

共通

Gitリポジトリ

学習用のGitリポジトリを`git clone`してください。

学習用のリポジトリは受講者ごとで、サブスクリプション入会後に作成されます。入会前に環境構築を試したい場合は下記のリポジトリで実施してください。

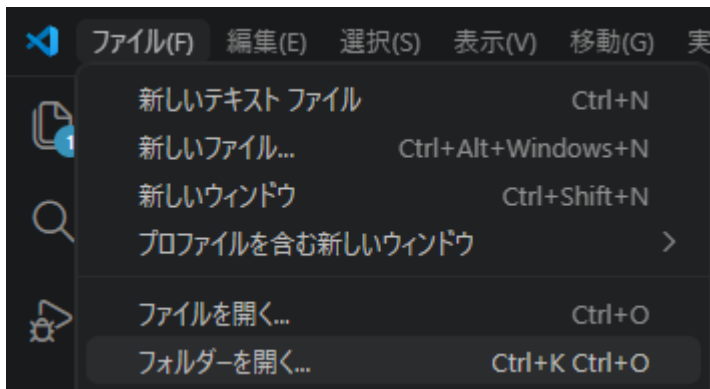
- JavaScript: <https://github.com/code-migaki/template-javascript>
- TypeScript: <https://github.com/code-migaki/template-typescript>
- Python: <https://github.com/code-migaki/template-python>
- Java: <https://github.com/code-migaki/template-java>
- Go: <https://github.com/code-migaki/template-go>

Visual Studio Code

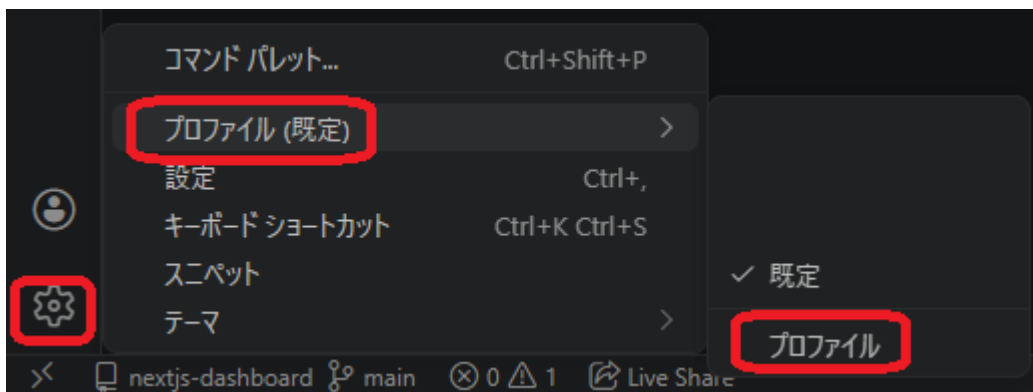
Visual Studio Codeをインストールしてください。

<https://code.visualstudio.com/>

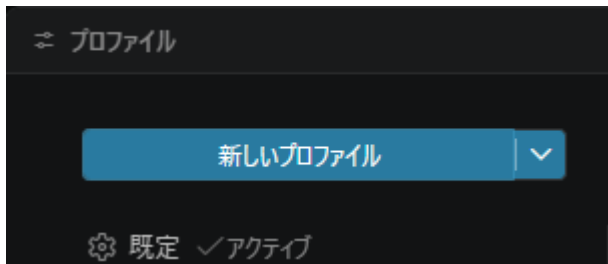
`clone`したディレクトリをVisual Studio Codeで開いてください。



