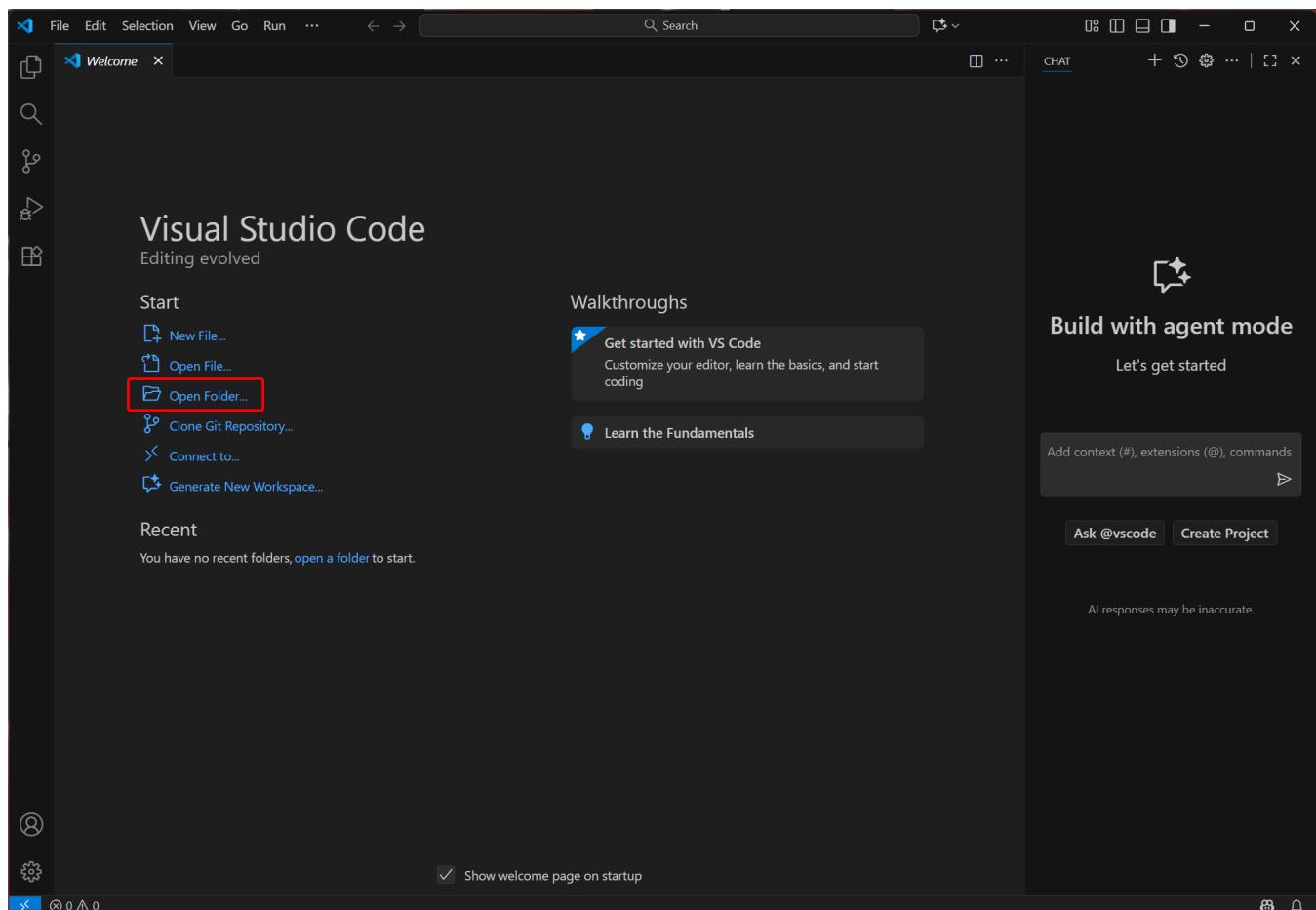


VSCode tutorials

1. Basic UI

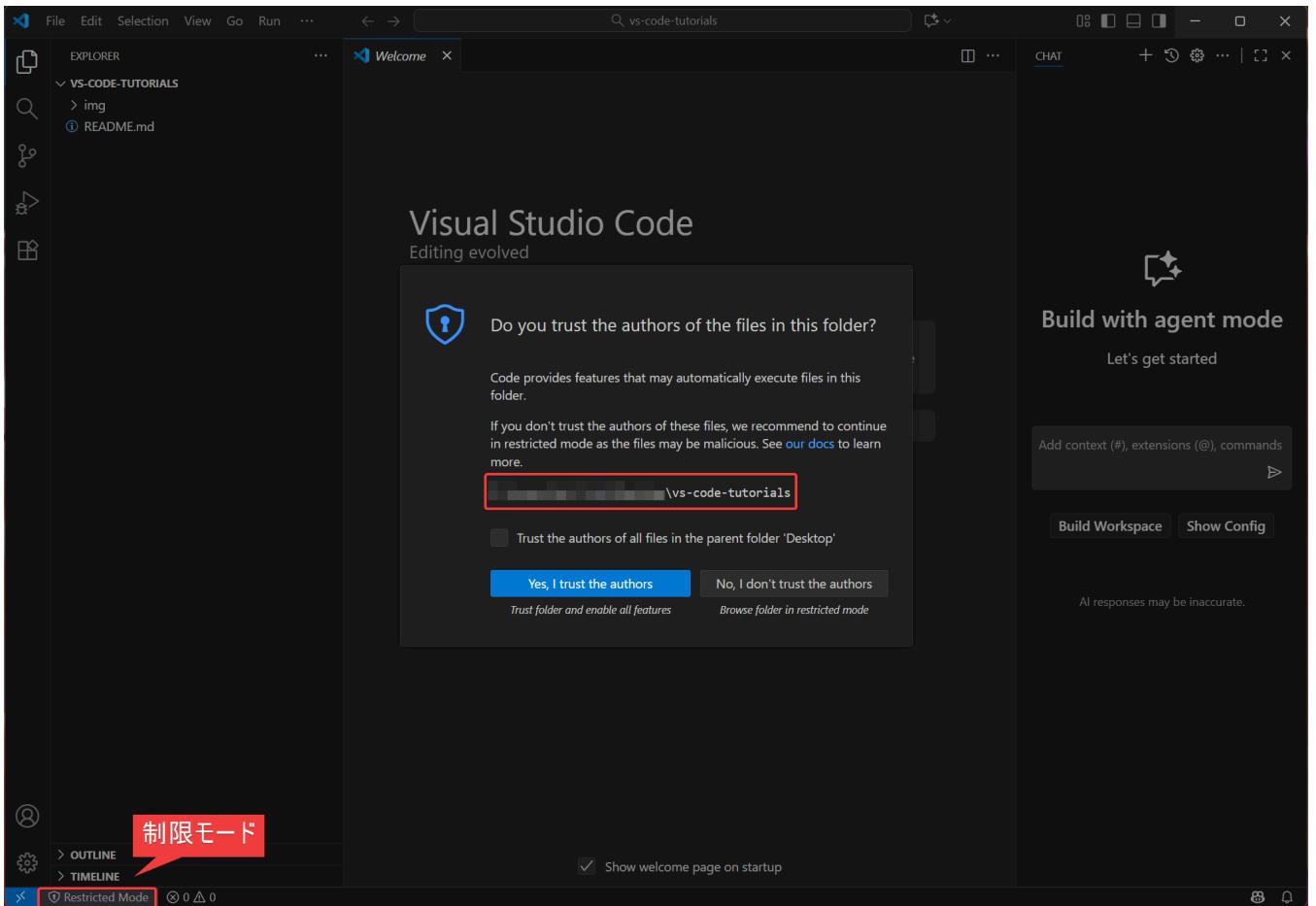
VScode を起動してください。

初期状態は以下の画面の通りになります。

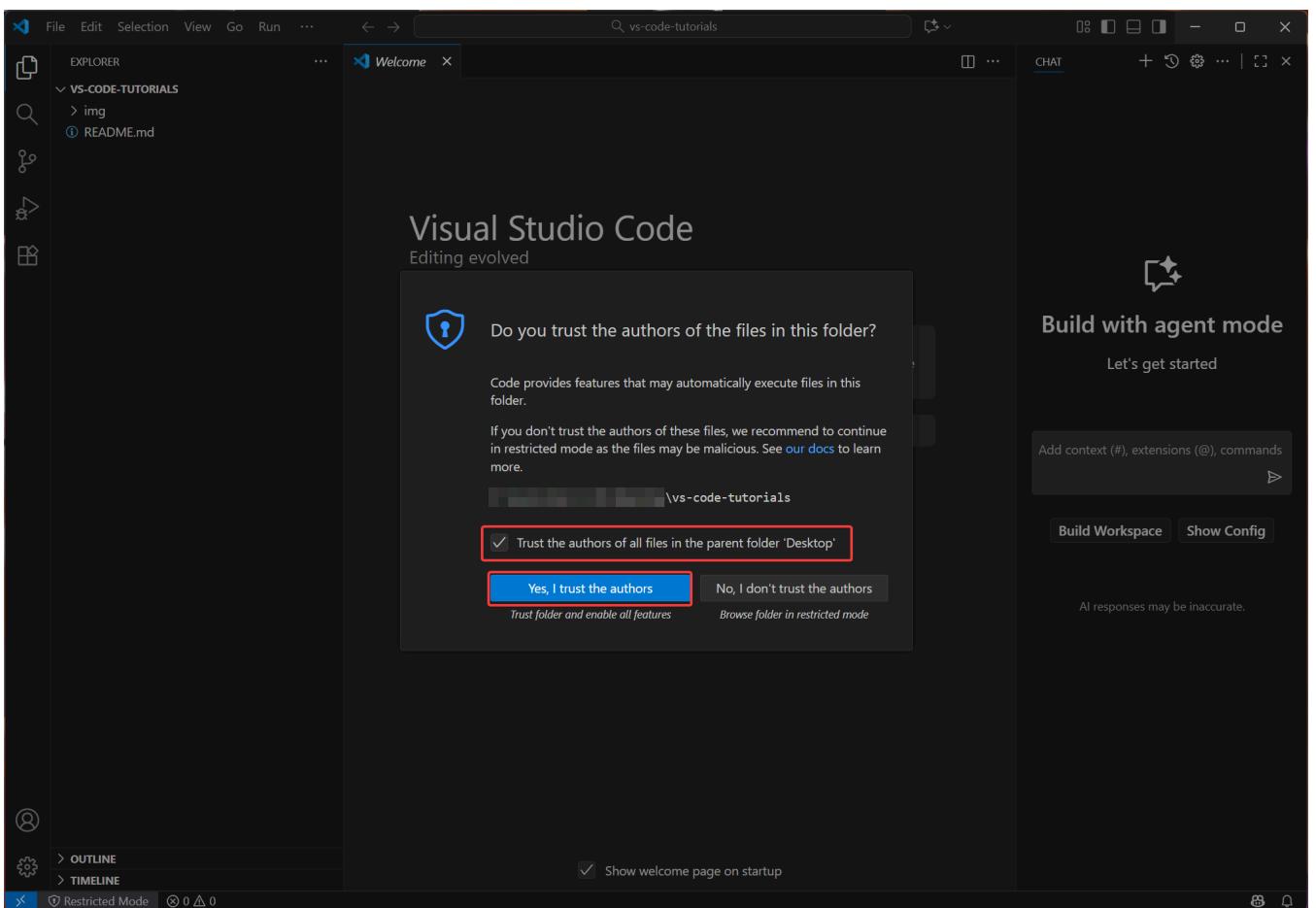


「Open Folder」をクリックして、開きたいフォルダーを選択してください。

初めて開くフォルダーやファイルだと、以下のような画面が表示される

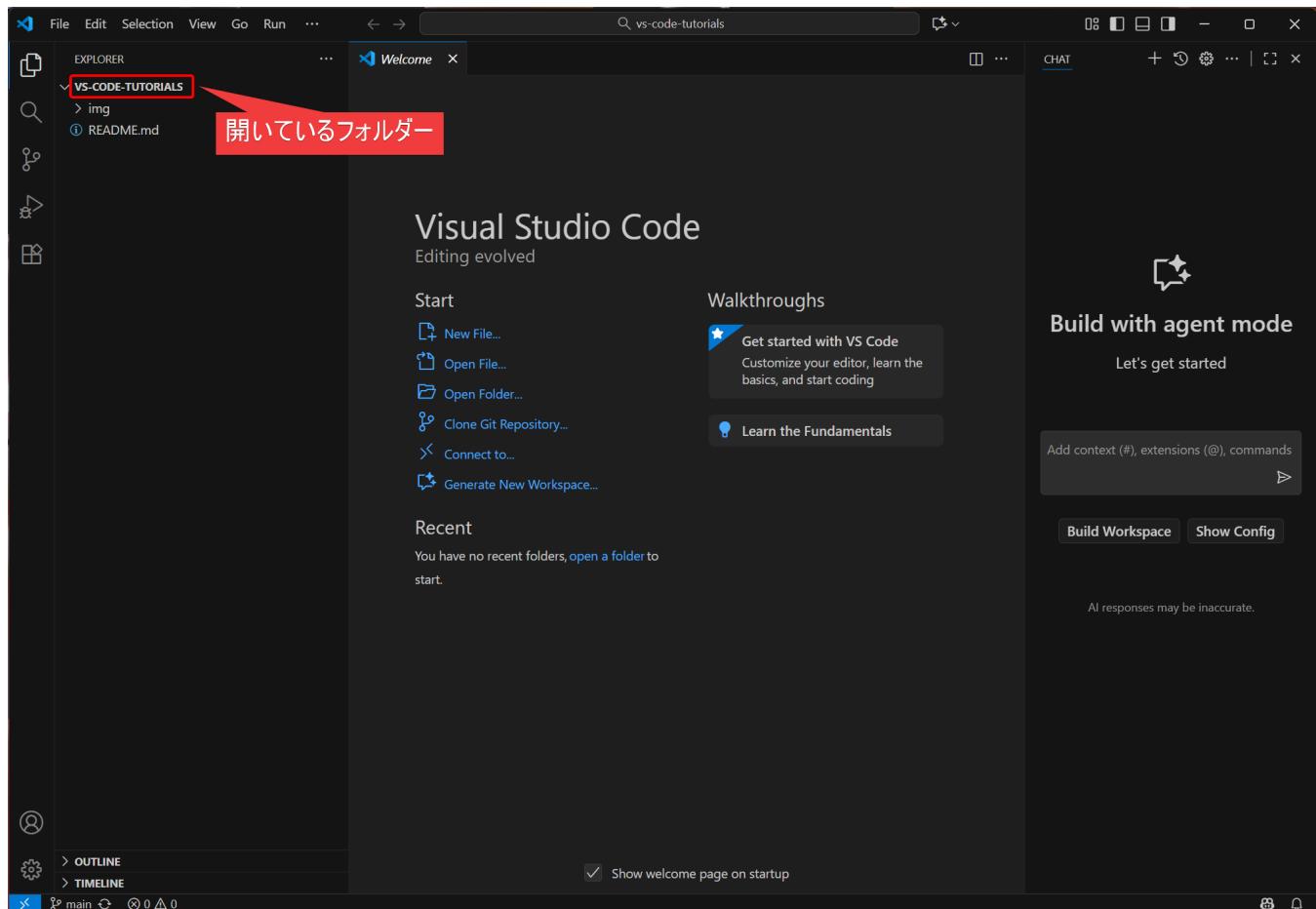


「Trust the authors....」のチェックボックスをチェックしないと、
「制限モード」で対象のファイルやフォルダーを開くので、要注意ください

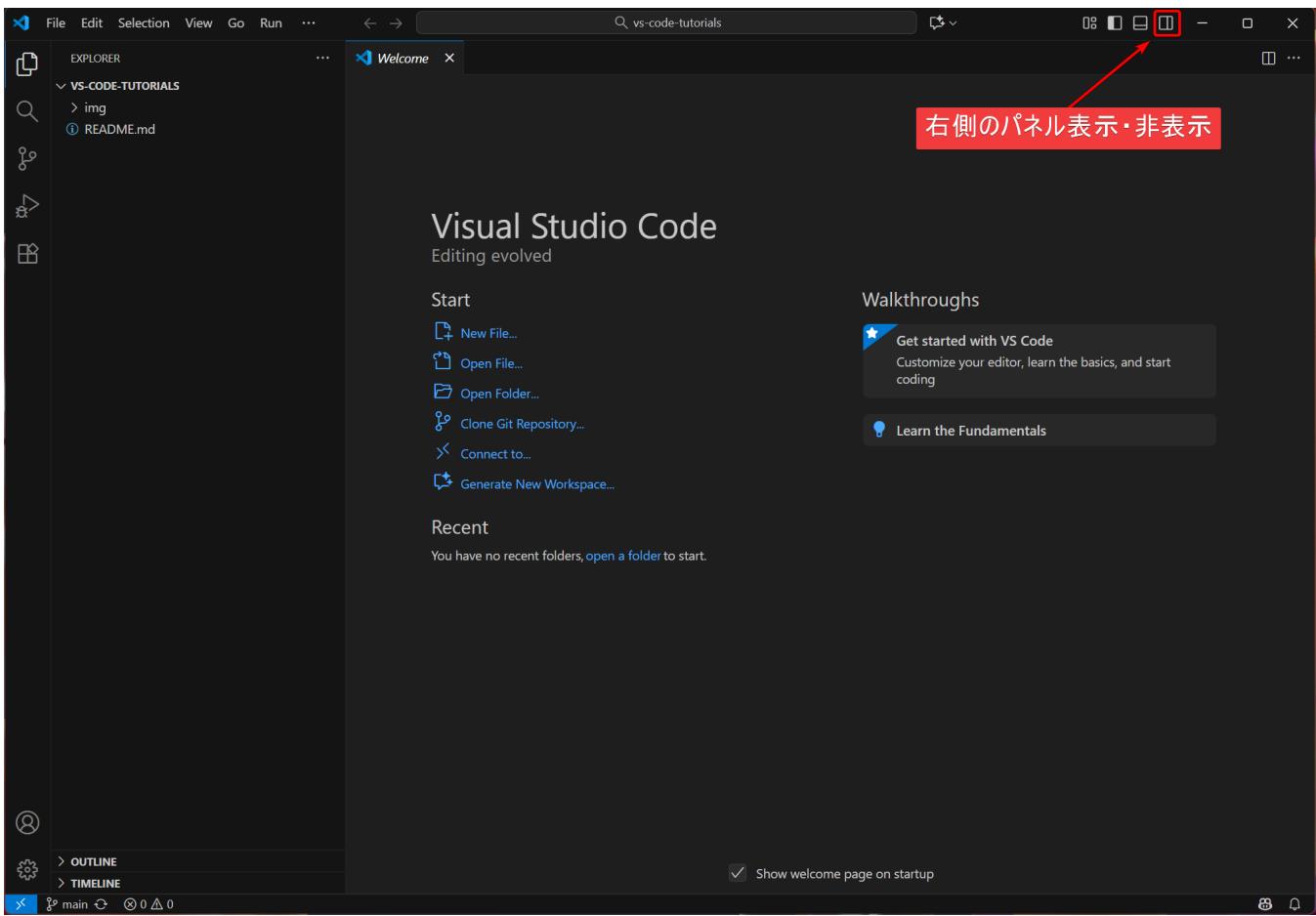


VS Code のユーザーインターフェースは 6 つの主要なエリアに分かれています:

- エディタ - ファイルを編集するメインエリアです。縦横に並べて、好きなだけ多くのエディタを開くことができます。
- プライマリサイドバー - プロジェクトでの作業中に役立つエクスプローラーなどのさまざまなビューが含まれています。
- セカンダリサイドバー - プライマリサイドバーの反対側にあります。デフォルトでは、チャットビューが含まれています。プライマリサイドバーからセカンダリサイドバーにビューをドラッグアンドドロップして移動できます。
- ステータスバー - 開いているプロジェクトと編集中のファイルに関する情報です。
- アクティビティバー - 最も左側に配置されています。ビュー間の切り替えを可能にし、Git が有効な場合の送信変更の数など、追加のコンテキスト固有のインジケーターを提供します。アクティビティバーの位置は変更できます。
- パネル - エディタ領域の下にある追加スペースです。デフォルトでは、出力、デバッグ情報、エラーと警告、および統合ターミナルが含まれています。パネルは、より多くの垂直スペースを確保するために左または右に移動することもできます。



利用しないサイドバー（右）は以下のような閉じることができる。



通常の利用する際は以下のような画面になる。

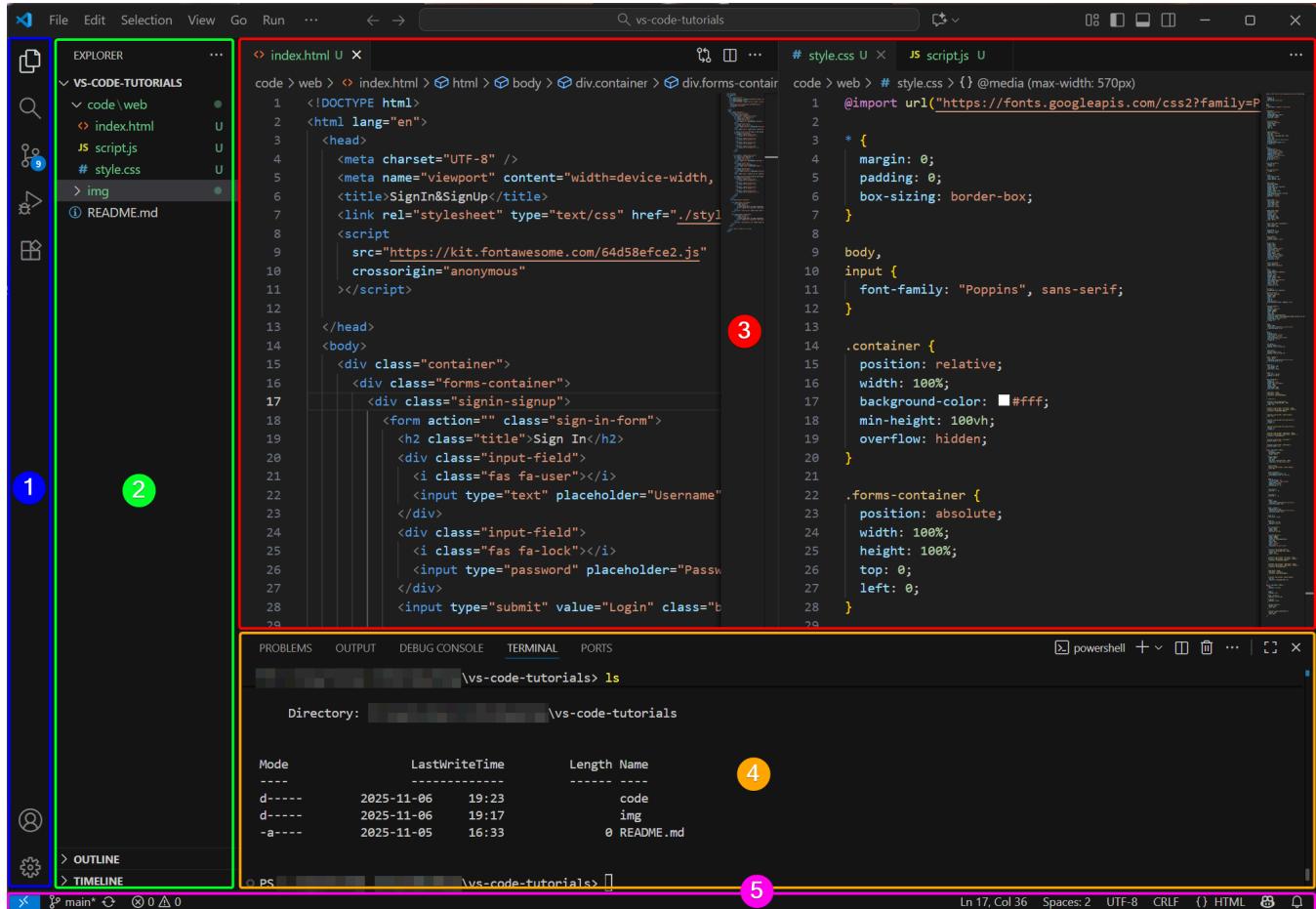
```

code > web > # style.css < JS script.js <
code > web > # style.css < {} @media (max-width: 570px)
1  @import url("https://fonts.googleapis.com/css2?family=P
2
3  * {
4    margin: 0;
5    padding: 0;
6    box-sizing: border-box;
7  }
8
9  body,
10 input {
11   font-family: "Poppins", sans-serif;
12 }
13
14 .container {
15   position: relative;
16   width: 100%;
17   background-color: #fff;
18   min-height: 100vh;
19   overflow: hidden;
20 }
21
22 .forms-container {
23   position: absolute;
24   width: 100%;
25   height: 100%;
26   top: 0;
27   left: 0;
28 }
29
30 .signin-signup {
31   position: absolute;
32   top: 50%;
33   transform: translate(-50%, -50%);
34   left: 75%;
35   width: 50%;
36   transition: 1s 0.7s ease-in-out;
37   display: grid;
38   grid-template-columns: 1fr;
39   z-index: 5;
40 }
41
42 form {
43   display: flex;

