb開発とAI分野で大きく違うので要注意）。
5. とくに海外では教育現場でも近年採用される傾向にある。



# 02. Ruby

## オススメ度

初心者: ★★★

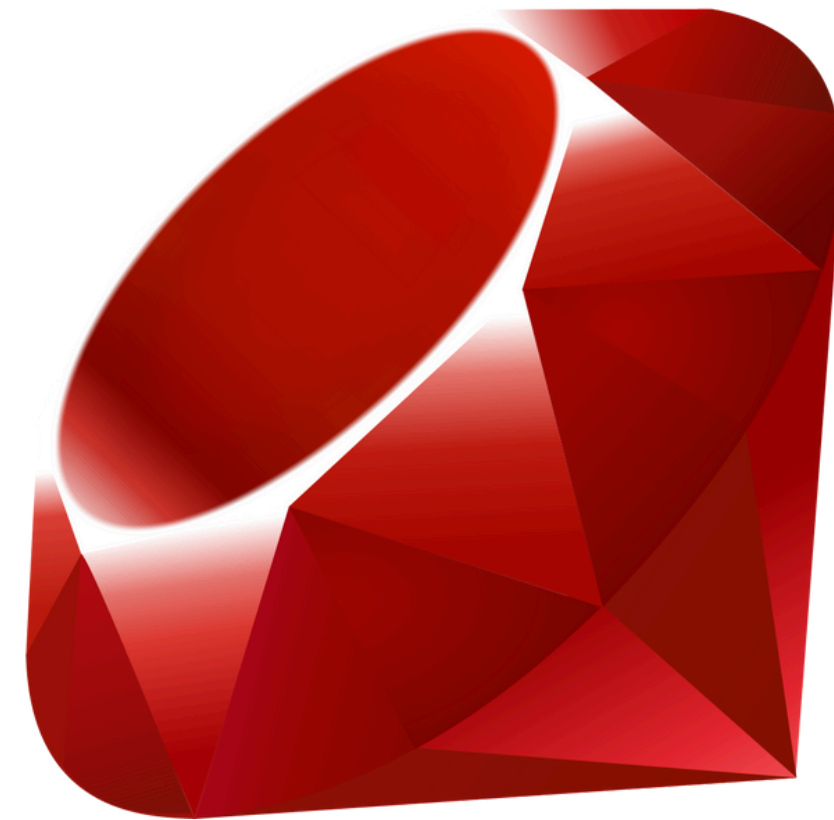
経験者: ★★☆☆

## 主な利用シーン

Webのバックエンド開発

## 特徴

1. Webアプリケーションの開発に適している。
2. フレームワークはRuby on Rails一択。
3. 国内外のスタートアップで採用されやすく、開発スピードが早い。
4. 国内の求人数は少ない。海外の求人は少ないが高単価。
5. 平坦なトレンド。
6. 未経験から入れる求人が比較的多い。



# 03. PHP

## オススメ度

初心者: ★★★

経験者: ★★☆☆

## 主な利用シーン

Webのバックエンド開発

## 特徴

1. Webアプリケーションのバックエンド開発に多用されている。
2. WordPressなどのCMSでも広く使われることから、Webサイト制作にもよく使われる。
3. 一時ダウントrendだったが、Laravelの登場によりぶり返した。
4. 国内の求人数は多い（TOP10内）。
5. 平坦なトレンド。
6. PHPかRubyはどちらかできれば良い。
7. 未経験から入れる求人が比較的多い。



# 04. JavaScript

## オススメ度

初心者：★★★★

経験者：★★★★

## 主な利用シーン

Webのフロントエンド開発、Webのバックエンド開発、スマホアプリ開発、  
デスクトップアプリ開発、GASによるSpreadSheet拡張、Chrome拡張機能開発

## 特徴

1. Webフロントエンド開発に不可欠。
2. 開発範囲が広い。Node.jsを使用すればサーバーサイドプログラミングも可能。ReactNativeでのスマホアプリ開発やElectronでのデスクトップアプリ開発も行える。
3. 始めるのが最も敷居が低い言語。
4. どのランキングでも大体上位。
5. フロントエンドエンジニアやるなら必須。



05.

