

現場で使うVS CODE

2020年12月18日(金)



今日話すこと



今日話すこと

- 拡張機能については話しません。

今日話すこと

- 拡張機能については話しません。
- Windows環境を想定してます。

今日話すこと


- 拡張機能については話しません。
- Windows環境を想定してます。
- 特定範囲の人にとってはあるあるだと思います。

自己紹介



@kabukawa

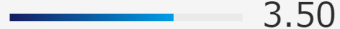
Kunihisa Abukawa

 <https://github.com/kabukawa>

 <https://lapras.com/public/YHXMSK7>

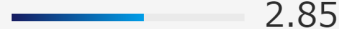
 <http://b.hatena.ne.jp/kabukawa/bookmark>

技術力



3.50

ビジネス力



2.85

影響力



3.76



<https://lapras.com/public/YHXMSK7>



L A P R A S



VS CODE使ってますか



VS CODE使ってますか

- 拡張機能便利ですね



VS CODE使ってますか

- 拡張機能便利ですね
- 様々な機能が追加できて、色々な目的で使える

VS CODE使ってますか

- 拡張機能便利ですね
- 様々な機能が追加できて、色々な目的で使える
- 単なるエディタというより環境

VS CODE使ってますか

- 拡張機能便利ですね
- 様々な機能が追加できて、色々な目的で使える
- 単なるエディタというより環境
- ナウい

VS CODE使ってますか

- 拡張機能便利ですね
- 様々な機能が追加できて、色々な目的で使える
- 単なるエディタというより環境
- ナウい

だがしかし



だがしかし

- 客先で開発をしていると、VS Code入れられない場合もある

だがしかし

- 客先で開発をしていると、VS Code入れられない場合もある
- 拡張機能を入れられない場合もある

だがしかし

- 客先で開発をしていると、VS Code入れられない場合もある
- 拡張機能を入れられない場合もある
 - 勝手にインストール禁止

だがしかし

- 客先で開発をしていると、VS Code入れられない場合もある
- 拡張機能を入れられない場合もある
 - 勝手にインストール禁止
 - そもそもネットに繋がっていない

だがしかし

- 客先で開発をしていると、VS Code入れられない場合もある
- 拡張機能を入れられない場合もある
 - 勝手にインストール禁止
 - そもそもネットに繋がっていない



VS CODEの無い世界



VS CODEの無い世界

- せめて、VS Codeだけでも入れたいんじゃ！

VS CODEの無い世界

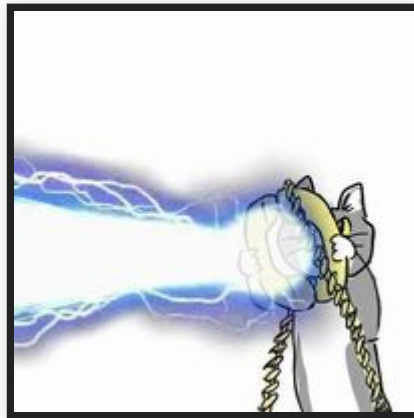
- せめて、VS Codeだけでも入れたいんじゃ！
- 拡張機能は、、、入れたいけど無理。

VS CODEの無い世界

- せめて、VS Codeだけでも入れたいんじゃない！
- 拡張機能は、、、入れたいけど無理。
- バージョン管理のために git も入れたいんじゃない！

VS CODEの無い世界

- せめて、VS Codeだけでも入れたいんじゃない！
- 拡張機能は、、、入れたいけど無理。
- バージョン管理のために git も入れたいんじゃない！



現場で説得



現場で説得

- お金かかりません。エディタです。

現場で説得

- お金かかりません。エディタです。
 - マイクロソフト製なの？じゃあいつか。

現場で説得

- お金かかりません。エディタです。
 - マイクロソフト製なの？じゃあいつか。
- Gitも必要なんっすよね。

現場で説得

- お金かかりません。エディタです。
 - マイクロソフト製なの？じゃあいつか。
- Gitも必要なんっすよね。
 - VS Code入れるとgitも必要になるので。

現場で説得

- お金かかりません。エディタです。
 - マイクロソフト製なの？じゃあいつか。
- Gitも必要なんっすよね。
 - VS Code入れるとgitも必要になるので。