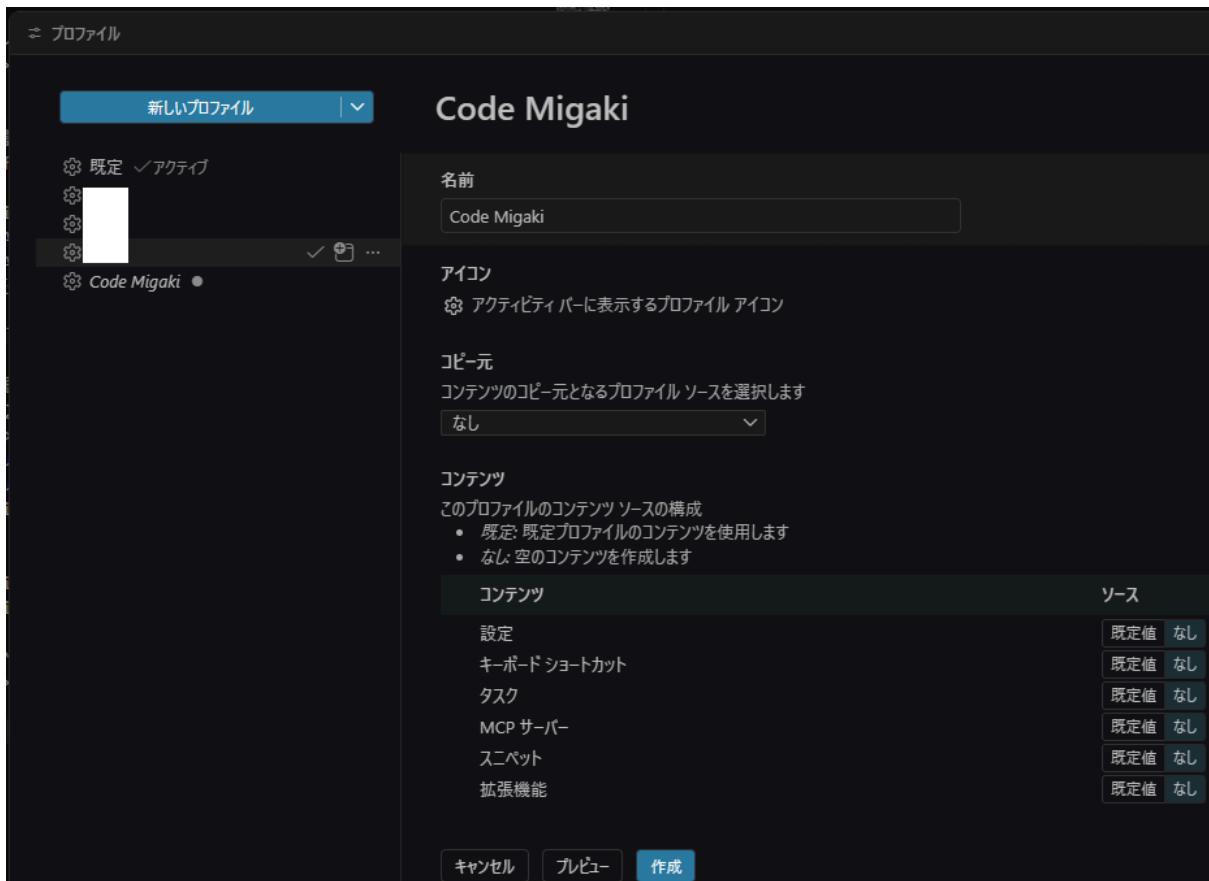
VS Codeでプロファイル画面を開いてください。



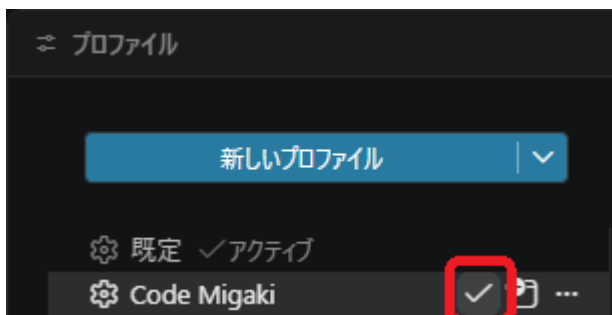
[新しいプロファイル]を押してください。



任意の名前を指定してください。その他の項目は全て[なし]を指定して[作成]を押してください。



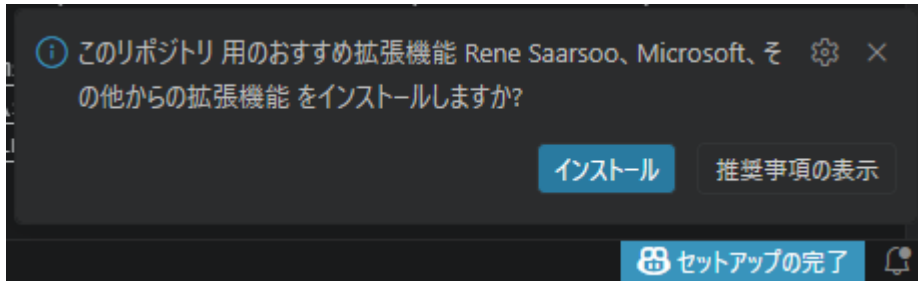
そのプロフィールをアクティブにしてください。



次回から、Visual Studio Codeで学習用のディレクトリを開くと、自動的にそのプロフィールがアクティブになります。

プロフィール画面を閉じてください。

Visual Studio Codeの右下に、おすすめ拡張機能のインストールを促すポップアップが表示されます。画面に従ってインストールしてください。なお、対象の拡張機能は学習する言語により異なります。