# TypeScript

## オススメ度

初心者：★★★★

経験者：★★★★

## 主な利用シーン

JavaScriptと同様（JavaScriptにできることすべて）

## 特徴

1. JavaScriptに静的型付け言語の特徴をもたせた拡張言語。
2. 近年急速に需要が爆上がりしている。
3. 静的型付け言語自体がトレンド。



# 06. HTML

## オススメ度

初心者：★★★★

経験者：★★★★

## 主な利用シーン

Webページの作成（構造）

## 特徴

- 1.Webページの構造を定義。プログラミング言語ではないが、Web開発の基本。
- 2.フロントエンドエンジニアやるなら必須。



# 07. CSS

## オススメ度

初心者：★★☆

経験者：★★★★

## 主な利用シーン

Webページの作成（装飾・アニメーション）

## 特徴

1. Webページのスタイルを定義。プログラミング言語ではないが、Web開発の基本。
2. 近年仕様が増えてきた。HTMLより難しい。
3. フロントエンドエンジニアやるなら必須。





08.

# Visual Basic



## オススメ度

初心者: ★★☆☆

経験者: ★☆☆☆

## 主な利用シーン

Webシステムの開発、Windows製品上での機能拡張

## 特徴

- 1.主に教育目的や小規模なビジネスアプリケーションで使われる。
- 2.エクセルの延長ではじめに学習する人が多い。
- 3.案件事態は少ない。ダウントレンド。



Visual Basic



09.

# Shell Script



## オススメ度

初心者: ★☆☆

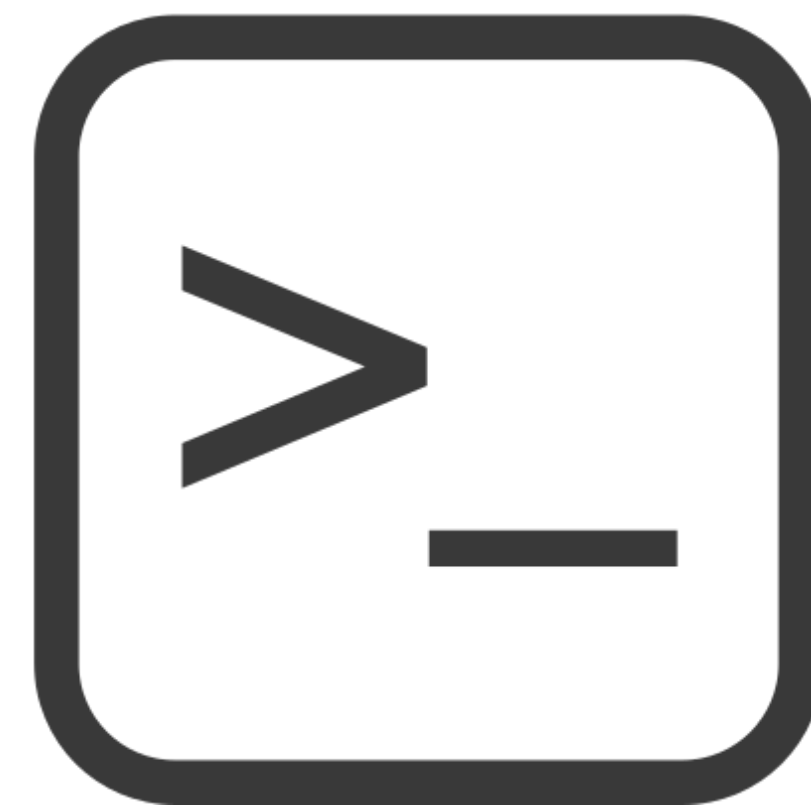
経験者: ★★☆☆

## 主な利用シーン

PCに向けての命令、簡単な自動化処理

## 特徴

1. コマンドの集合体。
2. システム管理タスクの自動化や簡単なスクリプト作成に用いられる（例えばPCの起動時に毎回決まった処理を行う、など）。
3. よく使う簡単な処理を再利用したい、自動化したい、ときに使う。



# 10.

# C言語

## オススメ度

初心者：★☆☆

経験者：★★☆

## 主な利用シーン

Webやスマホのバックエンド開発、デスクトップアプリ、ハードウェア製品におけるプログラム(車、家電、ロボット)、OS、など

## 特徴

- 1.近代プログラミング言語の祖。歴史が古い。
- 2.ポインタという仕様を用いてメモリを管理する必要がある。
- 3.Cを直接使う求人は国内外多い（TOP10位に入る）。
- 4.とにかく守備範囲が広い。