拡張を入れなくてもできること



拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)
- 画面分割、タブ切り替えによる複数ファイル編集

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)
- 画面分割、タブ切り替えによる複数ファイル編集
- ファイル内検索/grep

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)
- 画面分割、タブ切り替えによる複数ファイル編集
- ファイル内検索/grep
- 統合ターミナル

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)
- 画面分割、タブ切り替えによる複数ファイル編集
- ファイル内検索/grep
- 統合ターミナル
- Markdownのプレビュー

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)
- 画面分割、タブ切り替えによる複数ファイル編集
- ファイル内検索/grep
- 統合ターミナル
- Markdownのプレビュー
- Javascriptのデバッグ

拡張を入れなくてもできること

- ファイルの新規作成/更新/削除/名前変更など
- Gitによるバージョン管理(要gitコマンド)
- 高機能な編集機能(マルチカーソル等も使える)
- 画面分割、タブ切り替えによる複数ファイル編集
- ファイル内検索/grep
- 統合ターミナル
- Markdownのプレビュー
- Javascriptのデバッグ

などなど

あれ？



あれ？

結構素のVS Codeでも色々できるじゃん。



あれ？

結構素のVS Codeでも色々できるじゃん。

- 色々不自由のある現場でも活かせる

あれ？

結構素のVS Codeでも色々できるじゃん。

- 色々不自由のある現場でも活かせる
- 単純に便利

あれ？

結構素のVS Codeでも色々できるじゃん。

- 色々不自由のある現場でも活かせる
- 単純に便利
- これって最高では？

あれ？

結構素のVS Codeでも色々できるじゃん。

- 色々不自由のある現場でも活かせる
- 単純に便利
- これって最高では？



実際に使っている設定



実際に使っている設定

デフォルトから設定を変えているものを紹介。

実際に使っている設定

デフォルトから設定を変えているものを紹介。

- [設定] 画面の開き方

実際に使っている設定

デフォルトから設定を変えているものを紹介。

- [設定] 画面の開き方
 - Ctrl + ,のショートカット

実際に使っている設定

デフォルトから設定を変えているものを紹介。

- [設定] 画面の開き方
 - Ctrl + ,のショートカット
 - File > Preferences > Settings

実際に使っている設定

デフォルトから設定を変えているものを紹介。

- [設定] 画面の開き方
 - Ctrl + , のショートカット
 - File > Preferences > Settings
 - サイドメニューの歯車アイコンから「設定」を選択

文字コード自動判定



文字コード自動判定

- VS Codeのデフォルトは utf-8。

文字コード自動判定

- VS Codeのデフォルトは utf-8。
- だけどWindowsではShift-JISのファイルもあるので文字化けせずに編集したい。

文字コード自動判定

- VS Codeのデフォルトは utf-8。
 - だけどWindowsではShift-JISのファイルもあるので文字化けせずに編集したい。
1. 検索欄に `files.autoGuessEncoding` を打ち込み、チェックボックスを ON にする。

文字コード自動判定

- VS Codeのデフォルトは utf-8。
 - だけどWindowsではShift-JISのファイルもあるので文字化けせずに編集したい。
1. 検索欄に `files.autoGuessEncoding` を打ち込み、チェックボックスを ON にする。
 2. もしくは `settings.json` で以下の項目を記述する。



文字コード自動判定

- VS Codeのデフォルトは utf-8。
 - だけどWindowsではShift-JISのファイルもあるので文字化けせずに編集したい。
1. 検索欄に `files.autoGuessEncoding` を打ち込み、チェックボックスを ON にする。
 2. もしくは `settings.json` で以下の項目を記述する。

```
"files.autoGuessEncoding": true,
```

自動判定だけど



自動判定だけど

- うまく判定できないこともある。

自動判定だけど

- うまく判定できないこともある。
- JIS(iso-2022-jp)は変換できない。

自動判定だけど

- うまく判定できないこともある。
- JIS(iso-2022-jp)は変換できない。
 - 今どきJISのファイルなんて無いでしょ？
(いや、有るんですよ。。。)

自動判定だけど

- うまく判定できないこともある。
- JIS(iso-2022-jp)は変換できない。
 - 今どきJISのファイルなんて無いでしょ？
(いや、有るんですよ。。。)
 - GitBashに付属のvimで！(本末転倒)