```

ターミナルも開くと以下のような画面が面になる。

これで VScode のすべての UI になる。



複数のファイルを同時に開きたい場合は「サイドバイサイド」を活用する。

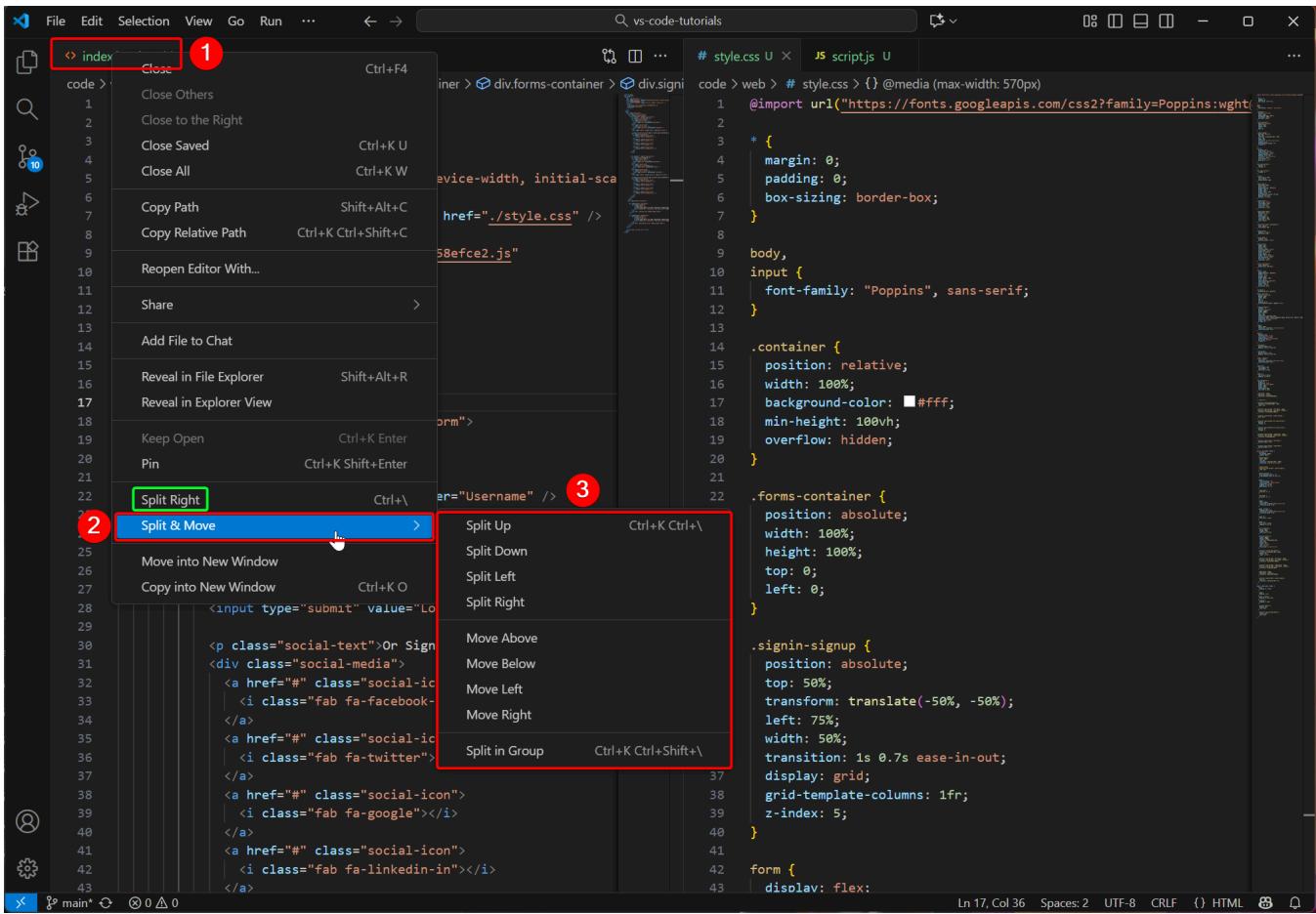
サイドバイサイド編集

縦横に並べて、好きなだけ多くのエディタを開くことができます。既にエディタを開いている場合、横に別のエディタを開く方法は複数あります：

- Alt キーを押しながら、エクスプローラービューでファイルを選択します。
- Ctrl+Shift キーでアクティブなエディタを 2 つに分割します。
- エクスプローラーのコンテキストメニューでファイル上の「横に開く」(Ctrl+Enter)を選択します。
- エディタの右上にある分割エディタボタンを選択します。
- エディタ領域の任意の側にファイルをドラッグアンドドロップします。ドラッグ中に Ctrl(macOS では Option)キーを押したままにすると、タブを移動する代わりにコピーします。
- クリックオープン(Ctrl+P)のファイルリストで Ctrl+Enter キーを押します。

マウスで操作する際は、対象のエディタータブに右クリックして、「Split & Move」を選択し、配置場所を選択する。

単純に右側に分割して表示したい場合は「Split Right」をクリックしてください



エディタグループ

エディタを分割する(分割エディタまたは横に開くコマンドを使用)と、新しいエディタ領域(編集グループ)が作成され、アイテムのグループを保持できます。縦横に並べて、好きなだけ多くのエディタグループを開くことができます。

これらは、エクスプローラービューの上部にある「開いているエディタ」セクションで明確に確認できます(エクスプローラービューで...>開いているエディタを切り替え)。

タブエディタグループ

ワークベンチ上でエディタグループをドラッグアンドドロップしたり、グループ間で個別のタブを移動したり、グループ全体を素早く閉じる(すべて閉じる)ことができます。

The screenshot shows the VS Code interface with the 'Open Editors' feature highlighted. The Explorer sidebar on the left lists 'VS-CODE-TUTORIALS' with 'code\web' expanded, showing files like 'index.html', 'script.js', '# style.css', and 'img'. The main editor area displays the 'index.html' file's code. The status bar at the bottom right shows 'Ln 17, Col 36' and other settings.

This screenshot illustrates the grouping feature in VS Code's 'Open Editors' view. The 'OPEN EDITORS' section in the Explorer sidebar is divided into two groups: 'GROUP 1' (containing 'index.html' and 'script.js') and 'GROUP 2' (containing '# style.css' and 'img'). The main editor area shows the 'index.html' file's code. The status bar at the bottom right shows 'Ln 17, Col 36' and other settings.

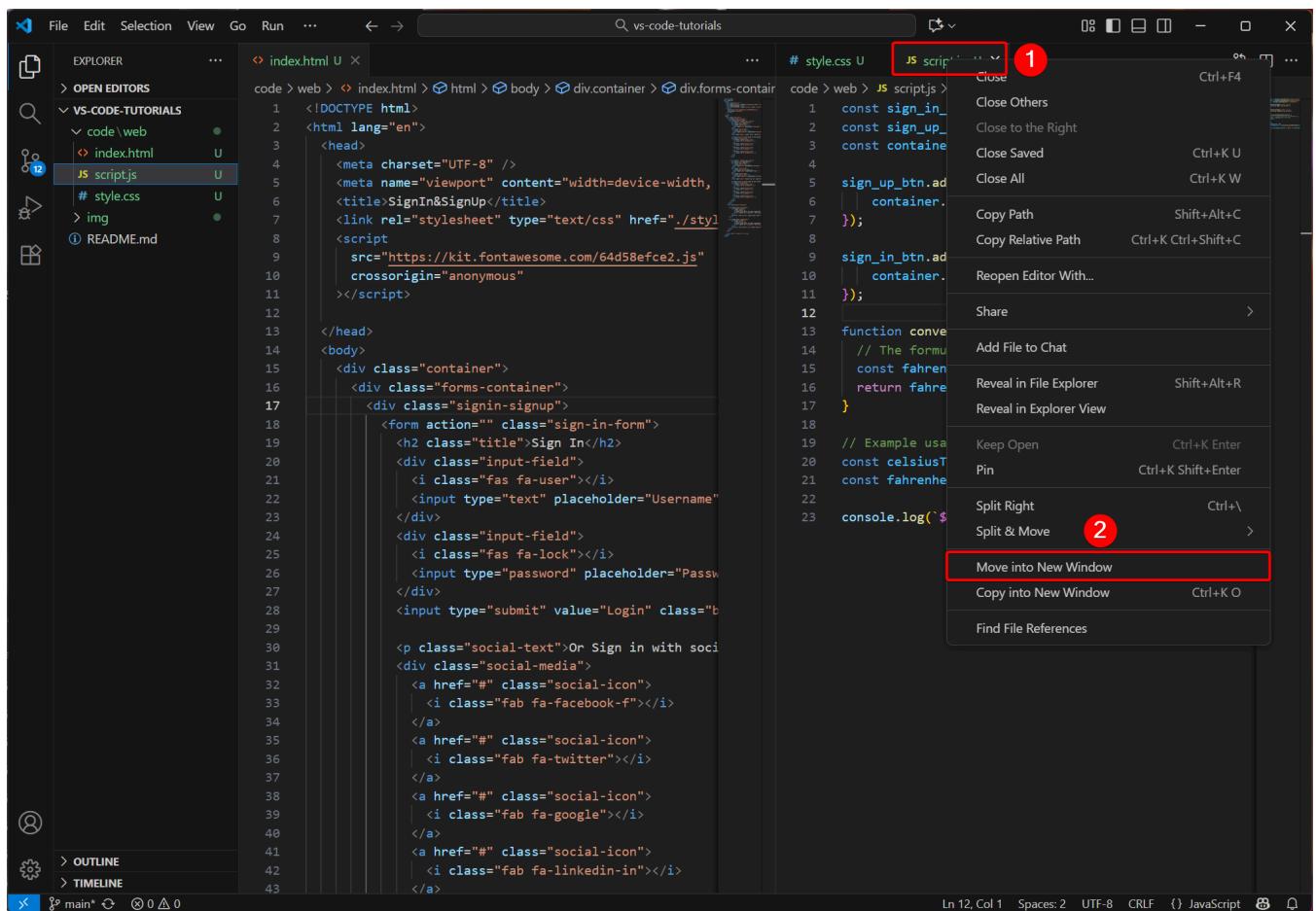
フローティングウィンドウ

エディタ、ターミナル、または特定のビューを独自のフローティングウィンドウに移動できます。これは、マルチモニター環境があり、別のモニターでファイルを開いたままにしておきたい場合に便利です。

VS Code ウィンドウとフローティングウィンドウで開かれた CSV ファイルを表示する OS デスクトップのスクリーンショット。

現在の VS Code ウィンドウからエディタタブをドラッグして、フローティングウィンドウで開きます。または、エディタタブのコンテキストオプションから「新しいウィンドウに移動」または「新しいウィンドウにコピー」を使用します。

フローティングウィンドウを画面の最前面に固定するには、タイトルレバーから「常に最前面に表示」オプション(ビンアイコン)を選択します。



The screenshot shows the Visual Studio Code interface with two floating windows. The top window is titled 'vs-code-tutorials' and contains the code for 'script.js'. The bottom window is titled '# style.css' and contains the code for 'style.css'. A red box highlights the floating nature of these windows.

```
const sign_in_btn = document.querySelector("#sign-in-btn");
const sign_up_btn = document.querySelector("#sign-up-btn");
const container = document.querySelector(".container");

sign_up_btn.addEventListener('click', () =>{
  container.classList.add("sign-up-mode");
});

sign_in_btn.addEventListener('click', () =>{
  container.classList.remove("sign-up-mode");
});

function convertCelsiusToFahrenheit(celsius) {
  // The formula for converting Celsius to Fahrenheit is: (Celsius * 9/5) + 32
  const fahrenheit = (celsius * 9 / 5) + 32;
  return fahrenheit;
}

// Example usage:
const celsiusTemperature = 25;
const fahrenheitTemperature = convertCelsiusToFahrenheit(celsiusTemperature);

console.log(`${celsiusTemperature}°C is equal to ${fahrenheitTemperature}°F`); // Output: 25°C is equal to 77°F
```

```
form {
  display: flex;
```

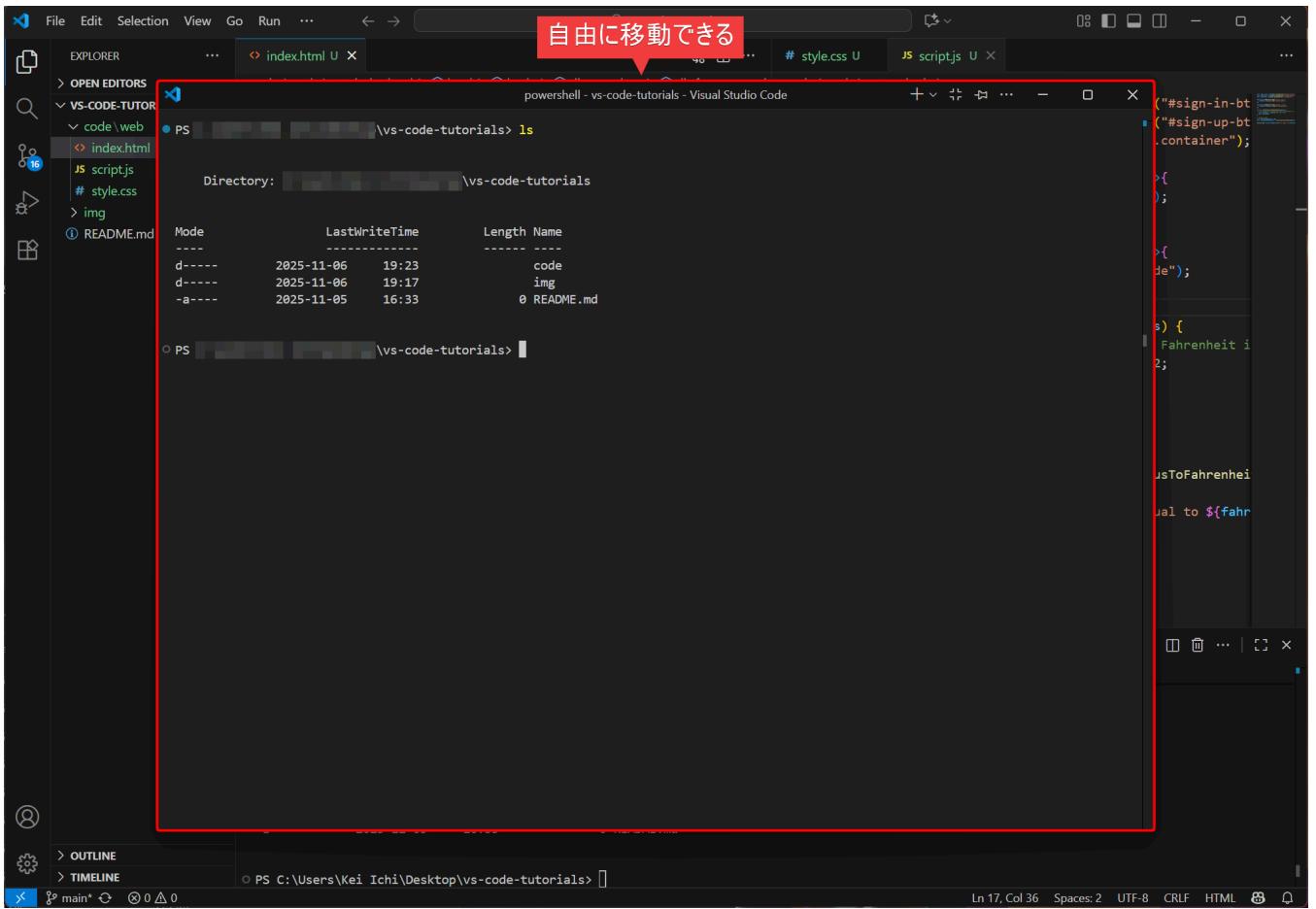
同じくターミナルもフローティングウィンドウで開ける

The screenshot shows the Visual Studio Code interface with a floating terminal window. The terminal window is titled 'vs-code-tutorials' and has a red circle with the number '1' at its bottom right corner. A red box highlights the floating nature of the terminal window. The terminal content shows a directory listing:

```
Directory: \vs-code-tutorials> ls
```

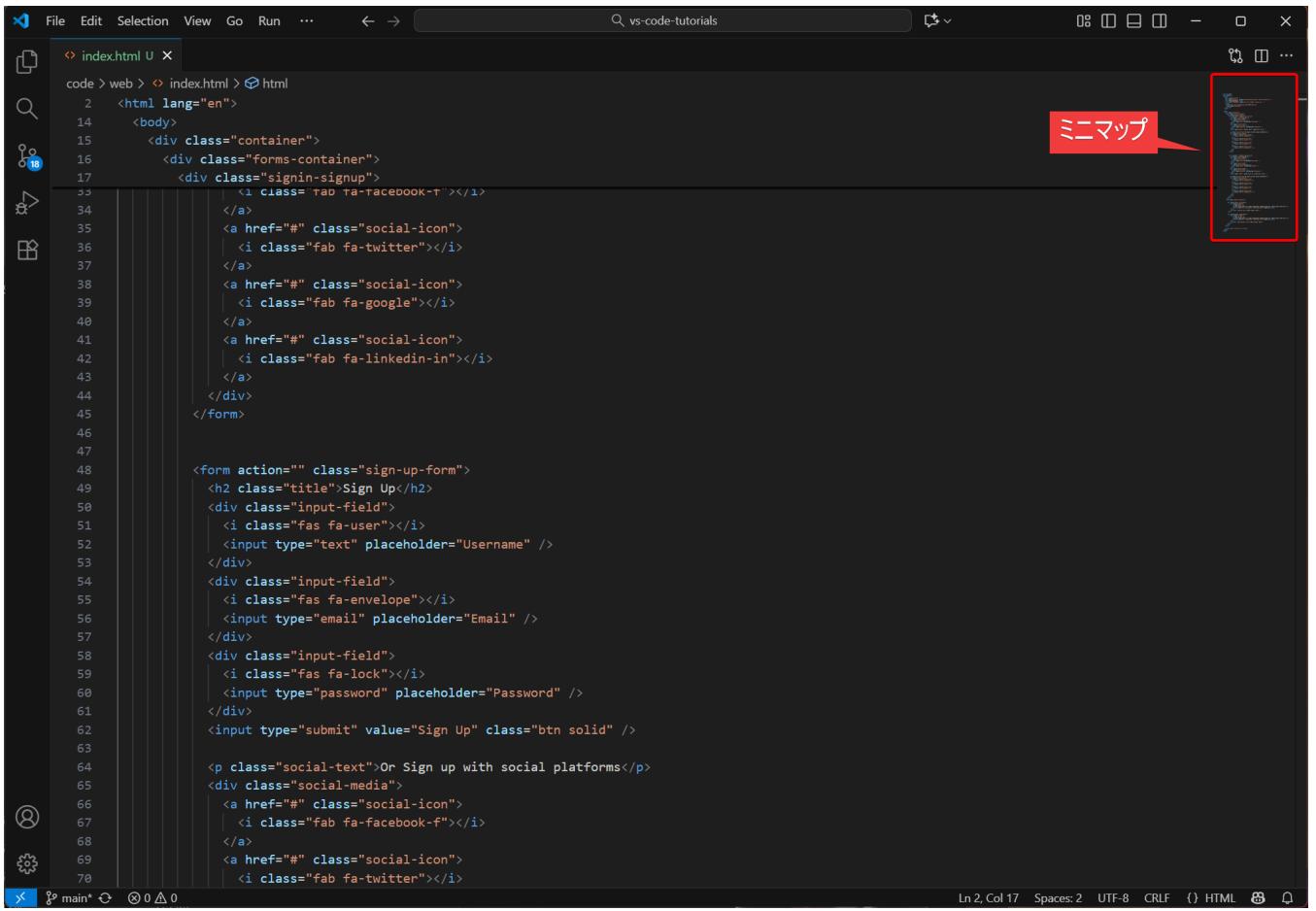
Mode	LastWriteTime	Length	Name
d----	2025-11-06 19:23		code
d----	2025-11-06 19:17		img
-a---	2025-11-05 16:33	0	README.md

A context menu is open in the terminal area, with the option 'New Terminal Window' highlighted by a red box and the number '2'.