# 11.

# C++

## オススメ度

初心者：★★☆

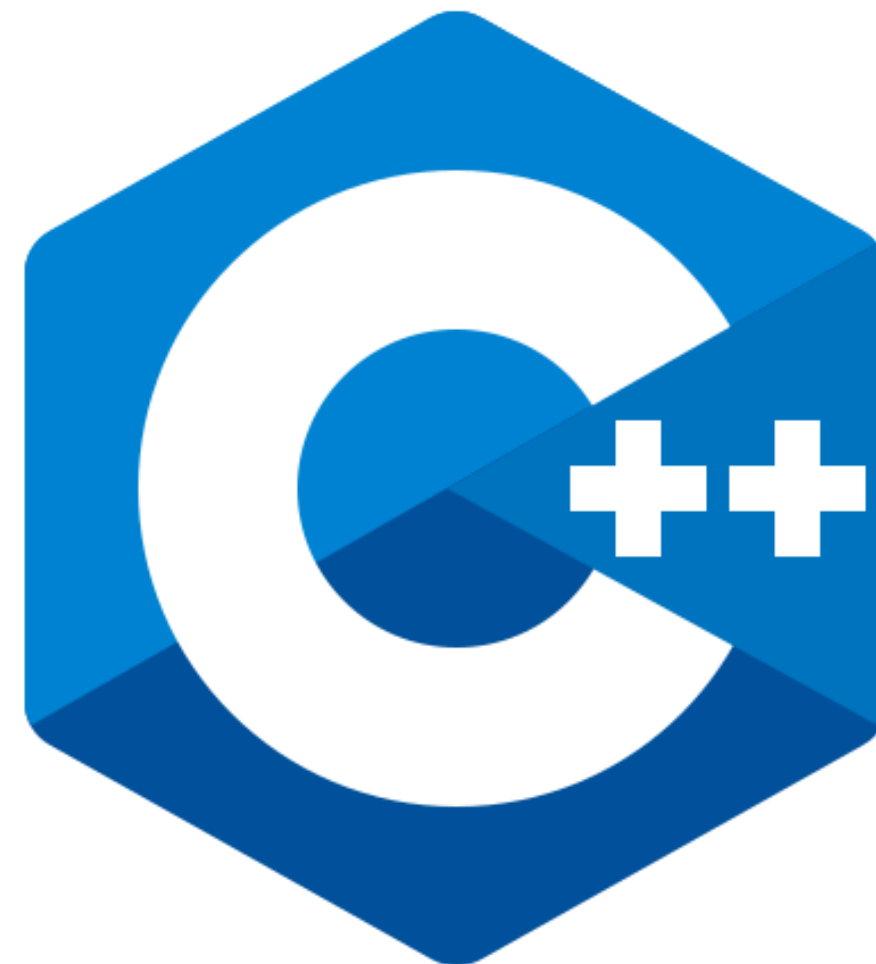
経験者：★★☆

## 主な利用シーン

ハードウェア製品におけるプログラム(車、家電、ロボット)、Webやスマホのバックエンド開発、デスクトップアプリ、ゲーム、XRアプリ

## 特徴

1. 高速に処理が可能。
2. ゲーム開発においてはUnreal Engineを使う場合C++一択。
3. スマートポインタという機能を使ってメモリ管理を行う仕様をもつが、難しい。
4. C++を直接使う求人は国内外多い (TOP10位に入る)。