おすすめの拡張機能以外はインストールしないでください。

学習効果を最大化するため、AI機能(GitHub Copilot)は使用不可の設定になっています。

JavaScript

環境構築

Node.js 24 をインストールしてください。

<https://nodejs.org/ja/download>

Visual Studio Codeで学習用ディレクトリを開いてください。

`npm install`を実行してください。

`npm run start`を実行してください。

プログラムが実行されることを確認してください。ターミナル上で適当な入力を行い、動作を確認してください。

```
$ npm run start
1つめの数は? 3
2つめの数は? 5
合計: 8
```

課題を開始するには

`src/practice0`ディレクトリを`practice1`ディレクトリなどにコピーしてください。

`src/main.js`に記述されている`practice0`を`practice1`などに変更してください。

コピー先ディレクトリ内に課題の内容を実装してください。

TypeScript

環境構築

Node.js 24 をインストールしてください。

<https://nodejs.org/ja/download>

Visual Studio Codeで学習用ディレクトリを開いてください。

`npm install`を実行してください。

`npm run start`を実行してください。

プログラムが実行されることを確認してください。ターミナル上で適当な入力を行い、動作を確認してください。

```
● $ npm run start
  > code-migaki@1.0.0 start
  > tsx src/main.ts

1つめの数は? 3
2つめの数は? 5
合計: 8
```

課題を開始するには

`src/practice0`ディレクトリを`practice1`ディレクトリなどにコピーしてください。

`src/main.js`に記述されている`practice0`を`practice1`などに変更してください。

コピー先ディレクトリ内に課題の内容を実装してください。

Python

環境構築

uvをインストールしてください。

<https://docs.astral.sh/uv/getting-started/installation/>

Python 3.14 をインストールしてください。

<https://www.python.org/downloads/>

Visual Studio Codeで学習用ディレクトリを開いてください。

`uv sync`を実行してください。

ターミナルを再起動してください。

`src`ディレクトリで`python -m main`を実行してください。

プログラムが実行されることを確認してください。ターミナル上で適当な入力を行い、動作を確認してください。

```
● $ python -m main
1つめの数は?3
2つめの数は?5
合計:8
```

課題を開始するには

`src/practice0`ディレクトリを`practice1`ディレクトリなどにコピーしてください。

`src/main.py`に記述されている`practice0`を`practice1`などに変更してください。

コピー先ディレクトリ内に課題の内容を実装してください。

Java

環境構築

Java 25をインストールしてください。

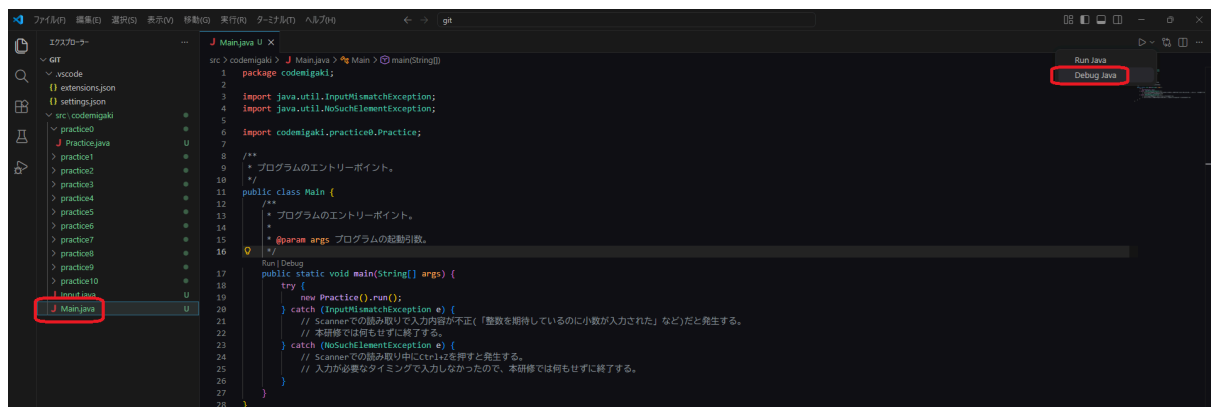
- Windows用: <https://www.oracle.com/jp/java/technologies/downloads/#jdk25-windows>
- Mac用: <https://www.oracle.com/jp/java/technologies/downloads/#jdk25-mac>