ミニマップ

ミニマップ(コードアウトライン)は、ソースコードの概要を表示し、素早いナビゲーションとコード理解に役立ちます。ファイルのミニマップはエディタの右側に表示されます。影付きの領域を選択またはドラッグして、ファイル内の異なるセクションに素早くジャンプできます。

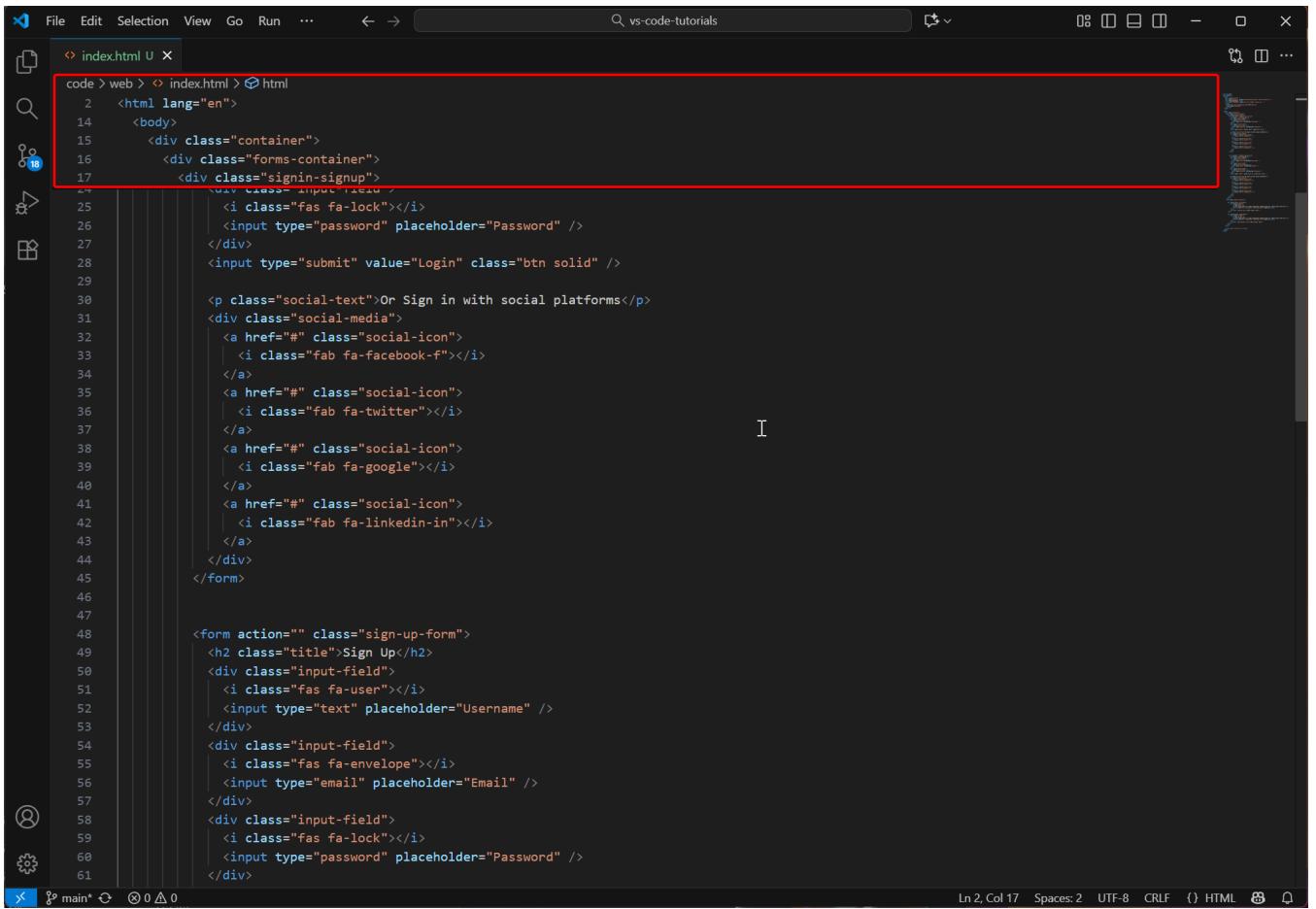


The screenshot shows the Visual Studio Code interface with the file "index.html" open. The code editor displays the HTML structure of a sign-up form. A red box highlights the "Minimap" icon in the top right corner of the interface. The minimap is a small preview of the entire document, showing the layout and position of the code blocks.

```
2   <html lang="en">
4     <body>
5       <div class="container">
6         <div class="forms-container">
7           <div class="signin-signup">
8             <a href="#" class="social-icon">
9               <i class="fab fa-facebook-f"></i>
10              </a>
11              <a href="#" class="social-icon">
12                <i class="fab fa-twitter"></i>
13              </a>
14              <a href="#" class="social-icon">
15                <i class="fab fa-google"></i>
16              </a>
17              <a href="#" class="social-icon">
18                <i class="fab fa-linkedin-in"></i>
19              </a>
20            </div>
21          </div>
22        </form>
23
24
25        <form action="" class="sign-up-form">
26          <h2 class="title">Sign Up</h2>
27          <div class="input-field">
28            <i class="fas fa-user"></i>
29            <input type="text" placeholder="Username" />
30          </div>
31          <div class="input-field">
32            <i class="fas fa-envelope"></i>
33            <input type="email" placeholder="Email" />
34          </div>
35          <div class="input-field">
36            <i class="fas fa-lock"></i>
37            <input type="password" placeholder="Password" />
38          </div>
39          <input type="submit" value="Sign Up" class="btn solid" />
40
41
42          <p class="social-text">Or Sign up with social platforms</p>
43          <div class="social-media">
44            <a href="#" class="social-icon">
45              <i class="fab fa-facebook-f"></i>
46            </a>
47            <a href="#" class="social-icon">
48              <i class="fab fa-twitter"></i>
49            </a>
50          </div>
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
```

ステイツキースクロール

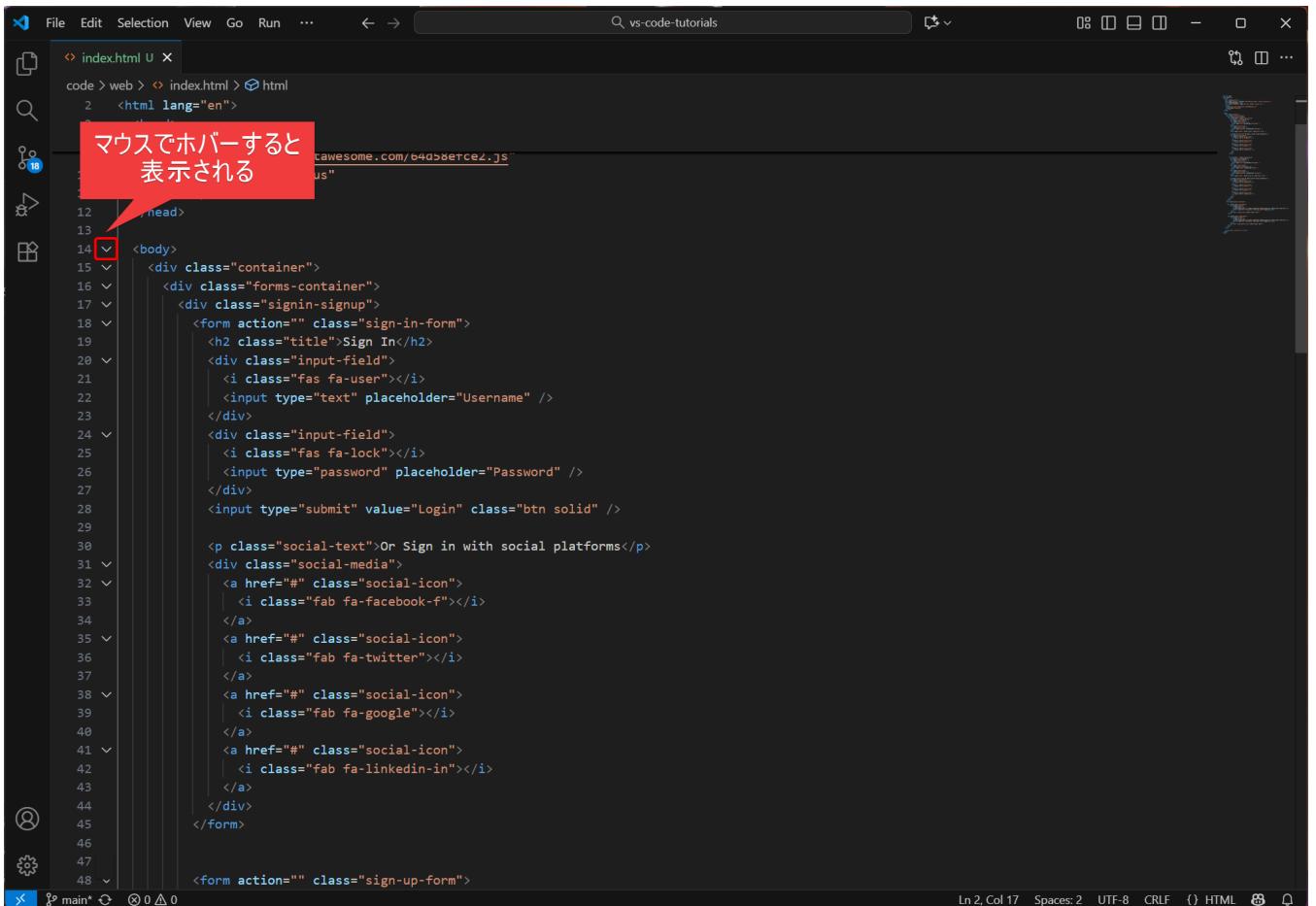
ステイツキースクロールは、現在表示されているネストされたスコープの開始行をエディタの上部に表示します。ファイル内のどこにいるかを示すことでナビゲーションを容易にし、現在のスコープの先頭に素早く戻ることができます。



```
2   <html lang="en">
14     <body>
15       <div class="container">
16         <div class="forms-container">
17           <div class="signin-signup">
18             <form action="#" class="sign-in-form">
19               <h2 class="title">Sign In</h2>
20               <div class="input-field">
21                 <i class="fas fa-user"></i>
22                 <input type="text" placeholder="Username" />
23               </div>
24               <div class="input-field">
25                 <i class="fas fa-lock"></i>
26                 <input type="password" placeholder="Password" />
27               </div>
28               <input type="submit" value="Login" class="btn solid" />
29
30             <p class="social-text">Or Sign in with social platforms</p>
31             <div class="social-media">
32               <a href="#" class="social-icon">
33                 <i class="fab fa-facebook-f"></i>
34               </a>
35               <a href="#" class="social-icon">
36                 <i class="fab fa-twitter"></i>
37               </a>
38               <a href="#" class="social-icon">
39                 <i class="fab fa-google"></i>
40               </a>
41               <a href="#" class="social-icon">
42                 <i class="fab fa-linkedin-in"></i>
43               </a>
44             </div>
45           </form>
46
47
48           <form action="#" class="sign-up-form">
49             <h2 class="title">Sign Up</h2>
50             <div class="input-field">
51               <i class="fas fa-user"></i>
52               <input type="text" placeholder="Username" />
53             </div>
54             <div class="input-field">
55               <i class="fas fa-envelope"></i>
56               <input type="email" placeholder="Email" />
57             </div>
58             <div class="input-field">
59               <i class="fas fa-lock"></i>
60               <input type="password" placeholder="Password" />
61             </div>

```

コードブロックを縮小場合は対象のブロックをクリックすると、縮小することができる。



マウスでホバーすると表示される

```
2   <html lang="en">
3     <head>
4       <meta charset="UTF-8" />
5       <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6       <title>Awesome App</title>
7       <link href="https://unpkg.com/tailwindcss@^2/dist/tailwind.min.css" rel="stylesheet" />
8       <script src="https://unpkg.com/awesome.js" defer></script>
9
10      <!-- Global styles -->
11      <style>
12        body {
13          font-family: Arial, sans-serif;
14          margin: 0;
15          padding: 0;
16        }
17        .container {
18          width: 1000px;
19          margin: auto;
20        }
21        .forms-container {
22          display: flex;
23          justify-content: space-around;
24        }
25        .form {
26          width: 450px;
27        }
28        .input-field {
29          position: relative;
30        }
31        .input-field i {
32          position: absolute;
33          top: 50%;
34          left: 15px;
35          color: #ccc;
36        }
37        .input-field input {
38          width: 100%;
39          height: 40px;
40          border: 1px solid #ccc;
41          padding: 0 15px;
42        }
43        .input-field::before {
44          content: '';
45          position: absolute;
46          bottom: -10px;
47          left: 50px;
48          width: 0;
49          height: 0;
50          border-left: 15px solid transparent;
51          border-right: 15px solid transparent;
52          border-top: 30px solid #ccc;
53        }
54        .social-text {
55          text-align: center;
56        }
57        .social-media {
58          display: flex;
59          justify-content: space-around;
60        }
61        .social-icon {
62          width: 40px;
63          height: 40px;
64          border-radius: 50%;
65          background-color: #fff;
66          display: flex;
67          align-items: center;
68          justify-content: center;
69          font-size: 1.5em;
70          color: #ccc;
71          text-decoration: none;
72        }
73        .social-icon i {
74          color: #ccc;
75        }
76        .social-icon:hover {
77          background-color: #f0f0f0;
78        }
79        .social-icon:hover i {
80          color: #007bff;
81        }
82        .btn {
83          width: 100px;
84          height: 40px;
85          border: 1px solid #007bff;
86          border-radius: 5px;
87          background-color: #007bff;
88          color: white;
89          text-decoration: none;
90          font-weight: bold;
91          font-size: 1em;
92          padding: 0 10px;
93          display: flex;
94          align-items: center;
95          justify-content: center;
96        }
97        .btn solid {
98          background-color: #007bff;
99          border: 1px solid #007bff;
100         color: white;
101        }
102        .btn solid:hover {
103          background-color: #00695c;
104          border: 1px solid #00695c;
105        }
106        .btn outline {
107          border: 1px solid #007bff;
108          color: #007bff;
109        }
110        .btn outline:hover {
111          border: 1px solid #00695c;
112        }
113      </style>
114    </head>
115    <body>
116      <div class="container">
117        <div class="forms-container">
118          <div class="signin-signup">
119            <form action="#" class="sign-in-form">
120              <h2 class="title">Sign In</h2>
121              <div class="input-field">
122                <i class="fas fa-user"></i>
123                <input type="text" placeholder="Username" />
124              </div>
125              <div class="input-field">
126                <i class="fas fa-lock"></i>
127                <input type="password" placeholder="Password" />
128              </div>
129              <input type="submit" value="Login" class="btn solid" />
130
131            <p class="social-text">Or Sign in with social platforms</p>
132            <div class="social-media">
133              <a href="#" class="social-icon">
134                <i class="fab fa-facebook-f"></i>
135              </a>
136              <a href="#" class="social-icon">
137                <i class="fab fa-twitter"></i>
138              </a>
139              <a href="#" class="social-icon">
140                <i class="fab fa-google"></i>
141              </a>
142              <a href="#" class="social-icon">
143                <i class="fab fa-linkedin-in"></i>
144              </a>
145            </div>
146          </form>
147
148          <form action="#" class="sign-up-form">
149            <h2 class="title">Sign Up</h2>
150            <div class="input-field">
151              <i class="fas fa-user"></i>
152              <input type="text" placeholder="Username" />
153            </div>
154            <div class="input-field">
155              <i class="fas fa-envelope"></i>
156              <input type="email" placeholder="Email" />
157            </div>
158            <div class="input-field">
159              <i class="fas fa-lock"></i>
160              <input type="password" placeholder="Password" />
161            </div>
162          </form>
163
164        </div>
165      </div>
166    </body>
167  </html>
```

A screenshot of the Visual Studio Code interface. The main area shows an HTML file named 'index.html' with the following code:

```
2  <html lang="en">
3    <head>
8      <script
9        src="https://kit.tontawesome.com/b4d58erce2.js"
10       crossorigin="anonymous"
11     ></script>
12   </head>
14 > <body> ...
105  </body>
106 </html>
```

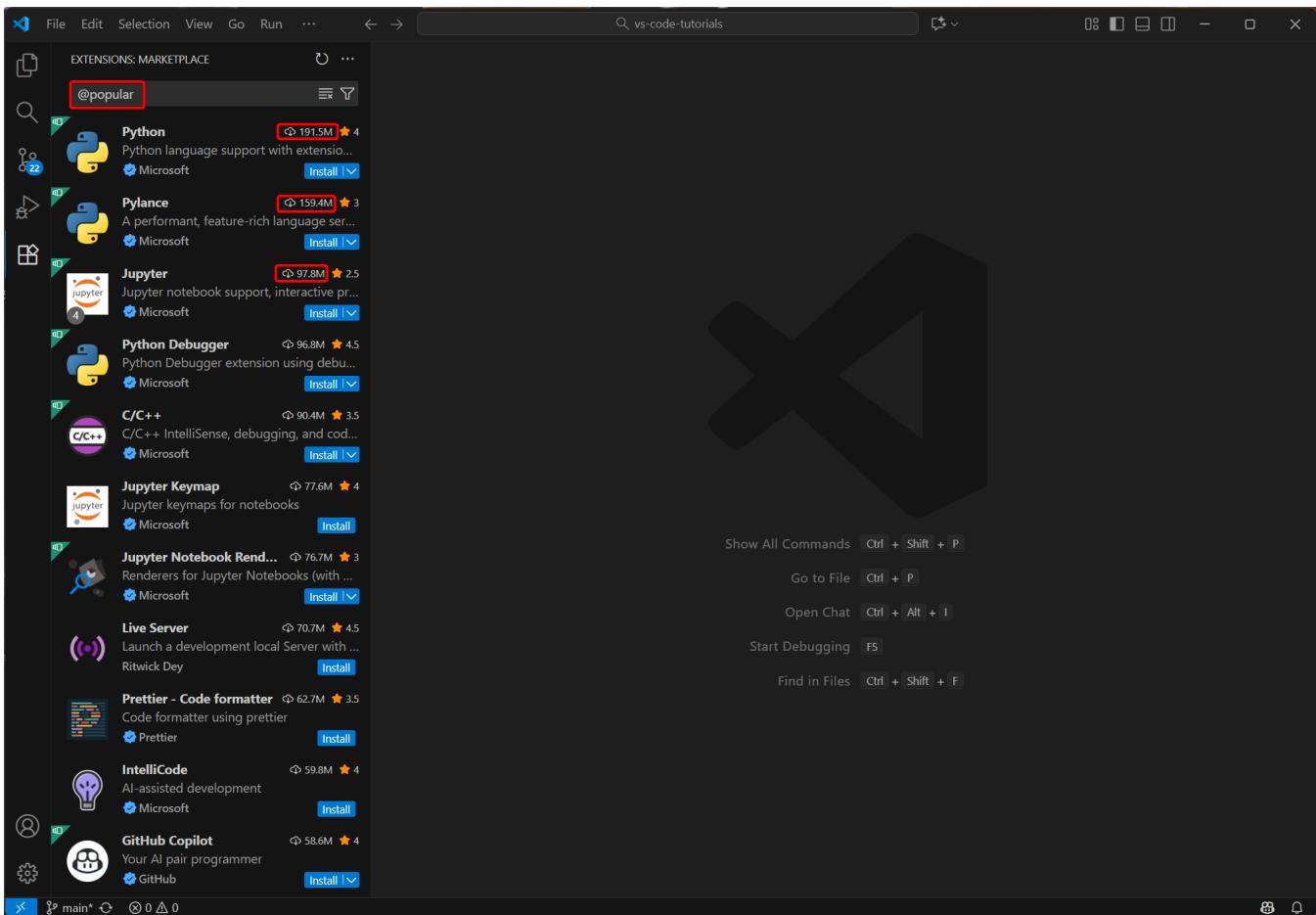
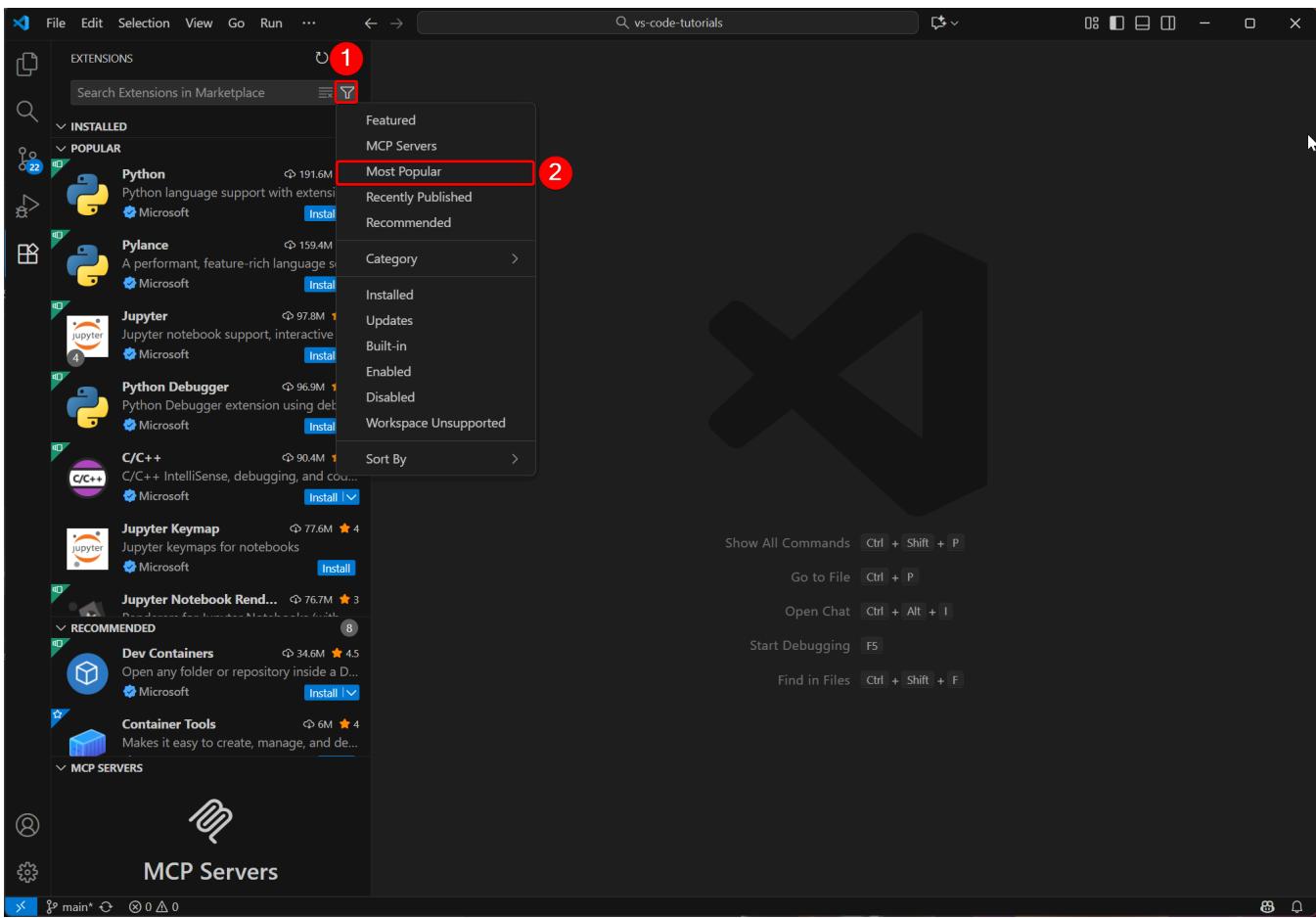
A red box highlights the content between line 14 and 105. A red callout bubble with the Japanese text 'コードブロックの中身が縮小されている' (The content of the code block is being collapsed) points to this red box.

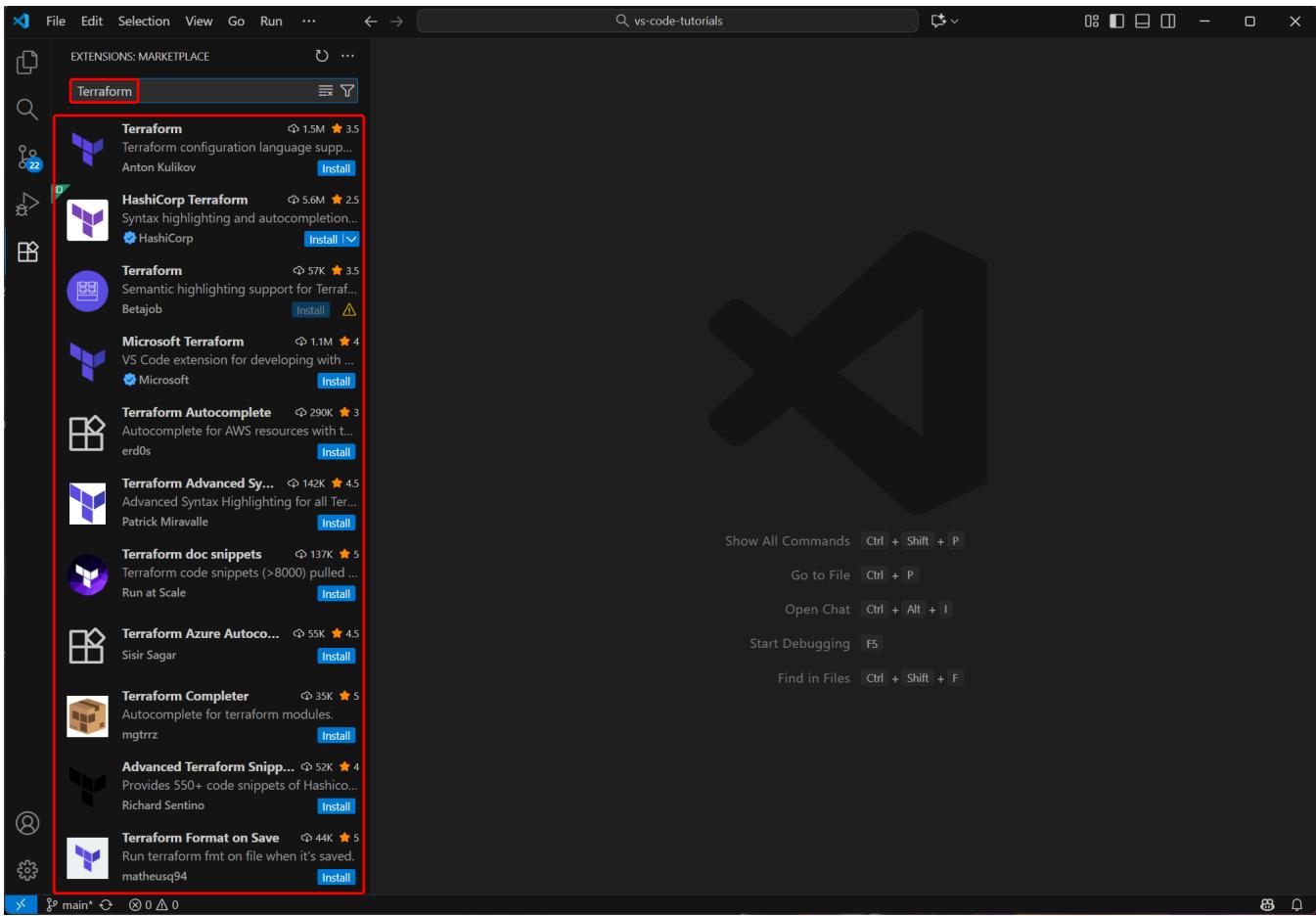
2. Install Extensions

A screenshot of the Visual Studio Code interface showing the Extensions view. The sidebar on the left lists installed extensions under the 'INSTALLED' section:

- Python (191.6M)
- Pylance (159.4M)
- Jupyter (97.8M)
- Python Debugger (96.9M)
- C/C++ (90.4M)
- Jupyter Keymap (77.6M)
- Jupyter Notebook Render (76.7M)
- Dev Containers (34.6M)
- Container Tools (6M)

The 'Python' extension is highlighted with a red box. The status bar at the bottom indicates 'Ln 2, Col 17 Spaces: 2 UTF-8 CRLF () HTML'.