# 12.

# C#

## オススメ度

初心者：★★☆

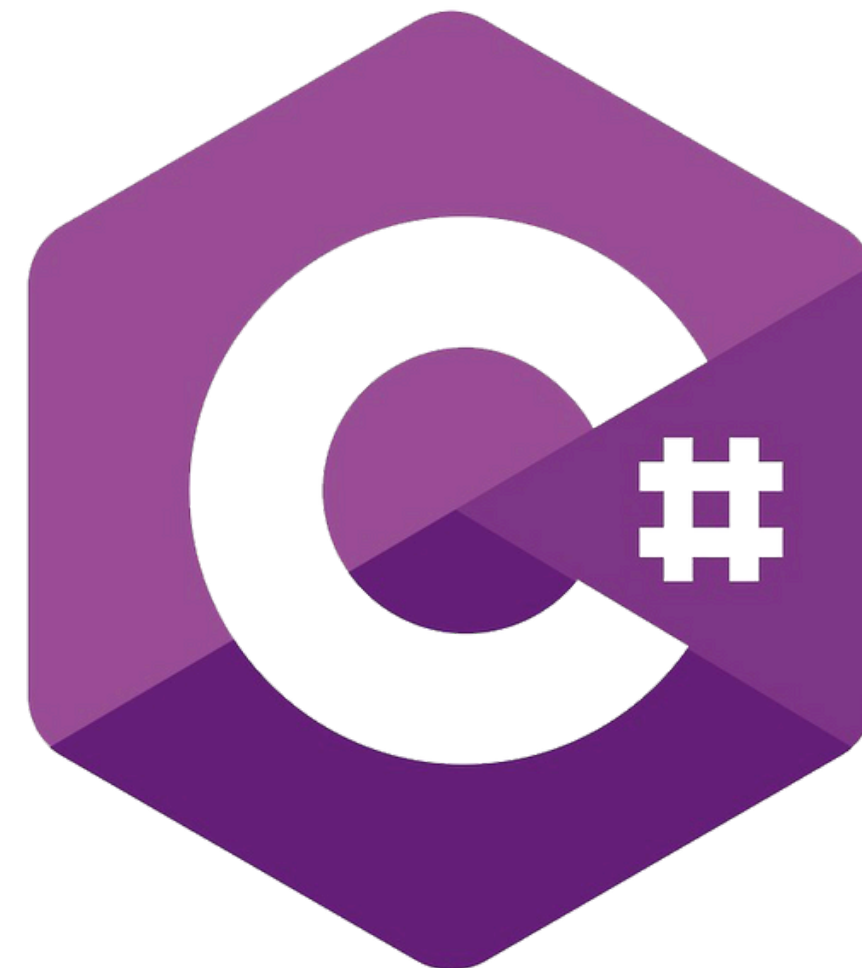
経験者：★★☆

## 主な利用シーン

ゲーム開発（Unity）、Webアプリケーション開発、基幹システム、デスクトップアプリ開発、スマホアプリ開発（iOS, Android）、XRアプリ

## 特徴

- 1.Unityでのゲーム開発では実質一択。
- 2.Webでは.NETフレームワークと組み合わせて使用され、特にWindowsプラットフォームでのアプリ開発に強いがこちらはダウントレンド。
- 3.ゲーム開発したい初心者・経験者にはおすすめだが、Web開発を目指す人にはおすすめしない。
- 4.スマホアプリはXamarinを使って開発できるがマイナー。



# 13. Java

## オススメ度

初心者：★★☆

経験者：★★★★

## 主な利用シーン

Webのバックエンド、スマホアプリのバックエンド・フロントエンド、スマホアプリのバックエンド、PC向けデスクトップアプリ開発、基幹システム、組み込みシステム、IoT

## 特徴

1. エンタープライズレベルのバックエンドシステムからAndroidアプリまで広範囲にわたって使用される。
2. 圧倒的な仕事の数。
3. 初学者にとってはPythonやRubyなどに比べて難しい。
4. 大企業での採用ケースが多いので、キャリアアップするために習得するのはアリ。



# 14. Kotlin



## オススメ度

初心者：★★☆

経験者：★★★★

## 主な利用シーン

Webのバックエンド、スマホアプリのバックエンド・フロントエンド、スマホアプリのバックエンド、PC向けデスクトップアプリ開発、基幹システム、組み込みシステム、IoT

## 特徴

1. JVM言語の1つ。Javaエンジニアは学習しやすい。
2. バックエンド実装でも使われるが、Androidアプリ開発においてJavaの代替として推奨されている。
3. 2017年といえばGoogleにより、「KotlinをAndroidの公式開発言語とする」と発表された年で、これをきっかけに注目されている。