自動判定だけど

- うまく判定できないこともある。
- JIS(iso-2022-jp)は変換できない。
 - 今どきJISのファイルなんて無いでしょ？
(いや、有るんですよ。。。)
 - GitBashに付属のvimで！(本末転倒)
 - GitBashに付属のiconv(だ～か～ら～)

自動判定だけど

- うまく判定できないこともある。
- JIS(iso-2022-jp)は変換できない。
 - 今どきJISのファイルなんて無いでしょ？
(いや、有るんですよ。。。)
 - GitBashに付属のvimで！(本末転倒)
 - GitBashに付属のiconv(だ～か～ら～)



改行コード



改行コード

- サーバーとのやり取りもあるので、改行コードは LF に。

改行コード

- サーバーとのやり取りもあるので、改行コードは LF に。
- 1. 検索欄に `files.eol` を打ち込み、ドロップダウンリストで “`\n`” を選択してデフォルトの改行コードを LF に変更する。

改行コード

- サーバーとのやり取りもあるので、改行コードはLFに。
1. 検索欄に `files.eol` を打ち込み、ドロップダウンリストで “`\n`” を選択してデフォルトの改行コードをLFに変更する。
 2. もしくは `settings.json` で以下の項目を記述する。

改行コード

- サーバーとのやり取りもあるので、改行コードはLFに。
1. 検索欄に `files.eol` を打ち込み、ドロップダウンリストで “`\n`” を選択してデフォルトの改行コードをLFに変更する。
 2. もしくは `settings.json` で以下の項目を記述する。

```
"files.eol": "\n",
```

別に改行コードは変えなくても



別に改行コードは変えなくても

- サーバーにファイルを持っていったときに ^M を見たくない。

別に改行コードは変えなくても

- サーバーにファイルを持っていたときに ^M を見たくない。
- 今どき、メモ帳でも 改行コード LF のファイルを扱える。

別に改行コードは変えなくても

- サーバーにファイルを持っていたときに ^M を見たくない。
- 今どき、メモ帳でも 改行コード LF のファイルを扱える。
- だったらどっちかに統一したほうが安心。

別に改行コードは変えなくても

- サーバーにファイルを持っていたときに ^M を見たくない。
- 今どき、メモ帳でも 改行コード LF のファイルを扱える。
- だったらどっちかに統一したほうが安心。



統合シェル



統合シェル

- VS Codeの統合ターミナルをGitBashに。



統合シェル

- VS Codeの統合ターミナルをGitBashに。
- 検索欄に `terminal.integrated.shell.windows` を打ち込み、`settings.json`で編集 で編集画面表示。
以下の内容を記述する。

統合シェル

- VS Codeの統合ターミナルをGitBashに。
- 検索欄に `terminal.integrated.shell.windows` を打ち込み、`settings.json`で編集 で編集画面表示。
以下の内容を記述する。

```
"terminal.integrated.shell.windows": "C:\\Program Files\\Git\\"
```

POWERSHELLじゃないの？



POWERSHELLじゃないの？

- Git for Windowsをインストールすると一緒に入る。

POWERSHELLじゃないの？

- Git for Windowsをインストールすると一緒に入る。
- WSLが使えない32bit Windows環境でbash使いたいなら

POWERSHELLじゃないの？

- Git for Windowsをインストールすると一緒に入る。
- WSLが使えない32bit Windows環境でbash使いたいなら
- WSL環境と設定ファイルを共用できる(俺得)

POWERSHELLじゃないの？

- Git for Windowsをインストールすると一緒に入る。
- WSLが使えない32bit Windows環境でbash使いたいなら
- WSL環境と設定ファイルを共用できる(俺得)
- 俺はbashが使いたい。

POWERSHELLじゃないの？

- Git for Windowsをインストールすると一緒に入る。
- WSLが使えない32bit Windows環境でbash使いたいなら
- WSL環境と設定ファイルを共用できる(俺得)
- 俺はbashが使いたい。