The screenshot shows the Visual Studio Code interface with the Marketplace extension open, displaying the details page for the HashiCorp Terraform extension. The search bar at the top contains the text 'vs-code-tutorials'. The left sidebar shows pinned extensions like 'File History', 'GitLens', and 'Prettier'. The main area shows the following details for the HashiCorp Terraform extension:

Extension: HashiCorp Terraform

開発元・開発者: HashiCorp (hashicorp.com) | 5,681,715 installs | ★★★★☆ (206)

Syntax highlighting and autocompletion for Terraform

Terraform Extension for Visual Studio Code

The HashiCorp Terraform Extension for Visual Studio Code (VS Code) with the Terraform Language Server adds editing features for Terraform, Terraform Stacks and Terraform Search files such as syntax highlighting, IntelliSense, code navigation, code formatting, module explorer and much more!

利用手順

Get started writing Terraform configurations with VS Code in three steps:

- Step 1: If you haven't done so already, install Terraform
- Step 2: Install the Terraform Extension for VS Code.
- Step 3: To activate the extension, open any folder or VS Code workspace containing Terraform, Terraform Stacks or Terraform Search files. Once activated, the Terraform language indicator will appear in the bottom right corner of the window.

New to Terraform? Read the [Terraform Learning guides](#)

See [Usage](#) for more detailed getting started information.

Read the [Troubleshooting Guide](#) for answers to common questions.

Features

This extension provides Terraform, Terraform Stacks and Terraform Search language features. For most features Terraform Stacks and Terraform Search support is implied. ▾ specific functionality is called out where appropriate.

Marketplace

- Identifier: hashicorp.terraform
- Version: 2.37.5
- Published: 9 years ago
- Last Updated: 1 month ago
- Released: 1 month ago

Categories

- Programming Languages
- Linters
- Formatters

Resources

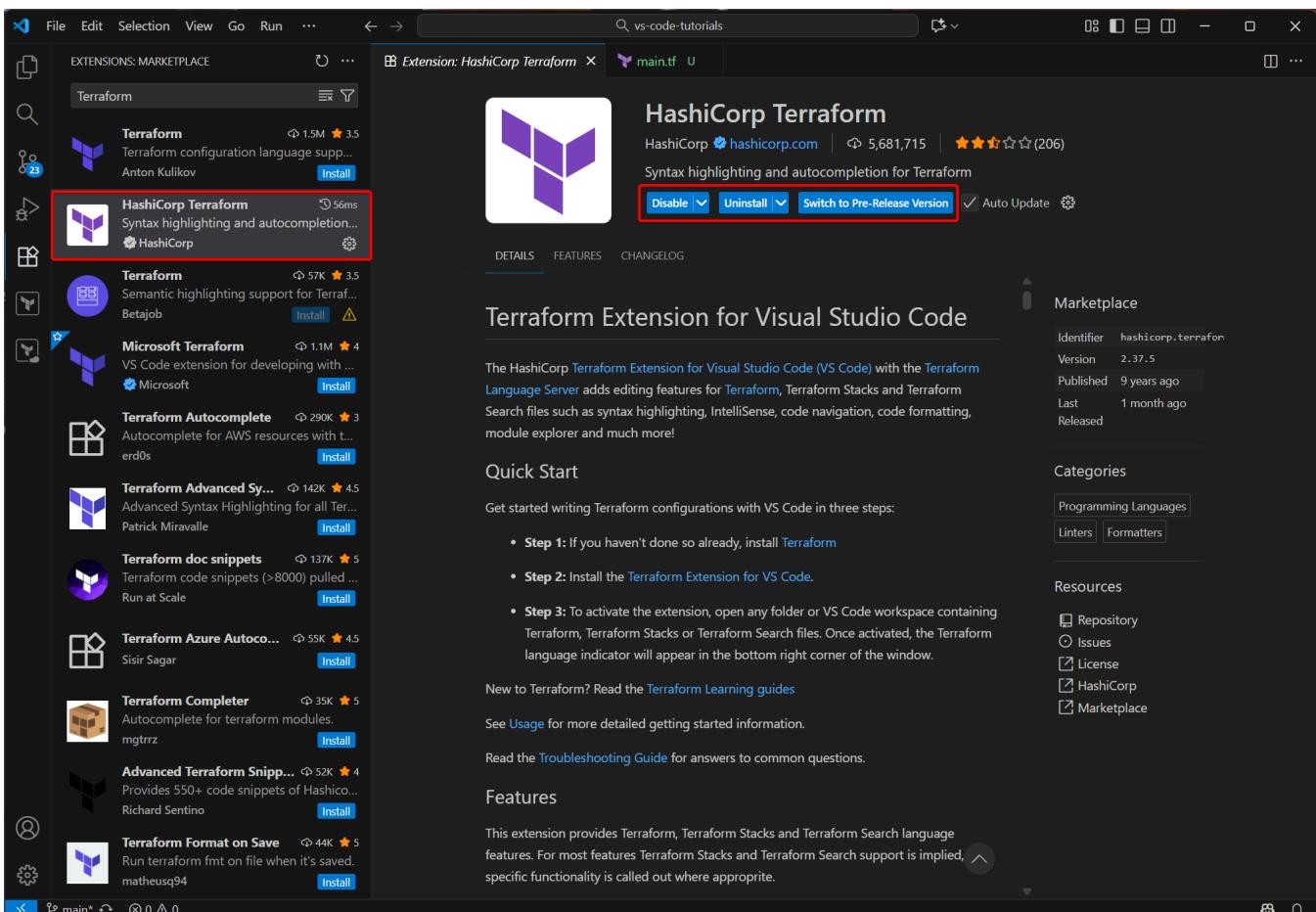
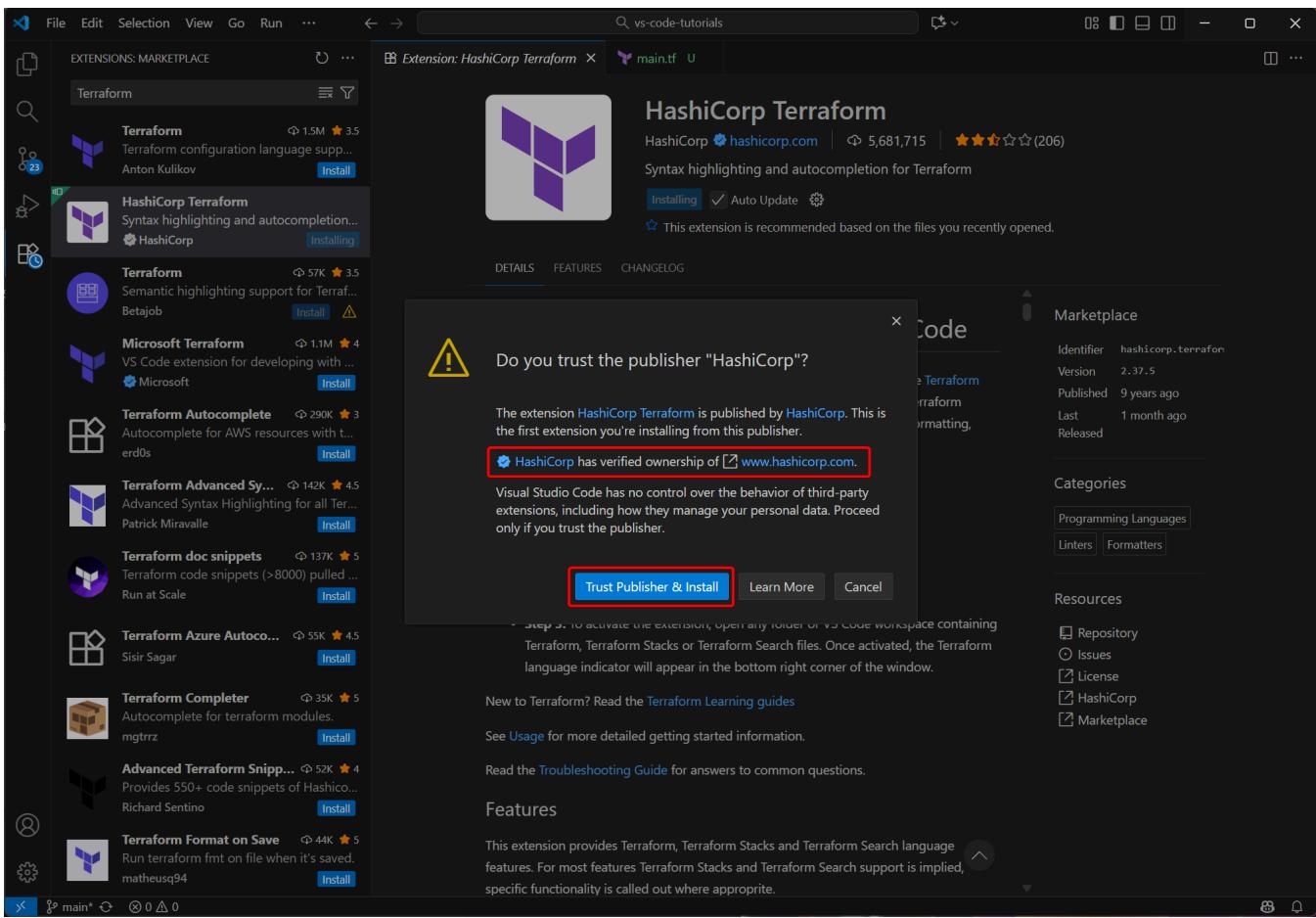
- Repository
- Issues
- License
- HashiCorp
- Marketplace

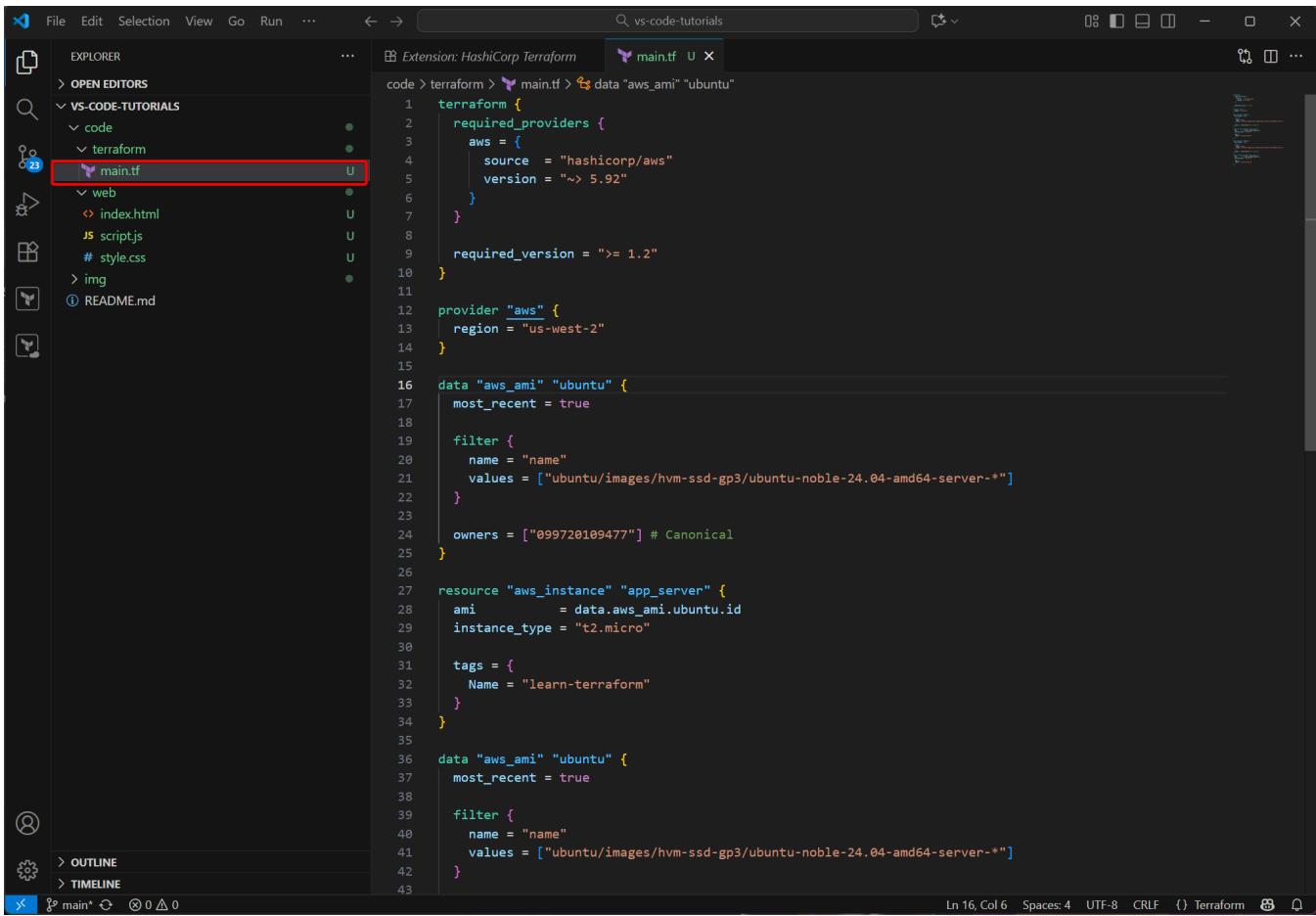
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder structure under "VS-CODE-TUTORIALS" containing "code", "terraform", and "img" folders. Inside "code", there are "index.html", "script.js", and "# style.css".
- Editor:** The "main.tf" file is open in the editor. The code is a Terraform configuration for creating an AWS instance. It includes providers for AWS and Ubuntu AMI, and resources for an AWS instance with specific tags and filters.
- Status Bar:** Shows "Ln 54, Col 2" and other standard status bar information.

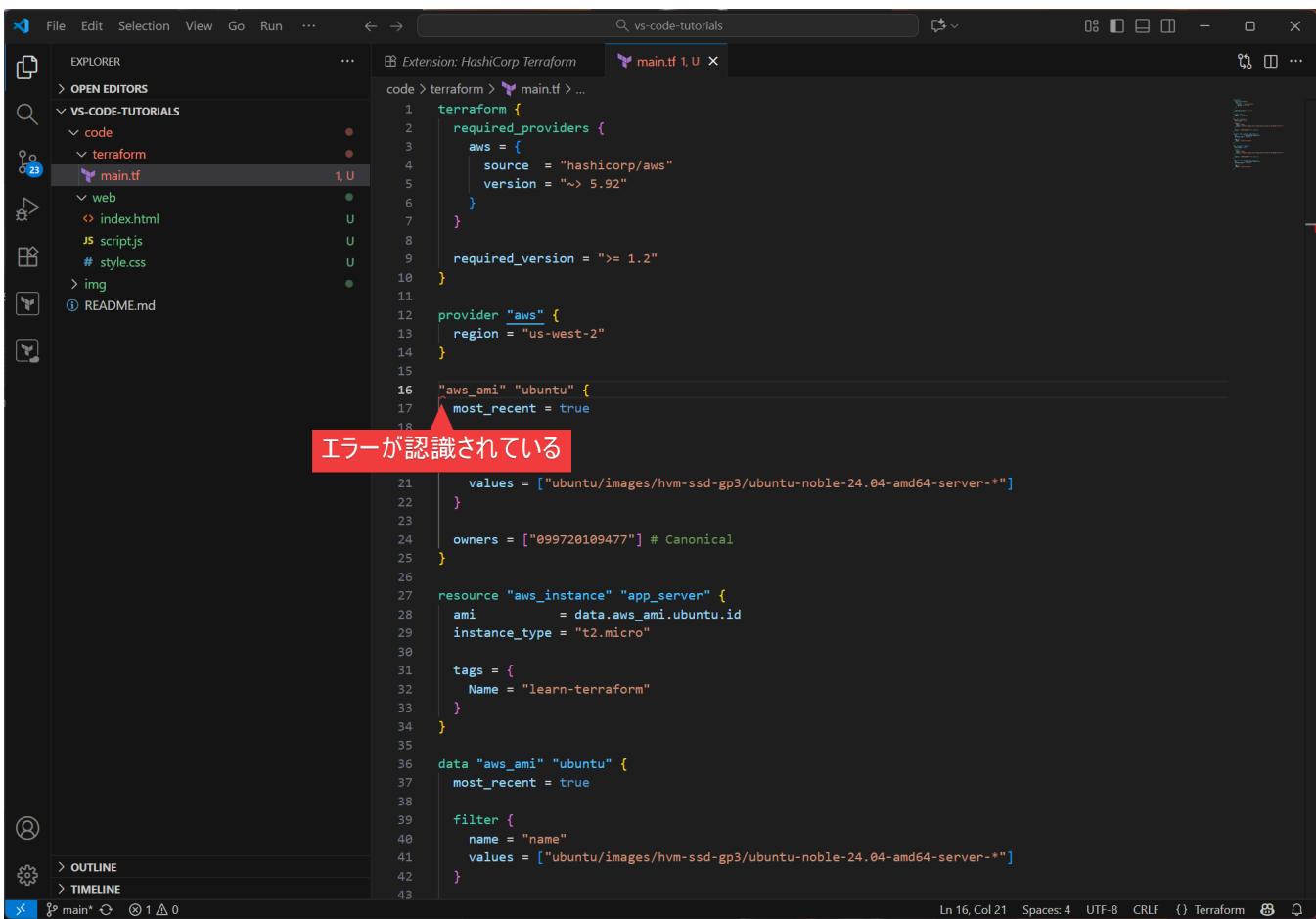
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder structure under "VS-CODE-TUTORIALS" containing "code", "terraform", and "img" folders. Inside "code", there are "index.html", "script.js", and "# style.css".
- Editor:** The "main.tf" file is open in the editor. A red box highlights the code area, and a tooltip message "エラーが認識されていない" (No error detected) is displayed above the cursor.
- Status Bar:** Shows "Ln 16, Col 1" and other standard status bar information.





```
code > terraform > main.tf > data "aws_ami" "ubuntu"
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 data "aws_ami" "ubuntu" {
17   most_recent = true
18
19   filter {
20     name = "name"
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
22   }
23
24   owners = ["099720109477"] # Canonical
25 }
26
27 resource "aws_instance" "app_server" {
28   ami           = data.aws_ami.ubuntu.id
29   instance_type = "t2.micro"
30
31   tags = {
32     Name = "learn-terraform"
33   }
34 }
35
36 data "aws_ami" "ubuntu" {
37   most_recent = true
38
39   filter {
40     name = "name"
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42   }
43 }
```



エラーが認識されている

```
code > terraform > main.tf > ...
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 "aws_ami" "ubuntu" {
17   most_recent = true
18
19   values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
20
21   owners = ["099720109477"] # Canonical
22 }
23
24 resource "aws_instance" "app_server" {
25   ami           = data.aws_ami.ubuntu.id
26   instance_type = "t2.micro"
27
28   tags = {
29     Name = "learn-terraform"
30   }
31
32 data "aws_ami" "ubuntu" {
33   most_recent = true
34
35   filter {
36     name = "name"
37     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
38   }
39 }
```

File Edit Selection View Go Run ...

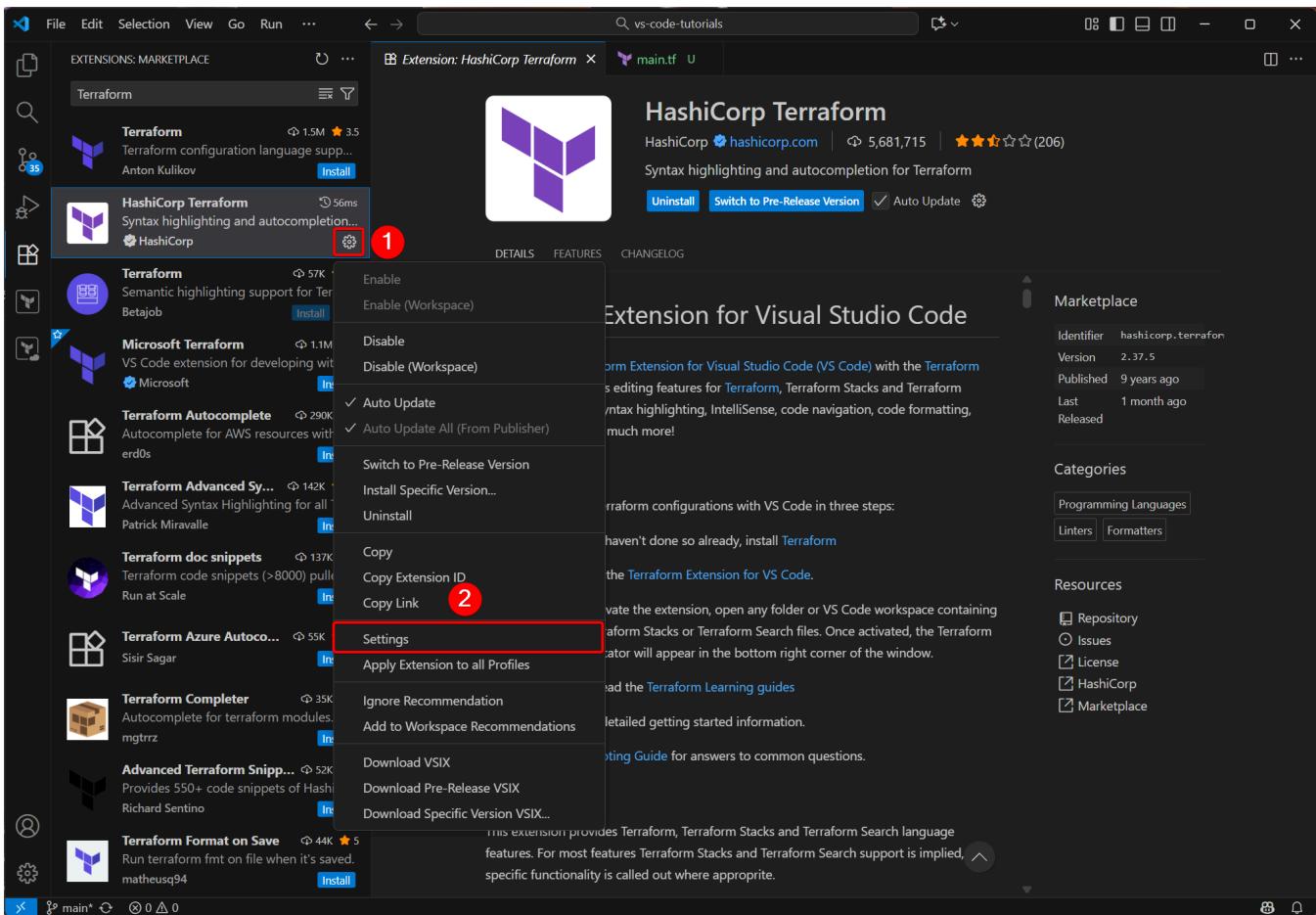
Extension: HashiCorp Terraform main.tf 1.U