# 15. Swift

## オススメ度

初心者：★★★★

経験者：★★★★

## 主な利用シーン

スマホアプリ（iOS）、タブレットアプリ（iOS）、Apple VisionでのXRアプリ

## 特徴

1. Appleのプラットフォームに特化した言語。
2. 静的言語の特徴を持ちながら初心者も比較的とっつきやすい。
3. 年収が高い求人が多い。
4. Appleが潰れない限り安泰。
5. Apple Visionの登場で盛り上がるかも？





# 16. Dart

## オススメ度

初心者：★☆☆

経験者：★★★★

## 主な利用シーン

スマホアプリ (iOS, Android)

## 特徴

- 1.Flutterと組み合わせてクロスプラットフォームモバイルアプリを開発。
- 2.Swift, Java, Kotlin, React経験者はとっつきやすい。
- 3.新規事業開発やスタートアップでよく用いられる。
- 4.ReactNativeと競合。



# 17. Scala

## オススメ度

初心者：★☆☆

経験者：★☆☆

## 主な利用シーン

Web、スマホアプリのバックエンド、PC向けデスクトップアプリ開発、基幹システム、組み込みシステム、IoT

## 特徴

1. JVM言語の1つ。Javaエンジニアは学習しやすい。
2. Better Javaと呼ばれ、Javaの資産を流用しつつJavaよりも少ないコード量で高いパフォーマンスを出せる。
3. オブジェクト指向言語と関数型言語両方の特徴を持つ。
4. 強い静的型付け。
5. 求人の年収は高いがダウントレンド。



# 18. Rust

## オススメ度

初心者：★☆☆

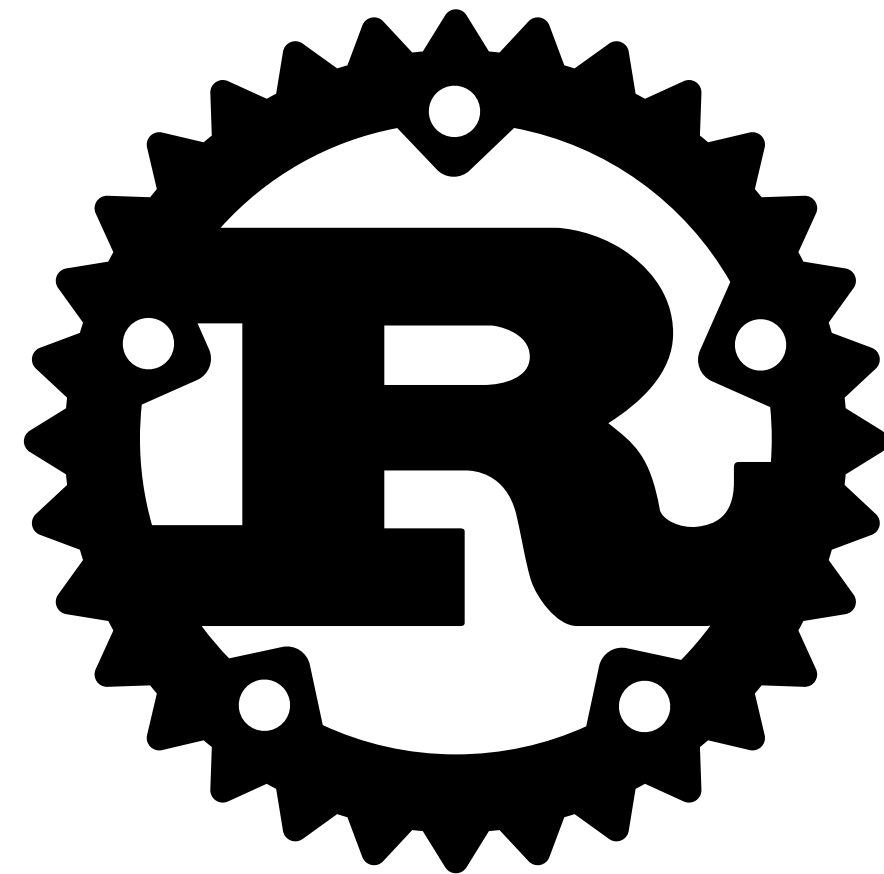
経験者：★★★★

## 主な利用シーン

Webやスマホのバックエンド開発、デスクトップアプリ、ハードウェア製品におけるプログラム(車、家電、ロボット)、OS、など

## 特徴

1. プログラマー人気が圧倒的に高い。
2. 高速で実行できる、ルールが厳格、エラーメッセージがわかりやすい。
3. まだニーズは一部だが将来性が非常に高い。C言語の役割を担っていく可能性。