Javaのインストール先ディレクトリを環境変数JAVA_HOMEに指定してください。

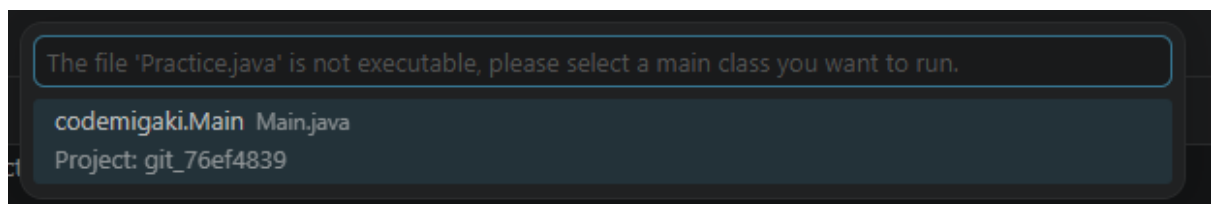
Visual Studio Codeで学習用ディレクトリを開いてください。

src/codemigaki/Main.javaを開いてください。

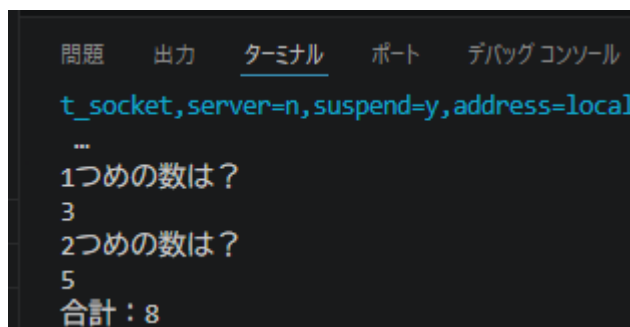
Debug Javaを選択してください。



提示された内容を選択してください。



プログラムが実行されることを確認してください。ターミナル上で適当な入力を行い、動作を確認してください。



課題を開始するには

`src/codemigaki/practice0`ディレクトリを`practice1`ディレクトリなどにコピーしてください。

コピー先ディレクトリにある`Practice.java`内に記述されている`practice0`を`practice1`などに変更してください。

`src/codemigaki/Main.java`に記述されている`practice0`を`practice1`などに変更してください。

コピー先ディレクトリ内に課題の内容を実装してください。

Go

環境構築

Go 1.26 をインストールしてください。

<https://go.dev/dl/>

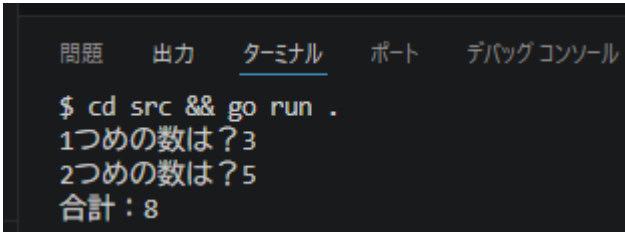
Visual Studio Codeで学習用ディレクトリを開いてください。

`src\main.go`を開いてください。

goplsのインストールを促す画面が表示されたら、指示に従ってインストールしてください。

`src`ディレクトリで`go run .`を実行してください。

プログラムが実行されることを確認してください。ターミナル上で適当な入力を行い、動作を確認してください。



```
問題 出力 ターミナル ポート デバッグ コンソール
$ cd src && go run .
1つめの数は?3
2つめの数は?5
合計:8
```

課題を開始するには

`src/practice0`ディレクトリを`practice1`ディレクトリなどにコピーしてください。

コピー先ディレクトリにある`practice.go`内に記述されている`practice0`を`practice1`などに変更してください。

`src/main.go`に記述されている`practice0`を`practice1`などに変更してください。

コピー先ディレクトリ内に課題の内容を実装してください。