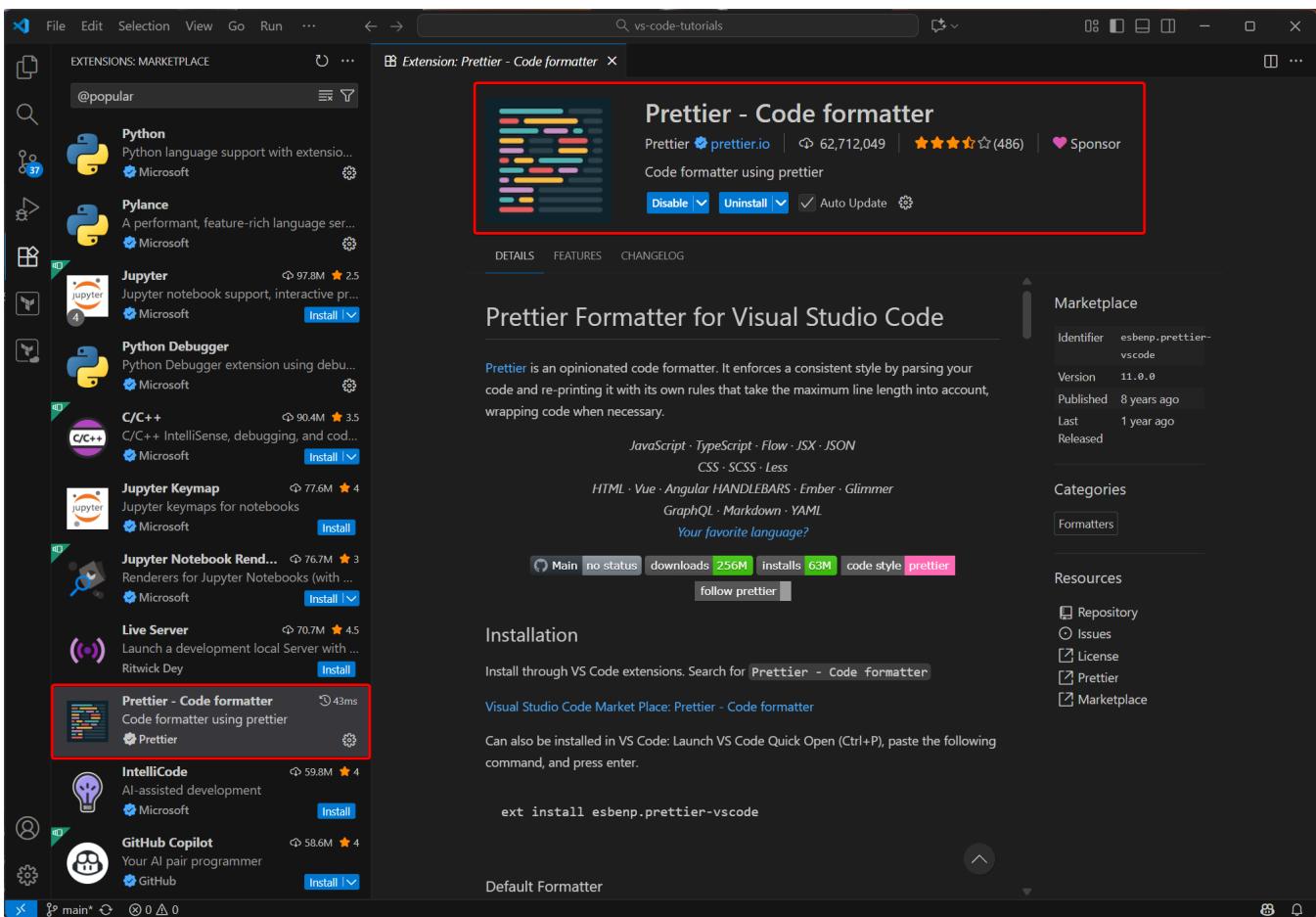
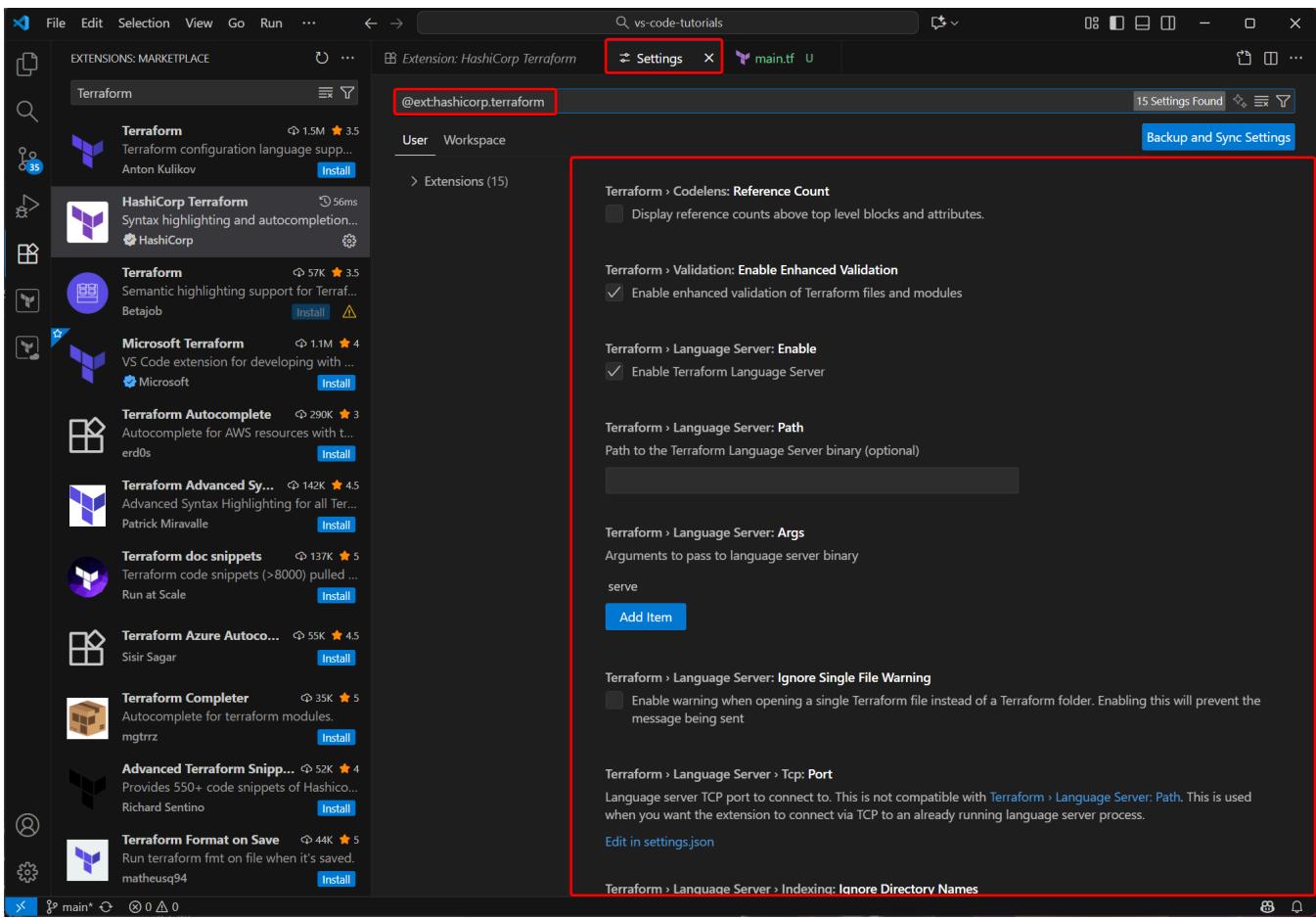
```

1  terraform {
2    required_providers {
3      aws = {
4        source  = "hashicorp/aws"
5        version = "~> 5.92"
6      }
7    }
8
9    required_version = ">= 1.2"
10  }
11
12 provider "aws" {
13
14   Invalid argument name: Argument names must not be quoted. Terraform
15   View Problem (Alt+F8) No quick fixes available
16   aws_amis "ubuntu" {
17     most_recent = true
18
19
20
21
22
23   owners = ["099720109477"] # Canonical
24
25   }
26
27   resource "aws_instance" "app_server" {
28     ami           = data.aws_ami.ubuntu.id
29     instance_type = "t2.micro"
30
31     tags = {
32       Name = "learn-terraform"
33     }
34   }
35
36   data "aws_ami" "ubuntu" {
37     most_recent = true
38
39     filter {
30       name = "name"
41       values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42     }
43

```

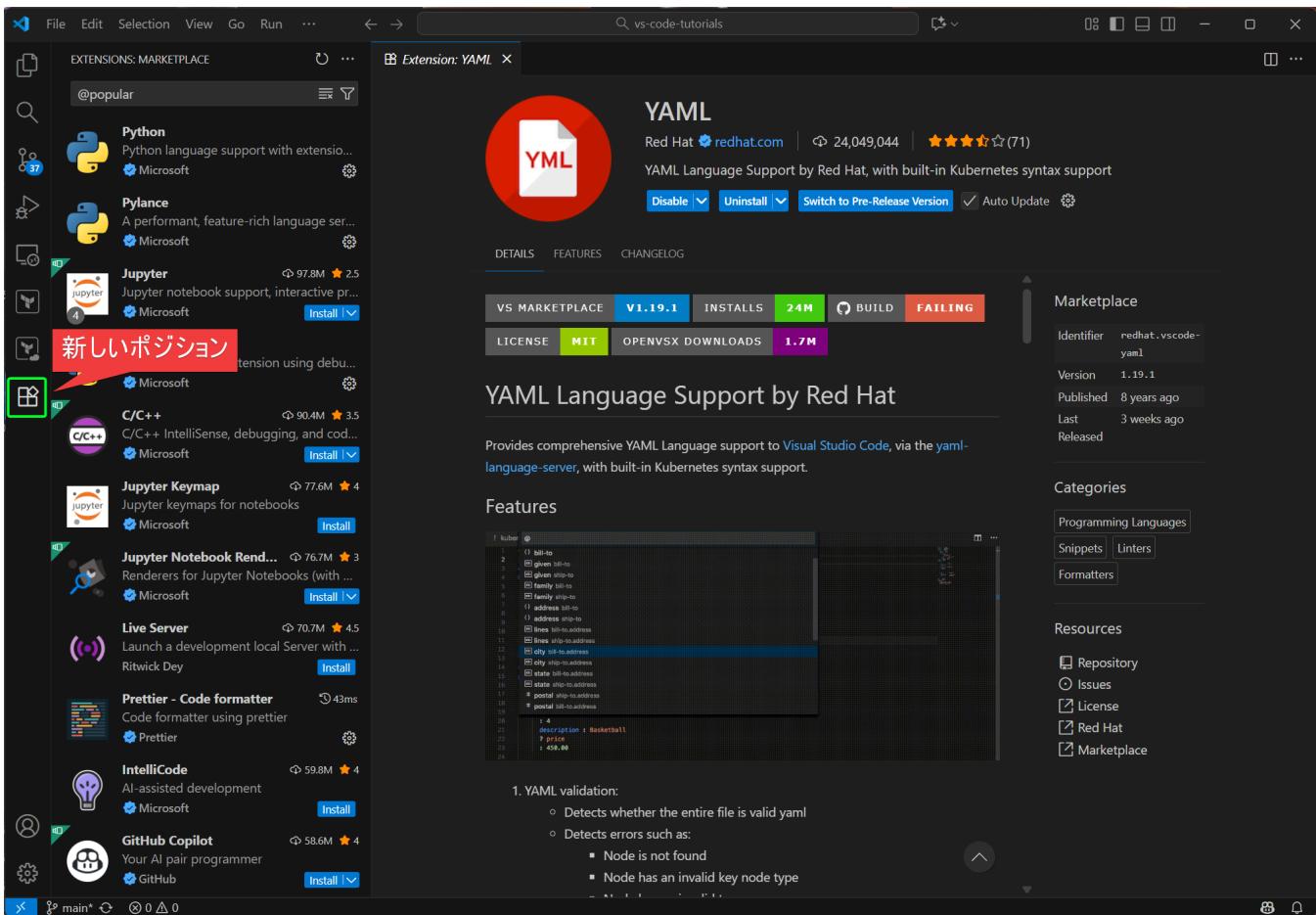
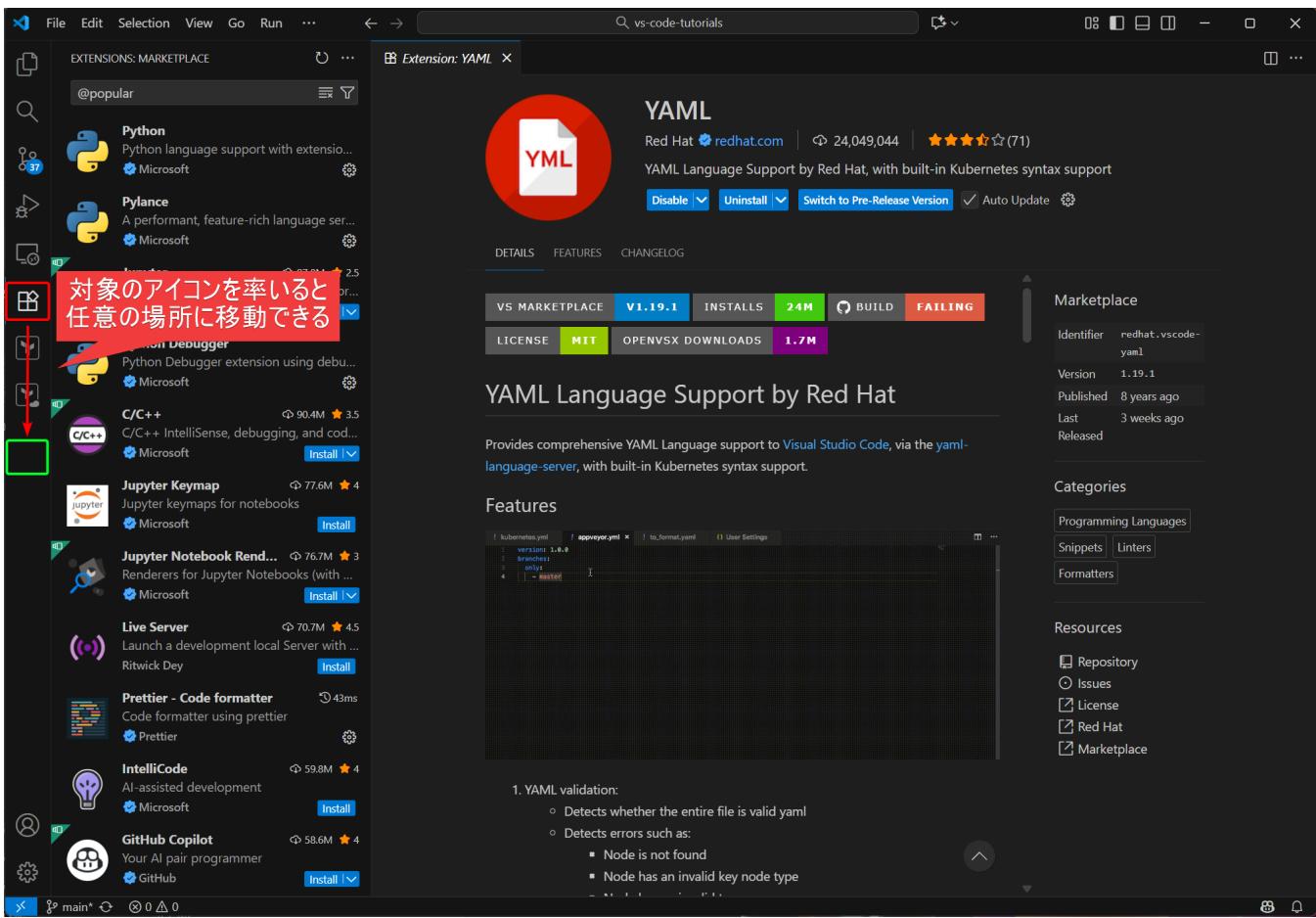
Ln 16, Col 21 Spaces: 4 UTF-8 CRLF {} Terraform





The screenshot shows the Visual Studio Code Marketplace interface. On the left, there's a sidebar with various extension categories like Java, C++, and Python. In the center, the 'Extension: WSL' page is displayed. The title 'WSL' is at the top, followed by a Microsoft logo and a star rating of 4.5 stars. Below that is a brief description: 'Open any folder in the Windows Subsystem for Linux (WSL) and take advantage of Visual ...'. There are buttons for 'Disable', 'Uninstall', and 'Auto Update'. Below the description, there's a section titled 'Visual Studio Code WSL' with a detailed description of what WSL does. To the right, there's a 'Marketplace' sidebar with details about the extension (Identifier: ms-vscode-remote.remote-wsl, Version: 0.104.3, Published: 6 years ago, Last Released: 1 month ago), a 'Categories' section (Other), and a 'Resources' section (Repository, Issues, License, Microsoft, Marketplace).

This screenshot shows the Visual Studio Code Marketplace interface again, but this time the 'Extension: Remote - SSH' page is highlighted. The title 'Remote - SSH' is at the top, followed by a Microsoft logo and a star rating of 4.5 stars. Below that is a brief description: 'Open any folder on a remote machine using SSH and take advantage of VS Code's full fea...'. There are buttons for 'Disable', 'Uninstall', 'Switch to Pre-Release Version', and 'Auto Update'. Below the description, there's a section titled 'Visual Studio Code Remote - SSH' with a detailed description of what the extension does. To the right, there's a 'Marketplace' sidebar with details about the extension (Identifier: ms-vscode-remote.remote-ssh, Version: 0.120.0, Published: 6 years ago, Last Released: 1 month ago), a 'Categories' section (Other), and a 'Resources' section (Repository, Issues, License, Microsoft, Marketplace). At the bottom, a screenshot of the VS Code interface showing a terminal window is visible.



3. Settings

This screenshot shows the VS Code interface with the 'Settings' menu open. The menu is located in the bottom-left corner of the workspace. A red circle labeled '1' highlights the gear icon in the bottom-left corner of the window. A red circle labeled '2' highlights the 'Settings' option in the menu.

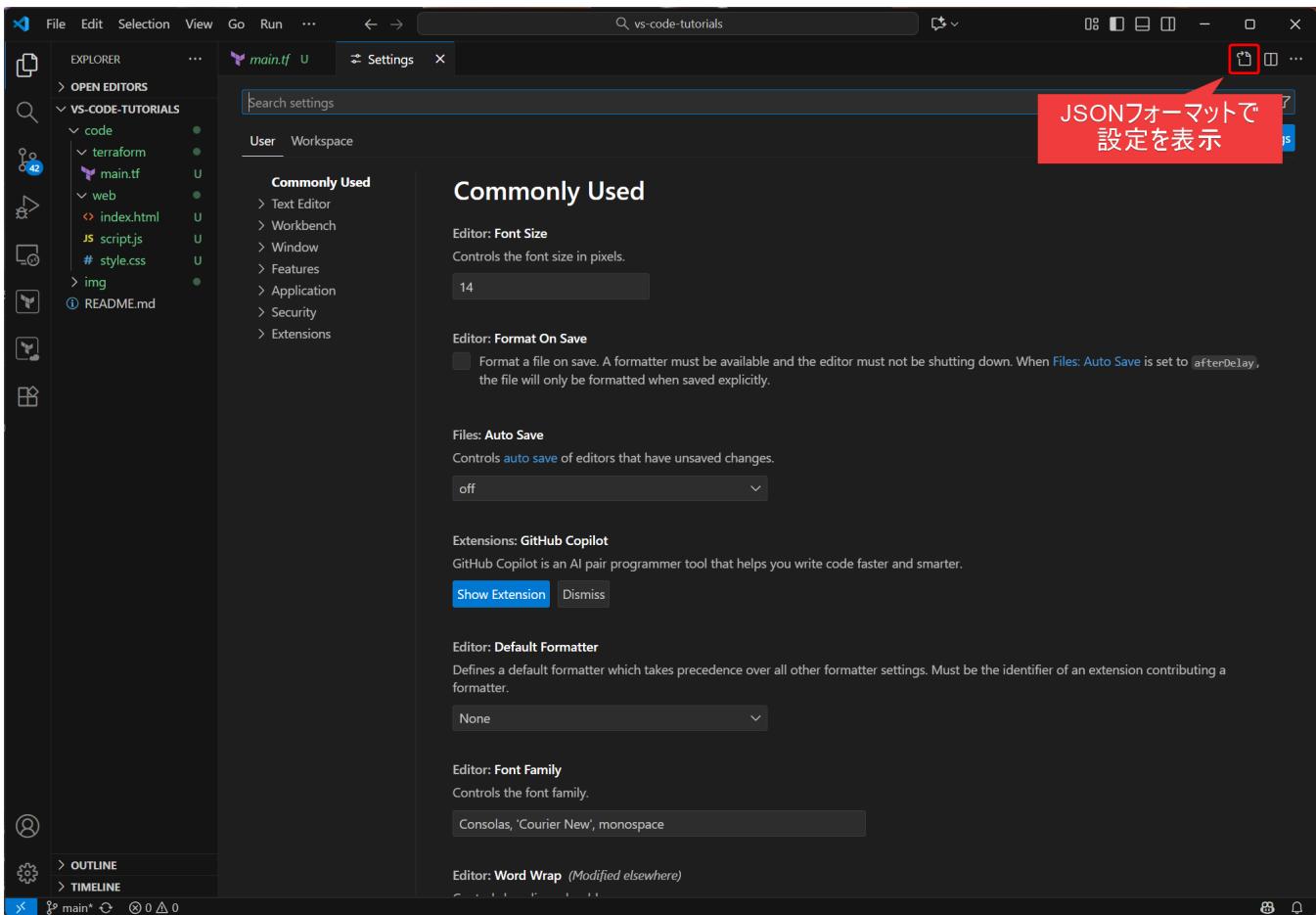
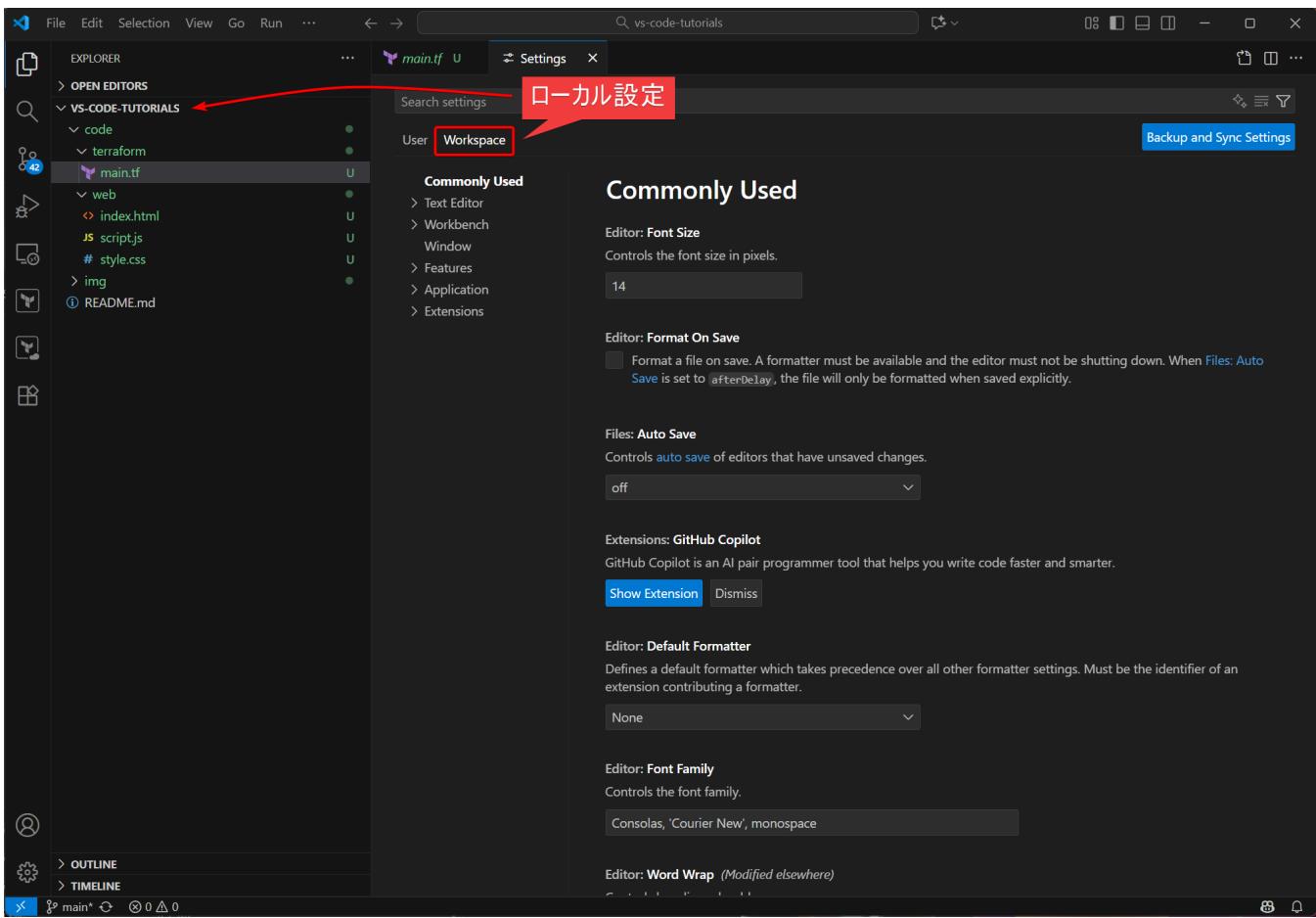
The main editor area displays a Terraform configuration file named 'main.tf'. The code defines an AWS instance resource using an Ubuntu AMI and tags it with 'Name = "learn-terraform"'.