# 19. Elixir

## オススメ度

初心者：★☆☆

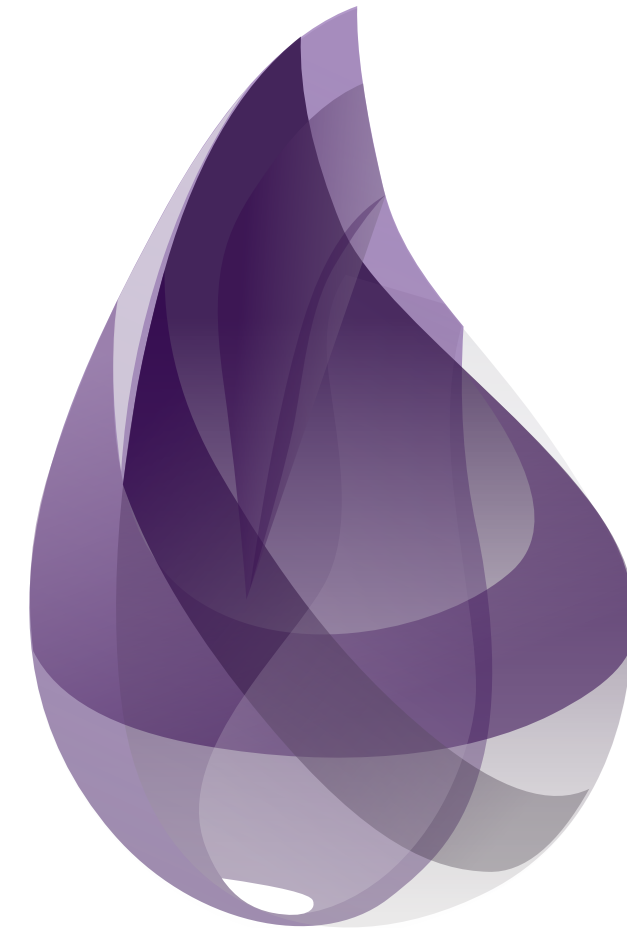
経験者：★★☆

## 主な利用シーン

Webやスマホのバックエンド開発、ハードウェア製品におけるプログラム(車、家電、ロボット)

## 特徴

1. 関数型言語。
2. 文法が分かり易い、Rubyに似ている。
3. 大量の並行処理を安定的に実行できる、拡張性が高い。
4. プログラマー人気が高く、Phoenix Frameworkの支持率が高い。
5. 海外での求人の年収は高い。
6. 国内の求人ではあまり見かけない。



# 20.

# Go言語

## オススメ度

初心者：★☆☆

経験者：★★★★

## 主な利用シーン

Web・スマホアプリのバックエンド開発、組み込みシステム、IoT、基幹システム

## 特徴

1. 並行処理やネットワークサーバーの開発に優れ、シンプルな構文を持つ。
2. 求人数が特段大きくはないが上昇トレンドで、単価も高い。有名企業や最規模サービスでの採用が多い。



21.

# R言語

## オススメ度

初心者：★☆☆

経験者：★☆☆

## 主な利用シーン

データ分析、グラフ、アルゴリズム

## 特徴

- 1.統計分析や機械学習を行うための言語で研究者向け。
- 2.他の言語やWebなどのプラットフォームとの互換性が弱い。
- 3.Pythonにはない研究者向けライブラリがあったりする。



# 22. Solidity

## オススメ度

初心者：★☆☆

経験者：★★☆

## 主な利用シーン

スマートコントラクト、NFT、DApps

## 特徴

1. ブロックチェーンの分野で用いられる言語。
2. スマートコントラクト、イーサリアム、Dapps、などの文脈でよく登場する。
3. ブロックチェーン関連の開発に使われる言語としてはSolidityの一強。
4. 業界が限られるので求人数は多くはないが単価が高い。
5. できる人が少ないので経験年数が少なくても参画できる機会は少なくない。
6. 学習するなら <https://cryptozombies.io/> がおすすめ。





# 参考文献

Stack Overflow Developer Survey 2023  
<https://survey.stackoverflow.co/2023/>

GitHub OCTOVERSE 2023  
<https://octoverse.github.com/>