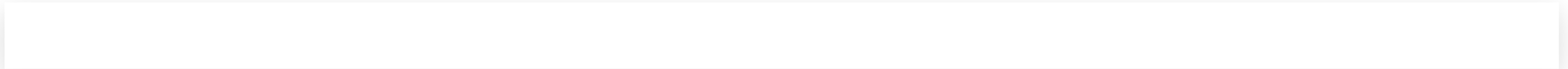
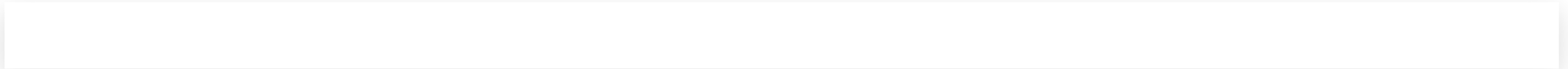


コマンドライン



コマンドライン

コマンドライン起動オプション



コマンドライン

コマンドライン起動オプション

- 現在のディレクトリでコードを開く

コマンドライン

コマンドライン起動オプション

- 現在のディレクトリでコードを開く

```
code .
```


コマンドライン

コマンドライン起動オプション

- 現在のディレクトリでコードを開く

```
code .
```

- 特定の行と列でファイルを開く `file : line [: character]`

コマンドライン

コマンドライン起動オプション

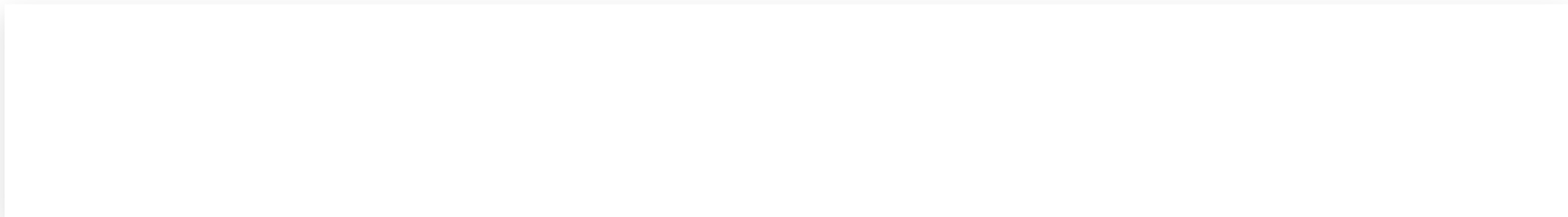
- 現在のディレクトリでコードを開く

```
code .
```

- 特定の行と列でファイルを開く file : line [: character]

```
code --goto package.json:10:5
```

差分



差分

- 差分エディタを開く

差分

- 差分エディタを開く

```
code --diff <file1> <file2>
```

差分

- 差分エディタを開く

```
code --diff <file1> <file2>
```

- .gitconfigに登録して、gitコマンドのdiffのときにVS Codeを使う。

差分

- 差分エディタを開く

```
code --diff <file1> <file2>
```

- .gitconfigに登録して、gitコマンドのdiffのときにVS Codeを使う。

```
[diff]
  tool = vscode
[difftool "vscode"]
  cmd = code --wait --diff $LOCAL $REMOTE
```

リセット



リセット

- すべての拡張機能を無効にして起動

リセット

- すべての拡張機能を無効にして起動

```
code --disable-extensions .
```

リセット

- すべての拡張機能を無効にして起動

```
code --disable-extensions .
```

- 設定などを全て削除

リセット

- すべての拡張機能を無効にして起動

```
code --disable-extensions .
```

- 設定などを全て削除
 1. %APPDATA%\Code を削除

リセット

- すべての拡張機能を無効にして起動

```
code --disable-extensions .
```

- 設定などを全て削除
 1. %APPDATA%\Code を削除
 2. %HOMEDRIVE%%HOMEPATH%.vscode を削除

まとめ

- VS Code、拡張機能が無くても結構できる子
- GitBashと組み合わせるとWSLが無くてもそれなりに使える。
- VS Codeを導入することで、少しでも現場の開発が楽しくなると良いですね。

まとめ

- VS Code、拡張機能が無くても結構できる子
- GitBashと組み合わせるとWSLが無くてもそれなりに使える。
- VS Codeを導入することで、少しでも現場の開発が楽しくなると良いですね。



謝辞

- 本日の発表資料は [vscode-reveal](#) で作成しました。
- 現場猫画像は からあげのるつぼ (@karaage_rutsubo) さんのツイートから拝借しました。
- わたしの使っている拡張機能などについては [この辺](#) に書いてあります。