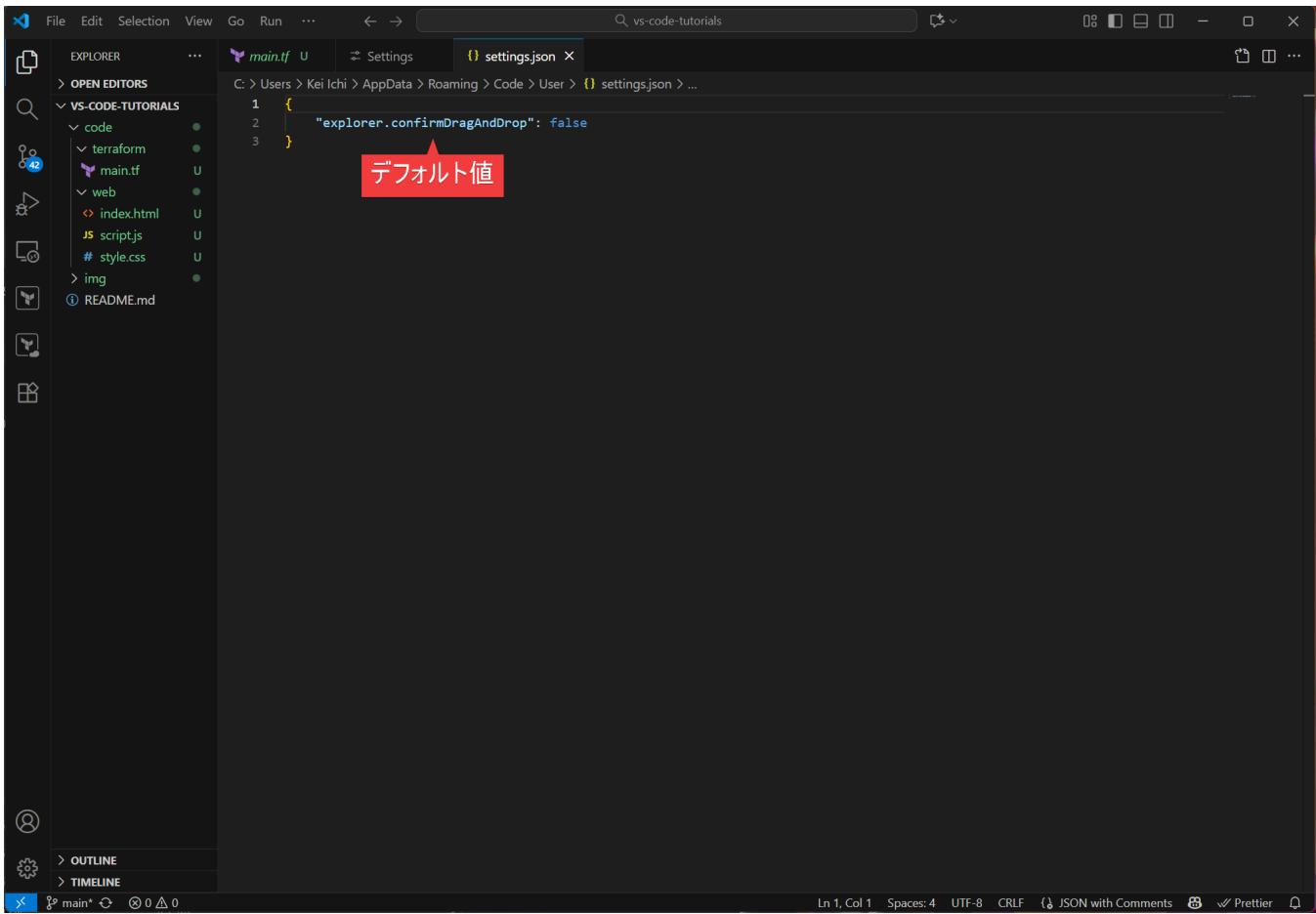
```
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 data "aws_ami" "ubuntu" {
17   most_recent = true
18
19   filter {
20     name = "name"
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
22   }
23
24   owners = ["099720109477"] # Canonical
25 }
26
27 resource "aws_instance" "app_server" {
28   ami           = data.aws_ami.ubuntu.id
29   instance_type = "t2.micro"
30
31   tags = {
32     Name = "learn-terraform"
33   }
34 }
35
36 data "aws_ami" "ubuntu" {
37   most_recent = true
38
39   filter {
40     name = "name"
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42 }
```

This screenshot shows the VS Code interface with the 'Settings' page open. A red box labeled 'グローバル設定' (Global Settings) points to the 'User' tab in the settings sidebar.

The main editor area displays the same 'main.tf' Terraform configuration file as the previous screenshot.

The settings page includes sections for 'Commonly Used' settings like Editor Font Size (set to 14), Editor Format On Save (disabled), Files Auto Save (set to off), Extensions GitHub Copilot (Show Extension), Editor Default Formatter (set to None), Editor Font Family (set to Consolas, 'Courier New', monospace), and Editor Word Wrap (Modified elsewhere).





File Edit Selection View Go Run ...

EXPLORER OPEN EDITORS VS-CODE-TUTORIALS

code

terraform

main.tf

web

index.html

script.js

style.css

img

README.md

main.tf

Settings

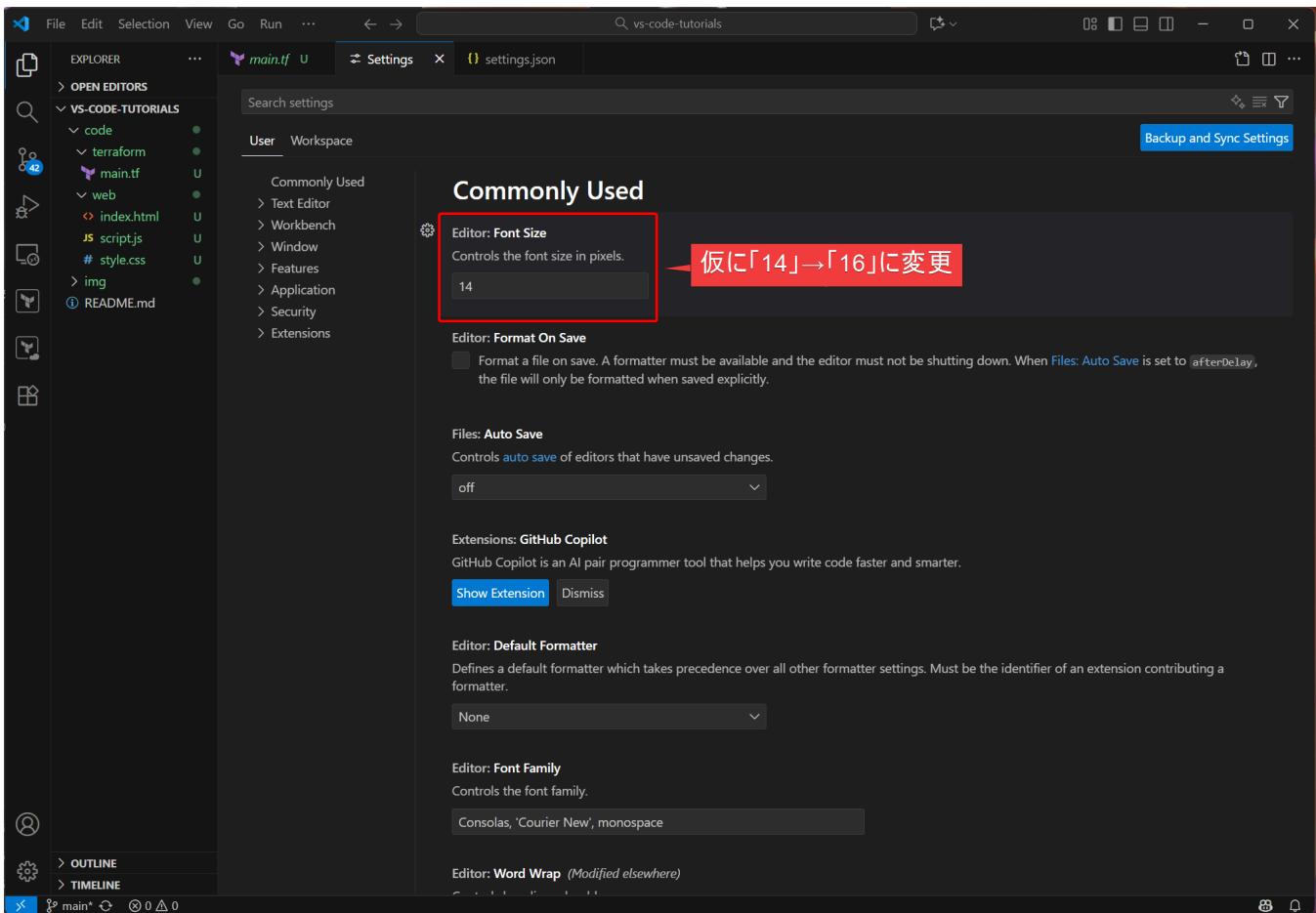
settings.json

```
C: > Users > Kei Ichi > AppData > Roaming > Code > User > settings.json > ...
```

```
1 {  
2   "explorer.confirmDragAndDrop": false  
3 }
```

デフォルト値

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JSON with Comments Prettier



File Edit Selection View Go Run ...

EXPLORER OPEN EDITORS VS-CODE-TUTORIALS

code

terraform

main.tf

web

index.html

script.js

style.css

img

README.md

main.tf

Settings

settings.json

Search settings

User Workspace

Commonly Used

Editor: Font Size

Controls the font size in pixels.

14

Backup and Sync Settings

Editor: Format On Save

Format a file on save. A formatter must be available and the editor must not be shutting down. When [Files: Auto Save](#) is set to `afterDelay`, the file will only be formatted when saved explicitly.

Files: Auto Save

Controls [auto save](#) of editors that have unsaved changes.

off

Extensions: GitHub Copilot

GitHub Copilot is an AI pair programmer tool that helps you write code faster and smarter.

Show Extension Dismiss

Editor: Default Formatter

Defines a default formatter which takes precedence over all other formatter settings. Must be the identifier of an extension contributing a formatter.

None

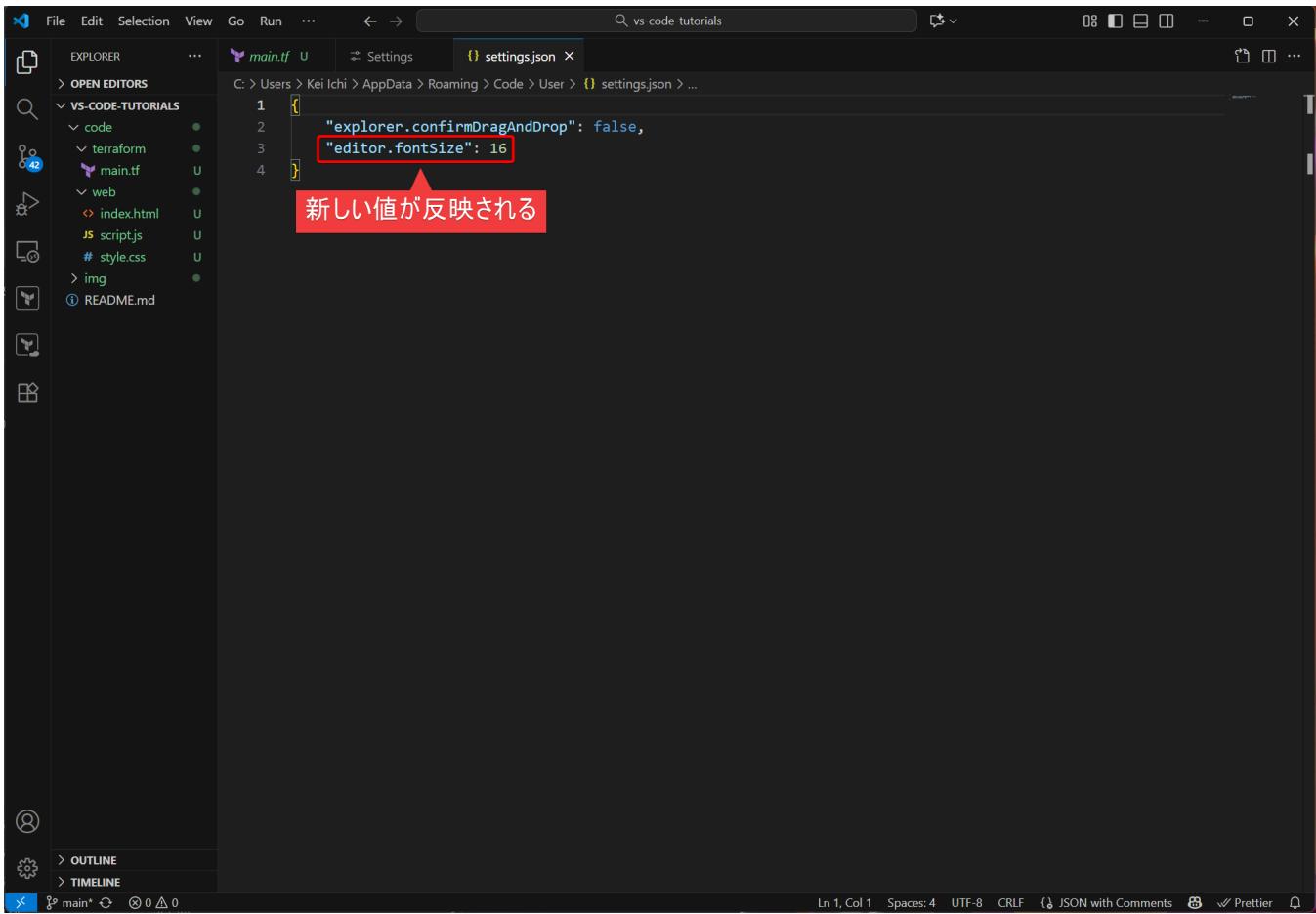
Editor: Font Family

Controls the font family.

Consolas, 'Courier New', monospace

Editor: Word Wrap (Modified elsewhere)

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JSON with Comments Prettier



File Edit Selection View Go Run ...

EXPLORER OPEN EDITORS VS-CODE-TUTORIALS

code main.tf terraform web index.html script.js style.css img README.md

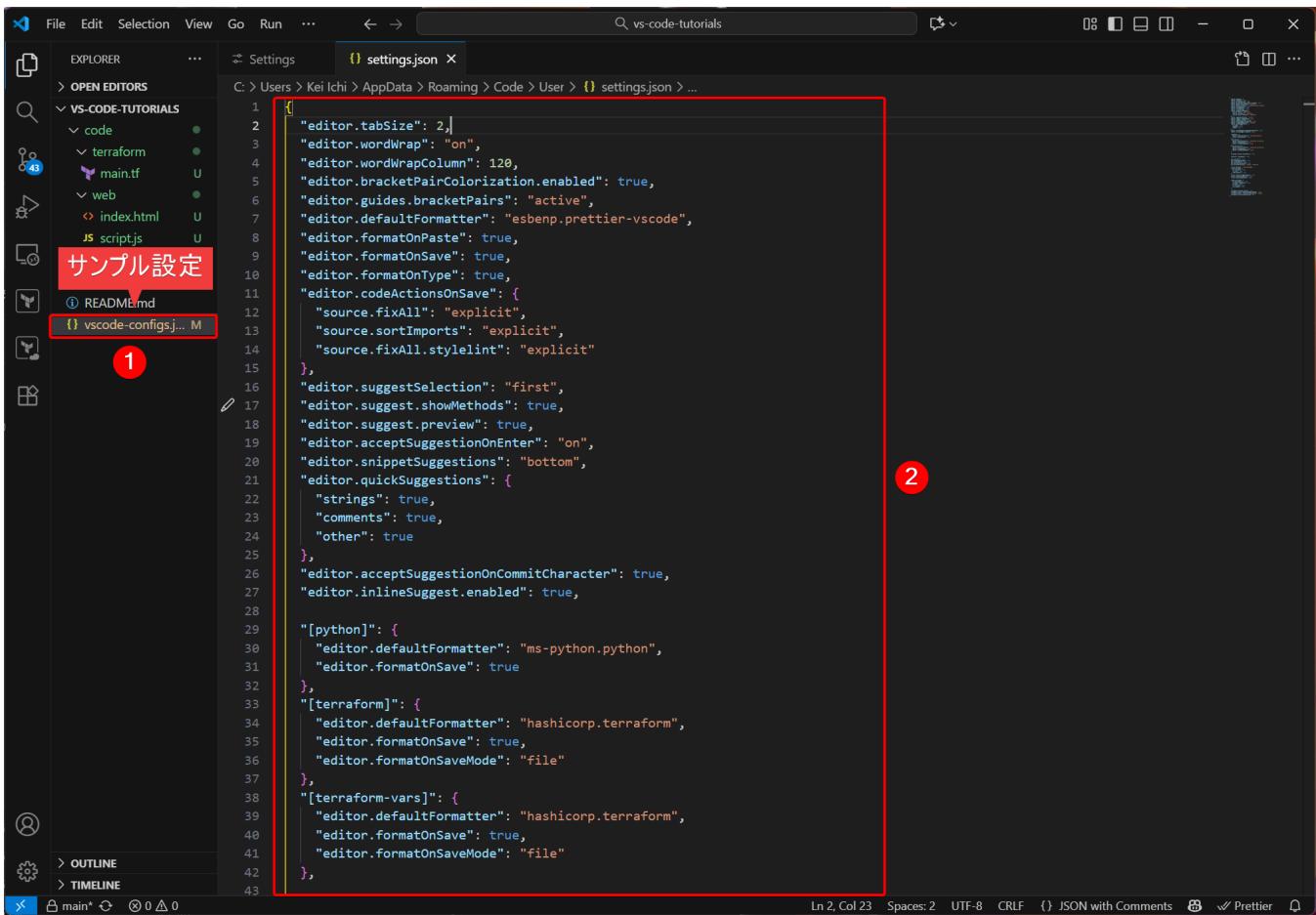
main* Settings settings.json

```
C:\> Users > Kei Ichi > AppData > Roaming > Code > User > settings.json > ...
```

```
1 [ "explorer.confirmDragAndDrop": false, "editor.fontSize": 16 }
```

新しい値が反映される

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF {} JSON with Comments ✅ Prettier



File Edit Selection View Go Run ...

EXPLORER OPEN EDITORS VS-CODE-TUTORIALS

code main.tf terraform web index.html script.js README.md vscode-configs.json

サンプル設定 1

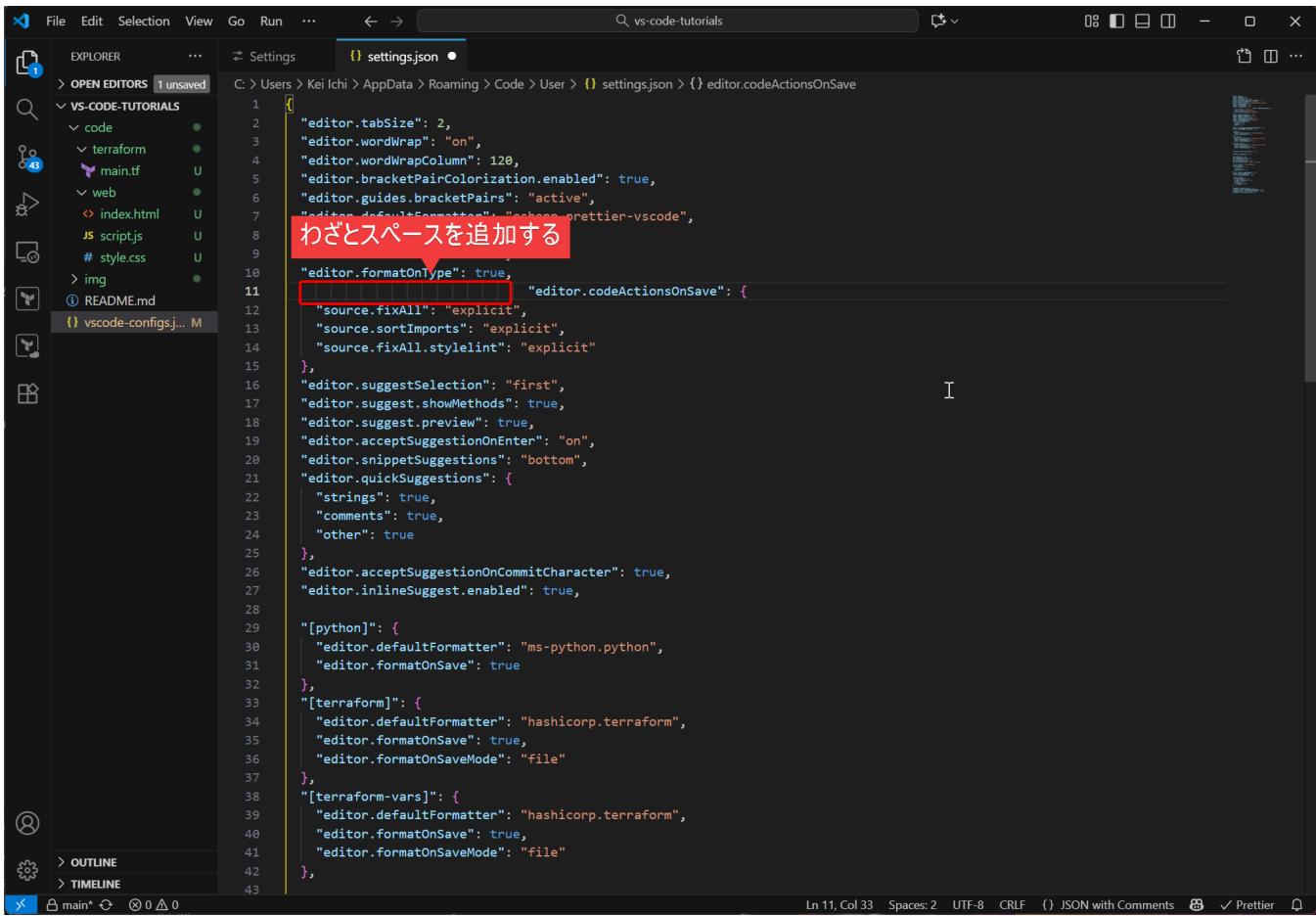
2

Settings settings.json

```
C:\> Users > Kei Ichi > AppData > Roaming > Code > User > settings.json > ...
```

