

1. 目的

- `dotfiles` リポジトリは macOS 開発環境を新しい Mac でもすぐ再現するための設定集とする。
- VS Code / Karabiner など、実際に動作に効く設定ファイルの「正本（SSOT）」をここで管理する。
- 学習メモやベストプラクティスのまとめは `Summary/` 側に置き、`dotfiles` には置かない。

2. リポジトリ名・場所

- リポジトリ名: `dotfiles`
- ローカルの配置:
 - ルート: `$HOME/Work/dotfiles/`
- GitHub とは SSH で接続し、Private リポジトリとして運用する。

3. フォルダ構成ルール（基本）

`$HOME/Work/dotfiles/` の中はアプリ単位でサブフォルダを切る：

```
$HOME/Work/dotfiles/
├── README.md          # dotfiles 全体の説明・セットアップ手順
└── vscode/            # VS Code 関連
    └── karabiner/      # Karabiner 関連 (将来 zsh / git 等も追加可)
```

各アプリに固有の詳細な説明を追加したい場合は、

`vscode/README.md`、`karabiner/README.md` を追加して良い。

4. VS Code 設定ファイルの扱い

4-1. 管理対象ファイル

`$HOME/Work/dotfiles/vscode/` には、以下を「正本」として保存する：

```
vscode/
├── settings.json       # VS Code 全体の設定
├── keybindings.json    # キーバインド設定
├── tasks.json           # 共通タスク (Node 実行 など)
└── snippets/            # 自作 snippet がある場合はフォルダごと
    └── extensions.txt   # 拡張機能の一覧 (後述)
```

4-2. 実ファイルとの対応

- 実際に VS Code が参照するファイル：

```
$HOME/Library/Application Support/Code/User/settings.json  
$HOME/Library/Application Support/Code/User/keybindings.json  
$HOME/Library/Application Support/Code/User/tasks.json  
$HOME/Library/Application Support/Code/User/snippets/
```

- 運用イメージ：

- dotfiles/vscode/ 側を編集 → 必要に応じて上記パスへ コピー or シンボリックリンク する。
- 新PCでも dotfiles を clone したあと、同様にコピー／リンクして環境を再現する。

4-3. tasks.json の前提

- tasks.json の command は 絶対パスではなく node を使う：

```
{  
  "version": "2.0.0",  
  "tasks": [  
    {  
      "label": "Run Node.js File",  
      "type": "shell",  
      "command": "node",  
      "args": ["${file}"],  
      "group": {  
        "kind": "build",  
        "isDefault": true  
      },  
      "presentation": {  
        "echo": true,  
        "reveal": "always",  
        "focus": false,  
        "panel": "shared"  
      },  
      "problemMatcher": []  
    }  
  ]  
}
```

- 前提： node コマンドが PATH に入っていること
(現在は Volta を利用しているが、将来別の方法でインストールしてもよい)

5. 拡張機能の管理 (extensions.txt)

- インストール済み拡張機能の一覧は `extensions.txt` として管理する：

```
# 現在の環境から一覧を生成
code --list-extensions > $HOME/Work/dotfiles/vscode/extensions.txt
```

- 新PCで同じ拡張をまとめて入れる手順：

```
cd $HOME/Work/dotfiles/vscode
cat extensions.txt | xargs -n 1 code --install-extension
```

6. Karabiner 設定の扱い

- 正本の保存場所：

```
$HOME/Work/dotfiles/karabiner/karabiner.json
```

- 実際に Karabiner が参照する場所：

```
$HOME/.config/karabiner/karabiner.json
```

- 運用：

- `dotfiles/karabiner/karabiner.json` を編集し、必要に応じて `$HOME/.config/karabiner/` にコピー or シンボリックリンクする。
- 将来 `complex_mods` などを追加する場合も、`dotfiles/karabiner/` 配下で管理する。

7. Summary フォルダとの役割分担

- dotfiles (このリポジトリ)：
 - 実際にアプリが読み込む設定ファイルの「正本」

- 新PCで環境を再現するためのコマンドや手順
- Summary (例：`$HOME/Documents/Summary/Editor/Vscode/`) :
 - VS Code / Karabiner 設定の考え方・ベストプラクティス・メモ を置く場所
 - 例：
 - 「キーバインド設計の方針」
 - 「おすすめ拡張機能の解説」
 - 「settings.json の重要な項目の説明」

ルール：

実際の設定ファイルそのものは `dotfiles/` に置く。
設定に関するナレッジ・説明は `Summary/` に置く。

8. 新しい Mac での大まかな手順（概要）

1. Homebrew / Volta / VS Code / Karabiner など必要なツールをインストールする。

2. `dotfiles` リポジトリを clone :

```
cd $HOME/Work  
git clone git@github.com:<ユーザー名>/dotfiles.git
```

3. VS Code 設定をコピー or シンボリックリンクする。

4. `extensions.txt` を使って拡張機能をまとめてインストールする。

5. Karabiner の設定を `$HOME/.config/karabiner/` に反映する。

（詳細なコマンドは `README.md` 側に記述する）

9. 運用ルール

- 設定を変更するときは、原則 `dotfiles` 側のファイルを編集して commit → push する。

- 直接 `~/Library/.../User/` のファイルを編集した場合は、後で必ず `dotfiles/vscode/` へ反映する。
- 大きな変更をする前には、`git status` / `git commit` をこまめに行い、履歴を残す。
- このリポジトリは 個人環境用 のため、当面は Private のまま運用する。