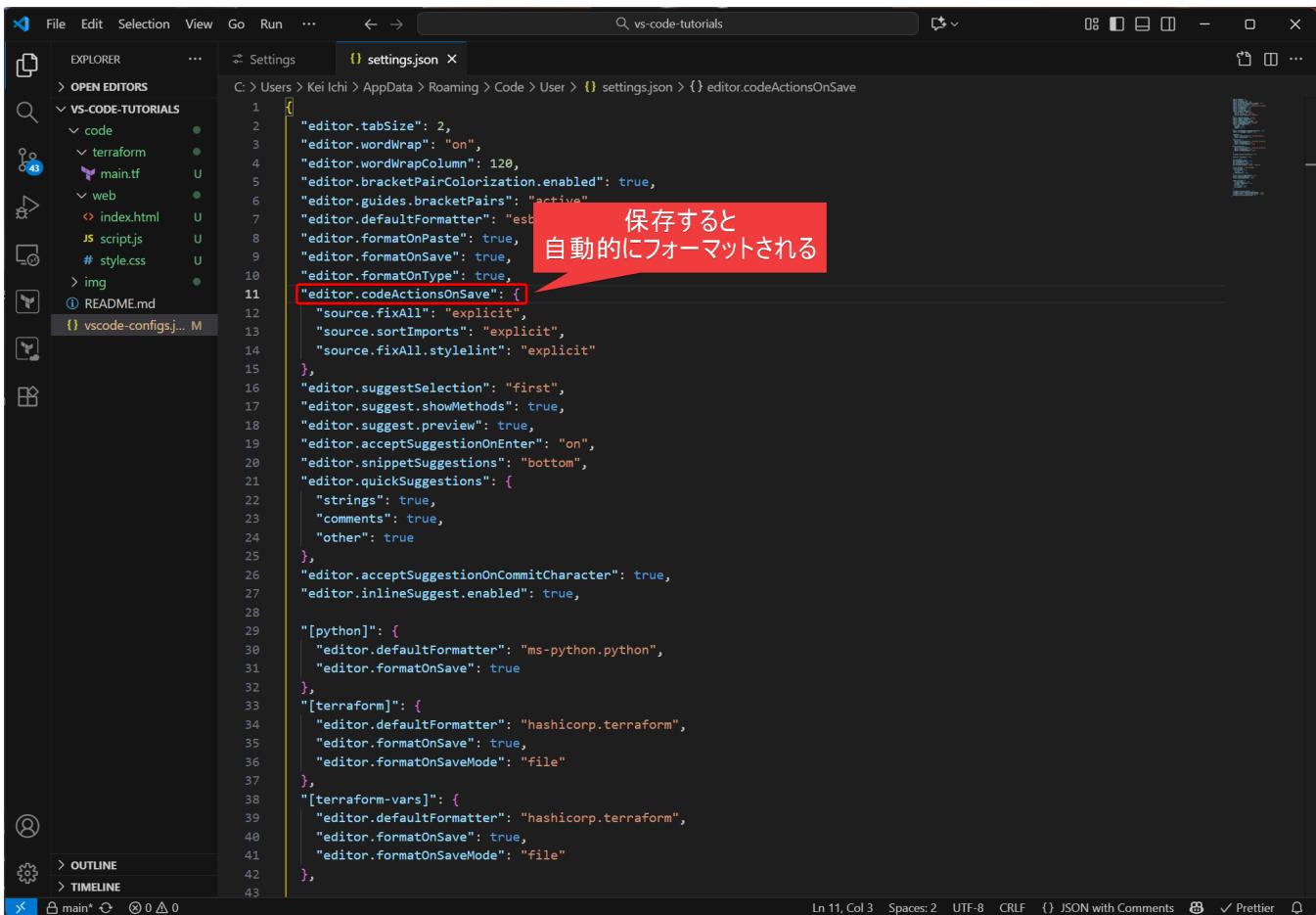
```
1 [ "editor.tabSize": 2, "editor.wordWrap": "on", "editor.wordWrapColumn": 120, "editor.bracketPairColorization.enabled": true, "editor.guides.bracketPairs": "active", "editor.defaultFormatter": "esbenp.prettier-vscode", "editor.formatOnPaste": true, "editor.formatOnSave": true, "editor.formatOnType": true, "editor.codeActionsOnSave": { "source.fixAll": "explicit", "source.sortImports": "explicit", "source.fixAll.stylelint": "explicit" }, "editor.suggestSelection": "first", "editor.suggest.showMethods": true, "editor.suggest.preview": true, "editor.acceptSuggestionOnEnter": "on", "editor.snippetSuggestions": "bottom", "editor.quickSuggestions": { "strings": true, "comments": true, "other": true }, "editor.acceptSuggestionOnCommitCharacter": true, "editor.inlineSuggest.enabled": true, "[python)": { "editor.defaultFormatter": "ms-python.python", "editor.formatOnSave": true }, "[terraform)": { "editor.defaultFormatter": "hashicorp.terraform", "editor.formatOnSave": true, "editor.formatOnSaveMode": "file" }, "[terraform-vars)": { "editor.defaultFormatter": "hashicorp.terraform", "editor.formatOnSave": true, "editor.formatOnSaveMode": "file" } ]
```

Ln 2, Col 23 Spaces: 2 UTF-8 CRLF {} JSON with Comments ✅ Prettier



The screenshot shows the VS Code interface with the settings.json file open in the editor. A red box highlights the line "editor.formatOnType": true, which is part of the "editor.codeActionsOnSave" configuration. The status bar at the bottom indicates "Ln 11, Col 33".

```
C:\> Users > Kei Ichi > AppData > Roaming > Code > User > settings.json > {} editor.codeActionsOnSave
1 {
2   "editor.tabSize": 2,
3   "editor.wordWrap": "on",
4   "editor.wordWrapColumn": 120,
5   "editor.bracketPairColorization.enabled": true,
6   "editor.guides.bracketPairs": "active",
7   "editor.defaultFormatter": "hashicorp.prettier-vscode",
8   "editor.formatOnType": true, // This line is highlighted by a red box
9   "editor.codeActionsOnSave": {
10     "source.fixAll": "explicit",
11     "source.sortImports": "explicit",
12     "source.fixAll.stylelint": "explicit"
13   },
14   "editor.suggestSelection": "first",
15   "editor.suggest.showMethods": true,
16   "editor.suggest.preview": true,
17   "editor.acceptSuggestionOnEnter": "on",
18   "editor.snippetSuggestions": "bottom",
19   "editor.quickSuggestions": {
20     "strings": true,
21     "comments": true,
22     "other": true
23   },
24   "editor.acceptSuggestionOnCommitCharacter": true,
25   "editor.inlineSuggest.enabled": true,
26
27 "[python]": {
28   "editor.defaultFormatter": "ms-python.python",
29   "editor.formatOnSave": true
30 },
31 "[terraform)": {
32   "editor.defaultFormatter": "hashicorp.terraform",
33   "editor.formatOnSave": true,
34   "editor.formatOnSaveMode": "file"
35 },
36 "[terraform-vars)": {
37   "editor.defaultFormatter": "hashicorp.terraform",
38   "editor.formatOnSave": true,
39   "editor.formatOnSaveMode": "file"
40 },
41
42 },
```



The screenshot shows the VS Code interface with the settings.json file open in the editor. A red box highlights the line "editor.codeActionsOnSave": {}, which is part of the "editor.codeActionsOnSave" configuration. The status bar at the bottom indicates "Ln 11, Col 33".

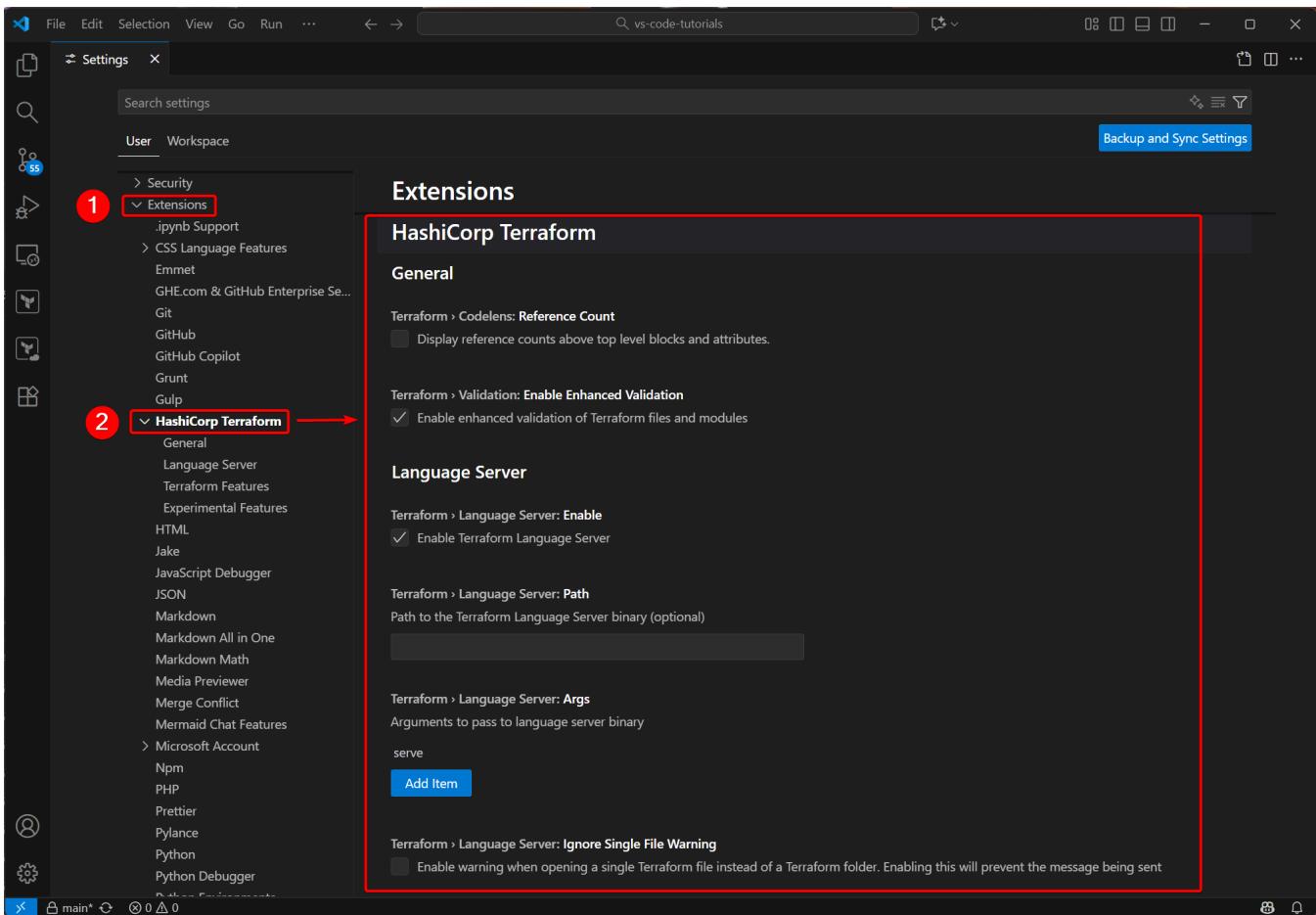
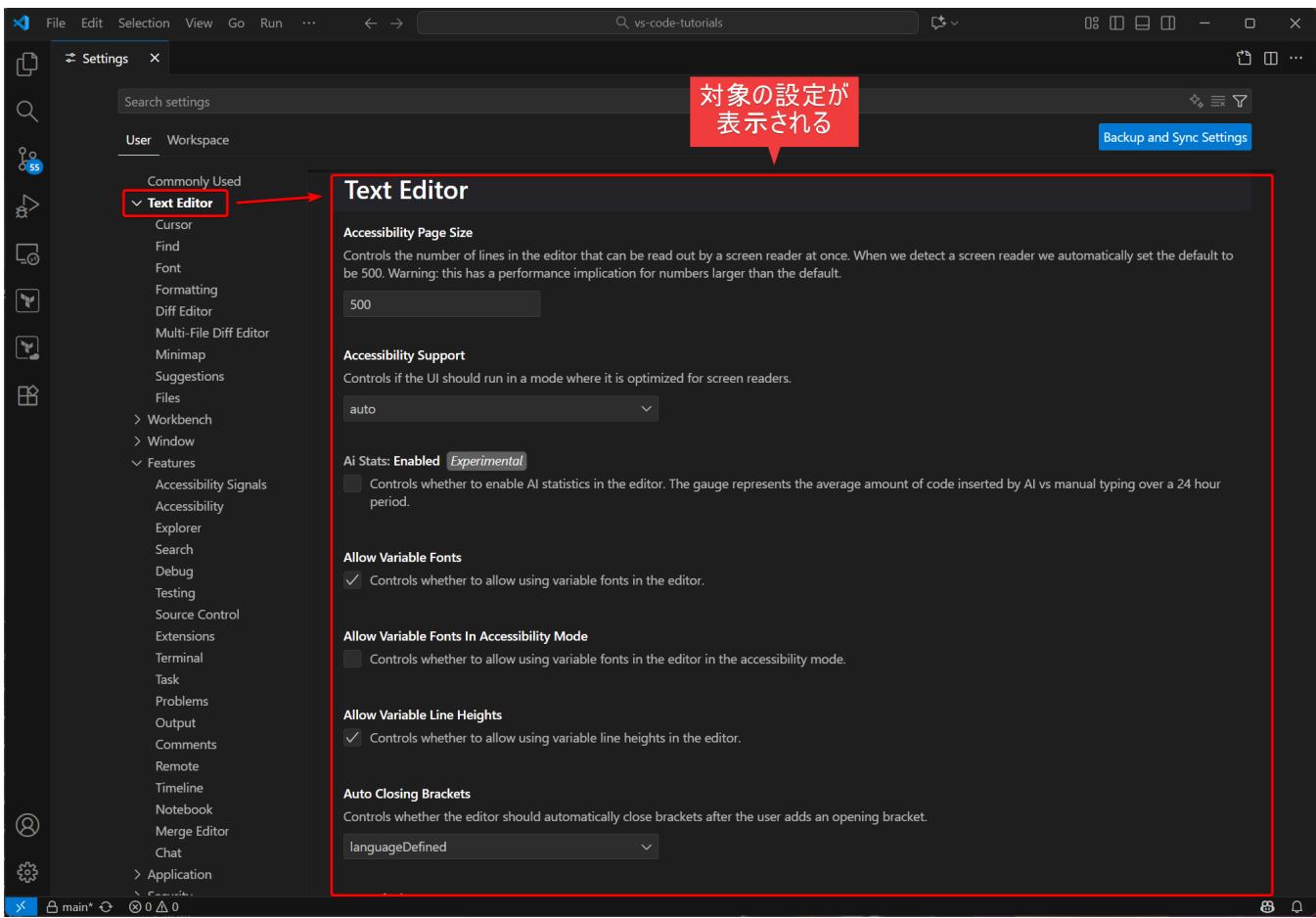
```
C:\> Users > Kei Ichi > AppData > Roaming > Code > User > settings.json > {} editor.codeActionsOnSave
1 {
2   "editor.tabSize": 2,
3   "editor.wordWrap": "on",
4   "editor.wordWrapColumn": 120,
5   "editor.bracketPairColorization.enabled": true,
6   "editor.guides.bracketPairs": "active",
7   "editor.defaultFormatter": "est",
8   "editor.formatOnPaste": true,
9   "editor.formatOnSave": true,
10  "editor.formatOnType": true,
11  "editor.codeActionsOnSave": {} // This line is highlighted by a red box
12  "source.fixAll": "explicit",
13  "source.sortImports": "explicit",
14  "source.fixAll.stylelint": "explicit"
15  },
16  "editor.suggestSelection": "first",
17  "editor.suggest.showMethods": true,
18  "editor.suggest.preview": true,
19  "editor.acceptSuggestionOnEnter": "on",
20  "editor.snippetSuggestions": "bottom",
21  "editor.quickSuggestions": {
22    "strings": true,
23    "comments": true,
24    "other": true
25  },
26  "editor.acceptSuggestionOnCommitCharacter": true,
27  "editor.inlineSuggest.enabled": true,
28
29 "[python]": {
30   "editor.defaultFormatter": "ms-python.python",
31   "editor.formatOnSave": true
32 },
33 "[terraform)": {
34   "editor.defaultFormatter": "hashicorp.terraform",
35   "editor.formatOnSave": true,
36   "editor.formatOnSaveMode": "file"
37 },
38 "[terraform-vars)": {
39   "editor.defaultFormatter": "hashicorp.terraform",
40   "editor.formatOnSave": true,
41   "editor.formatOnSaveMode": "file"
42 },
```

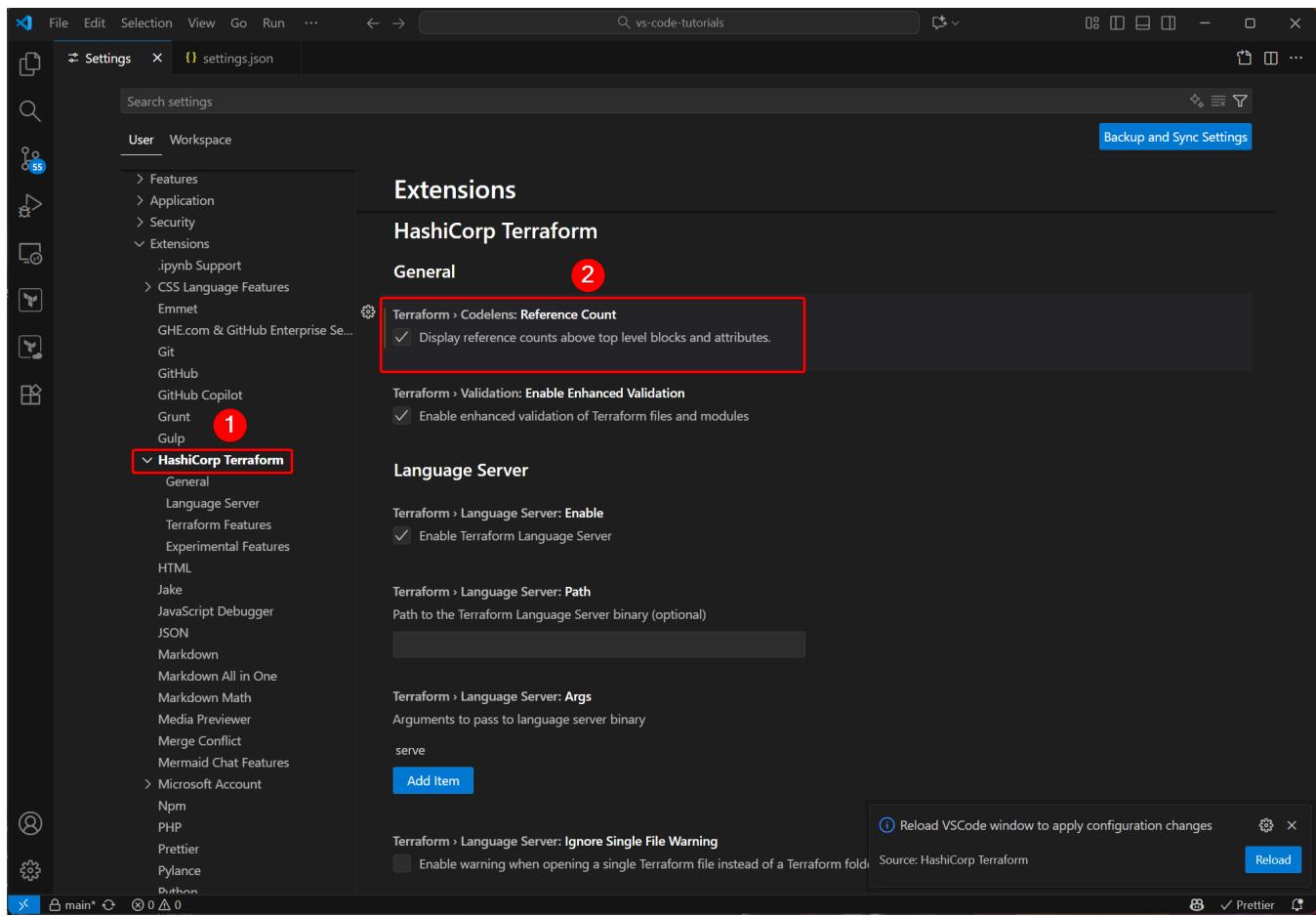
The screenshot shows the VS Code interface with the Terraform main.tf file open. A red box highlights the line 'data "aws_ami" "ubuntu" [', with a callout bubble pointing to it containing the text '同じくスペースをたくさん追加する' (Add many spaces like this). The code completion dropdown is visible above the line.

```
1 terraform {  
2   required_providers {  
3     aws = {  
4       source  = "hashicorp/aws"  
5       version = "~> 5.92"  
6     }  
7   }  
8  
9   required_version = ">= 1.2"  
10 }  
11  
12 provider "aws" {  
13   region = "us-west-2"  
14 }  
15  
16 data "aws_ami" "ubuntu" [  
17   most_recent = true  
18  
19   filter {  
20     name  = "name"  
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]  
22   }  
23  
24   owners = ["099720109477"] # Canonical  
25 }  
26  
27 resource "aws_instance" "app_server" {  
28   ami          = data.aws_ami.ubuntu.id  
29   instance_type = "t2.micro"  
30  
31   tags = {  
32     Name = "learn-terraform"  
33   }  
34 }  
35  
36 data "aws_ami" "ubuntu" {  
37   most_recent = true  
38  
39   filter {  
40     name  = "name"  
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]  
42 }
```

The screenshot shows the VS Code interface with the Terraform main.tf file open. A red box highlights the line 'data "aws_ami" "ubuntu" [', with a callout bubble pointing to it containing the text '保存するだけで自動的に形を整える' (Formatting happens automatically when you save). The code completion dropdown is visible above the line.

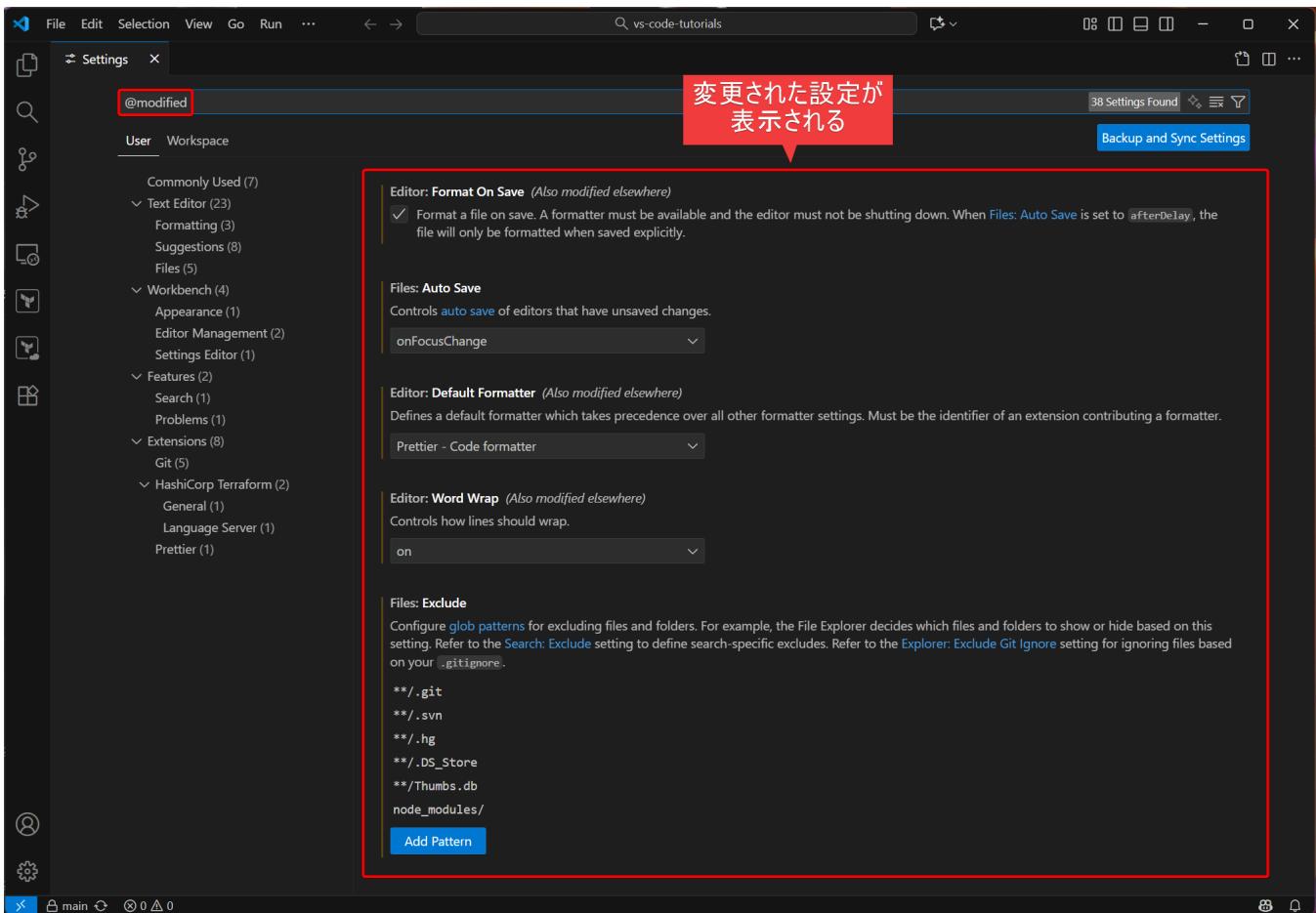
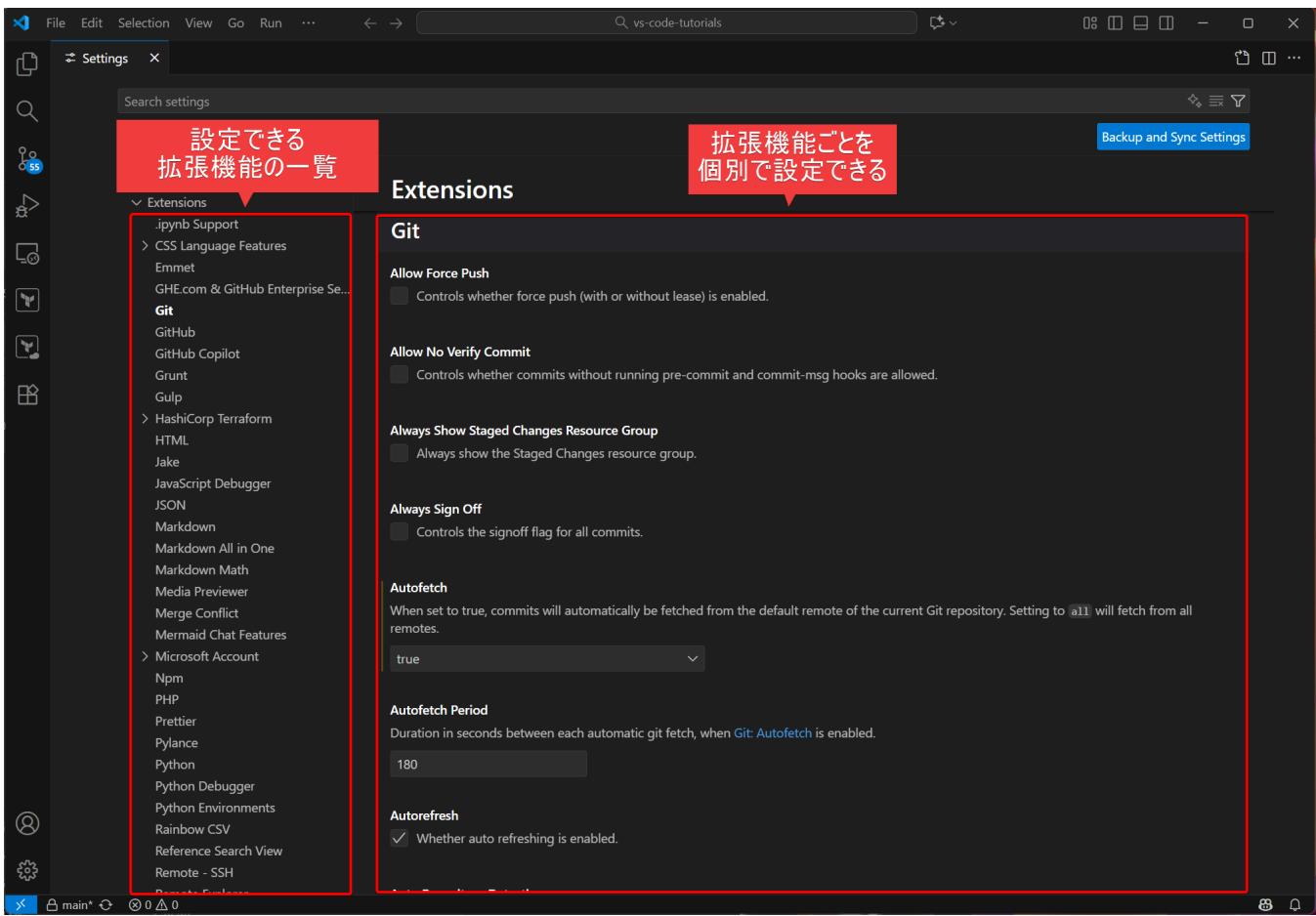
```
1 terraform {  
2   required_providers {  
3     aws = {  
4       source  = "hashicorp/aws"  
5       version = "~> 5.92"  
6     }  
7   }  
8  
9   required_version = ">= 1.2"  
10 }  
11  
12 provider "aws" {  
13   region = "us-west-2"  
14 }  
15  
16 data "aws_ami" "ubuntu" {  
17   most_recent = true  
18  
19   filter {  
20     name  = "name"  
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]  
22   }  
23  
24   owners = ["099720109477"] # Canonical  
25 }  
26  
27 resource "aws_instance" "app_server" {  
28   ami          = data.aws_ami.ubuntu.id  
29   instance_type = "t2.micro"  
30  
31   tags = {  
32     Name = "learn-terraform"  
33   }  
34 }  
35  
36 data "aws_ami" "ubuntu" {  
37   most_recent = true  
38  
39   filter {  
40     name  = "name"  
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]  
42 }
```

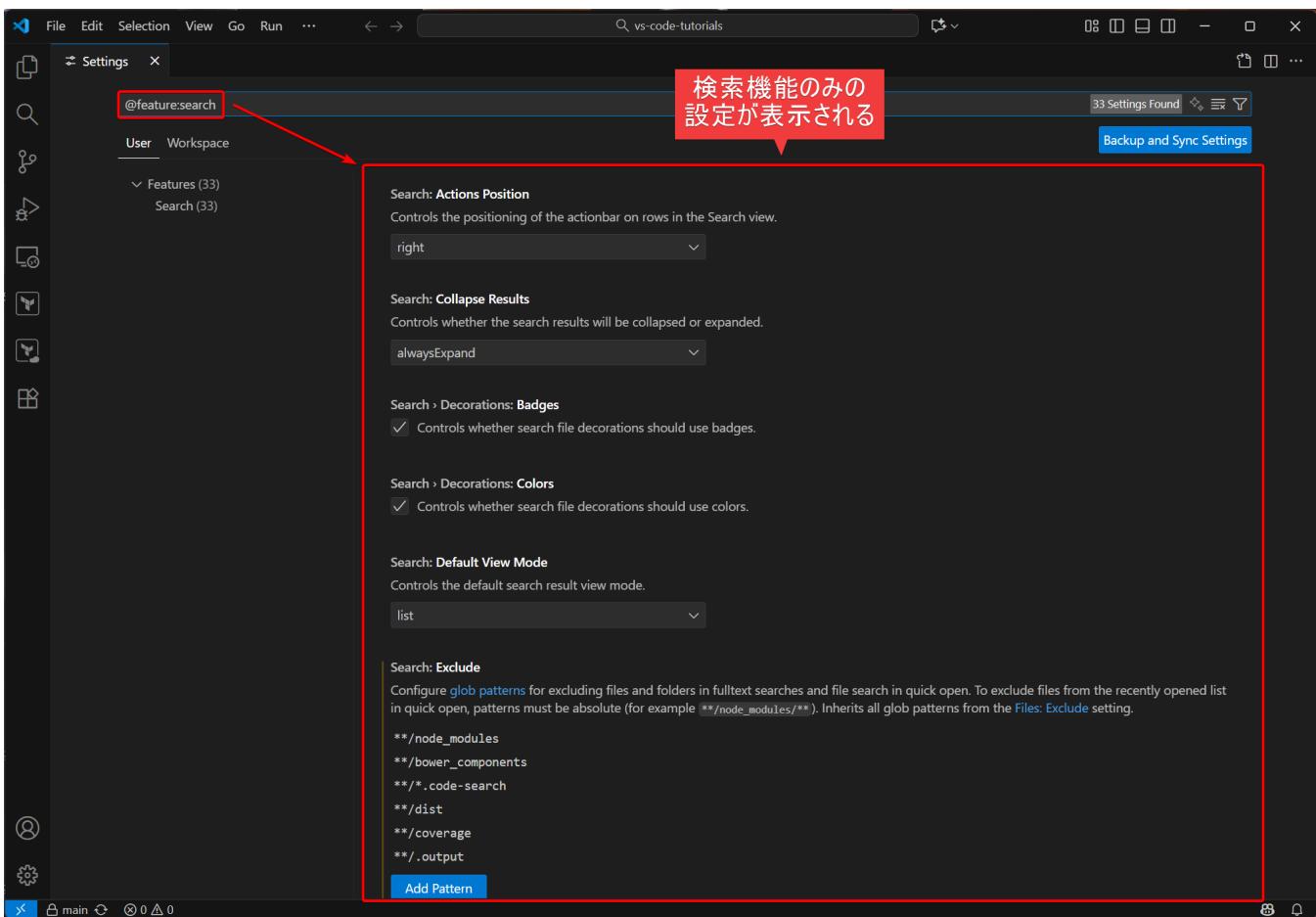
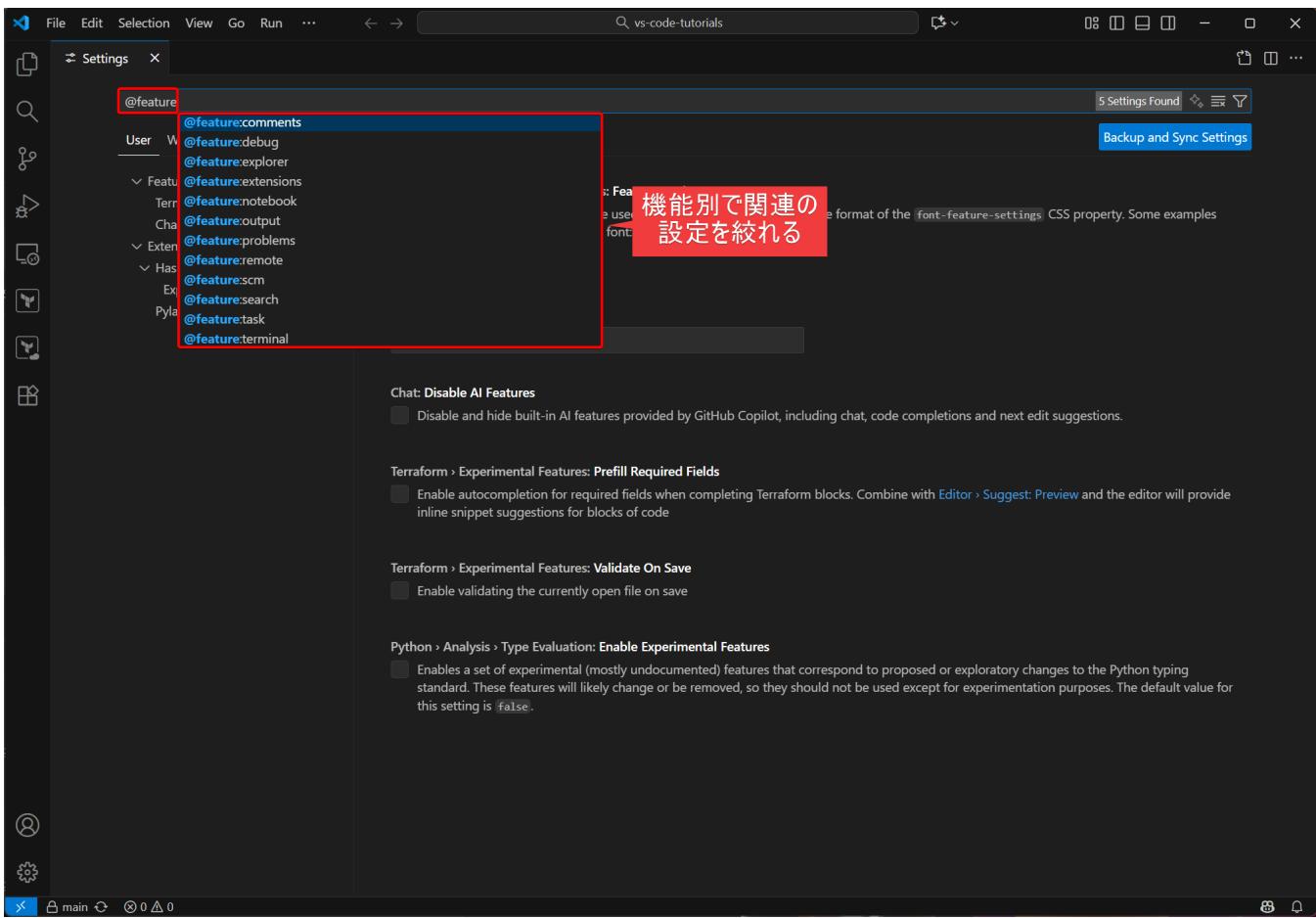


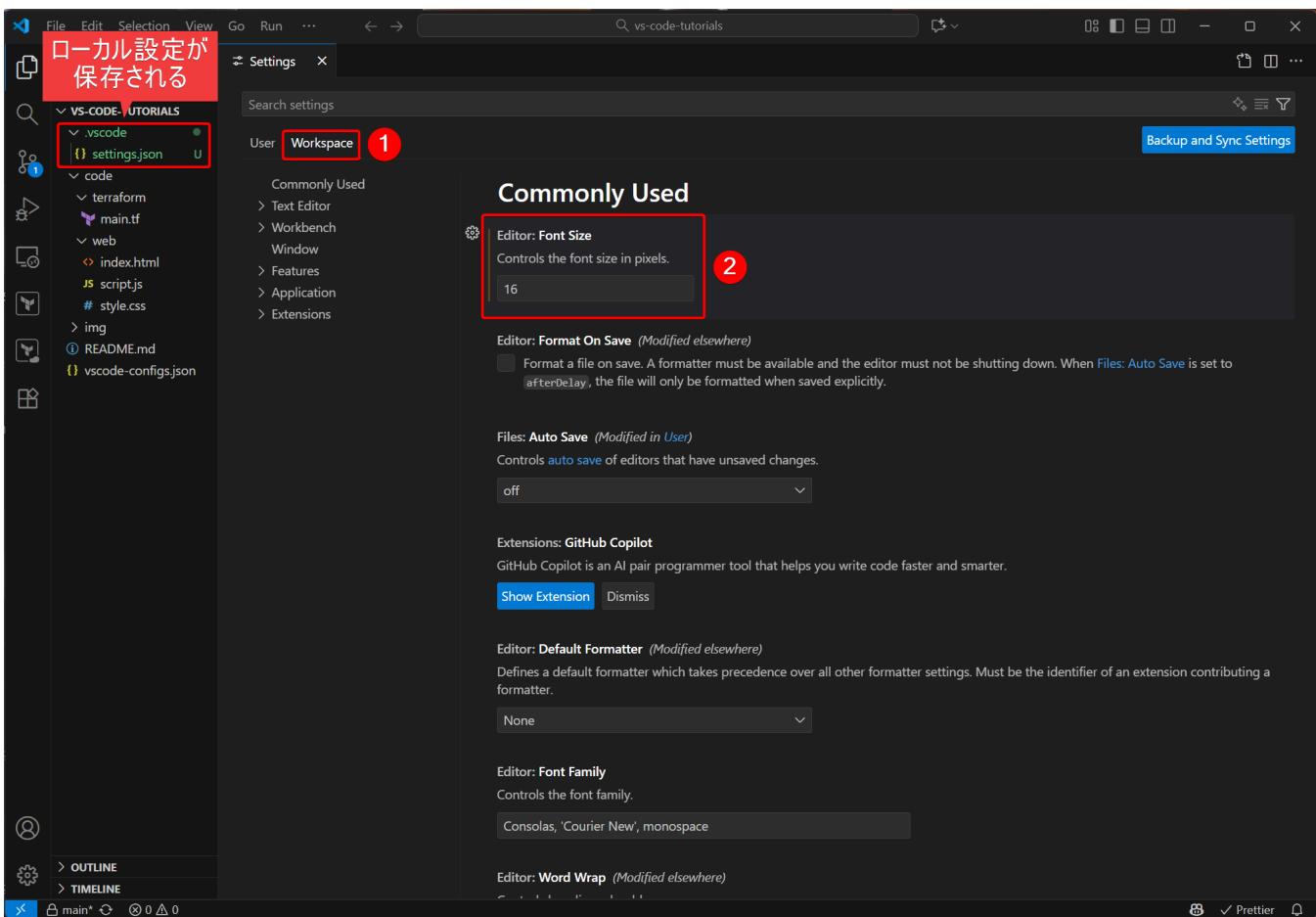
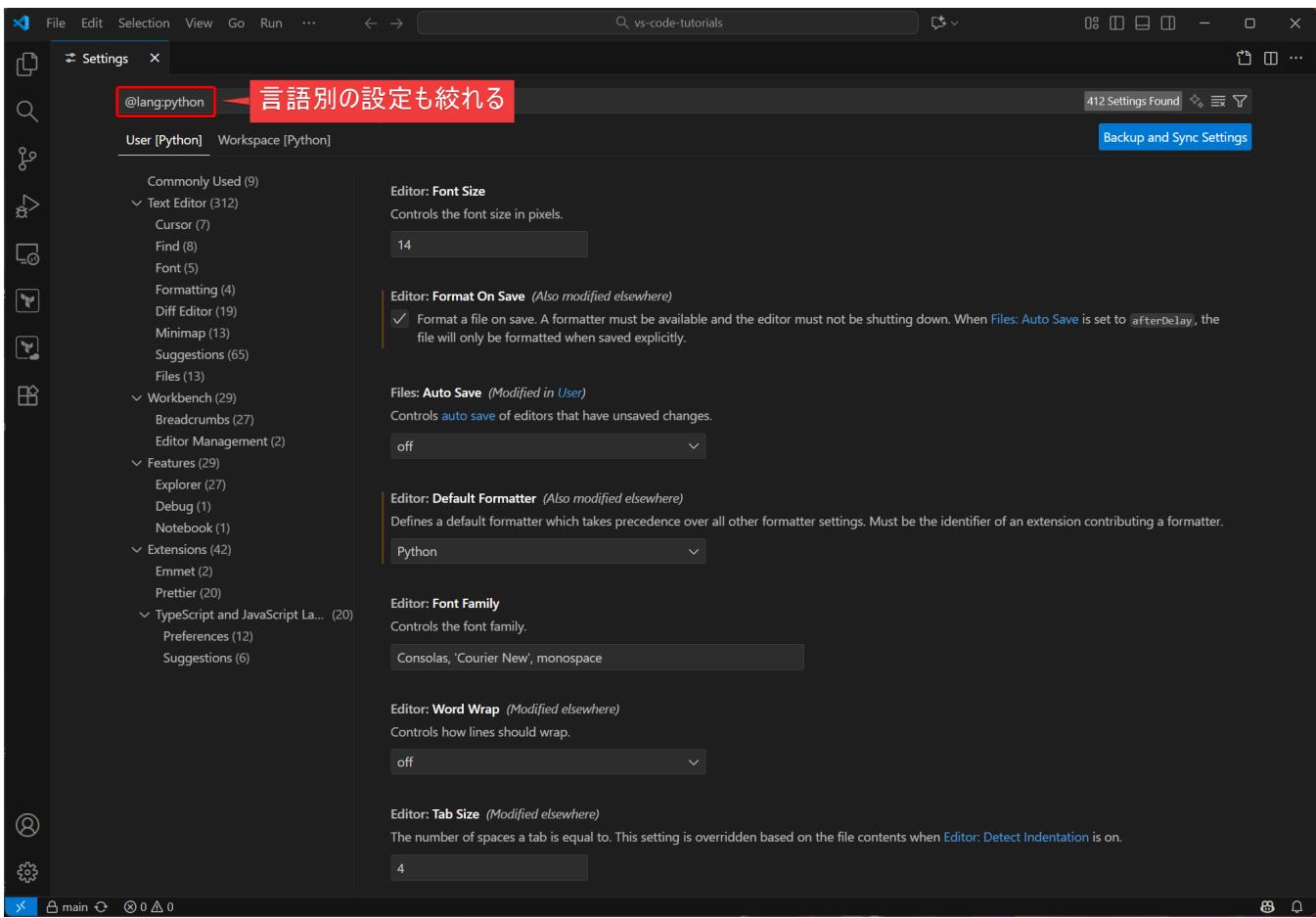


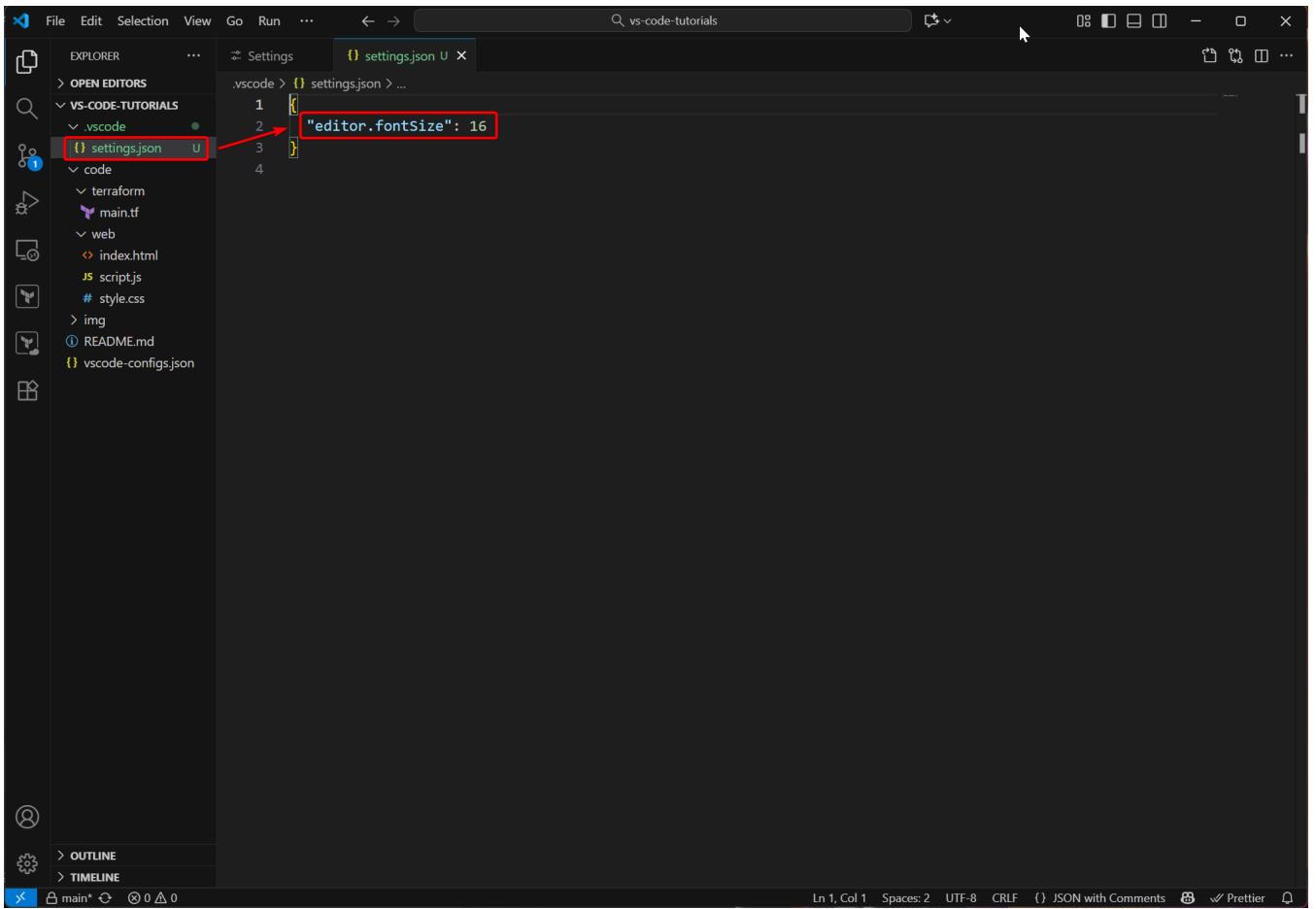
```
C: > Users > Kei Ichi > AppData > Roaming > Code > User > settings.json > terraform.codelens.referenceCount
55 "files.exclude": {
56   "**/.git": true,
57   "node_modules/": true
58 },
59 "files.trimTrailingWhitespace": true,
60 "files.insertFinalNewline": true,
61 "files.trimFinalNewlines": true,
62
63
64 "search.exclude": {
65   "**/node_modules": true,
66   "**/bower_components": true,
67   "**/dist": true,
68   "**/coverage": true,
69   "**/.output": true
70 },
71
72 "workbench.settings.editor": "ui",
73 "workbench.editor.highlightModifiedTabs": true,
74 "workbench.editor.titleScrollbarSizing": "large",
75 "workbench.colorCustomizations": {},
76 "terrafrom.codelens.referenceCount": true
77 }
78 }
```

The screenshot shows the VS Code code editor displaying the contents of the 'settings.json' file. The line 'terrafrom.codelens.referenceCount': true is highlighted with a red box. The status bar at the bottom indicates the file is JSON with comments and is prettier-ed.

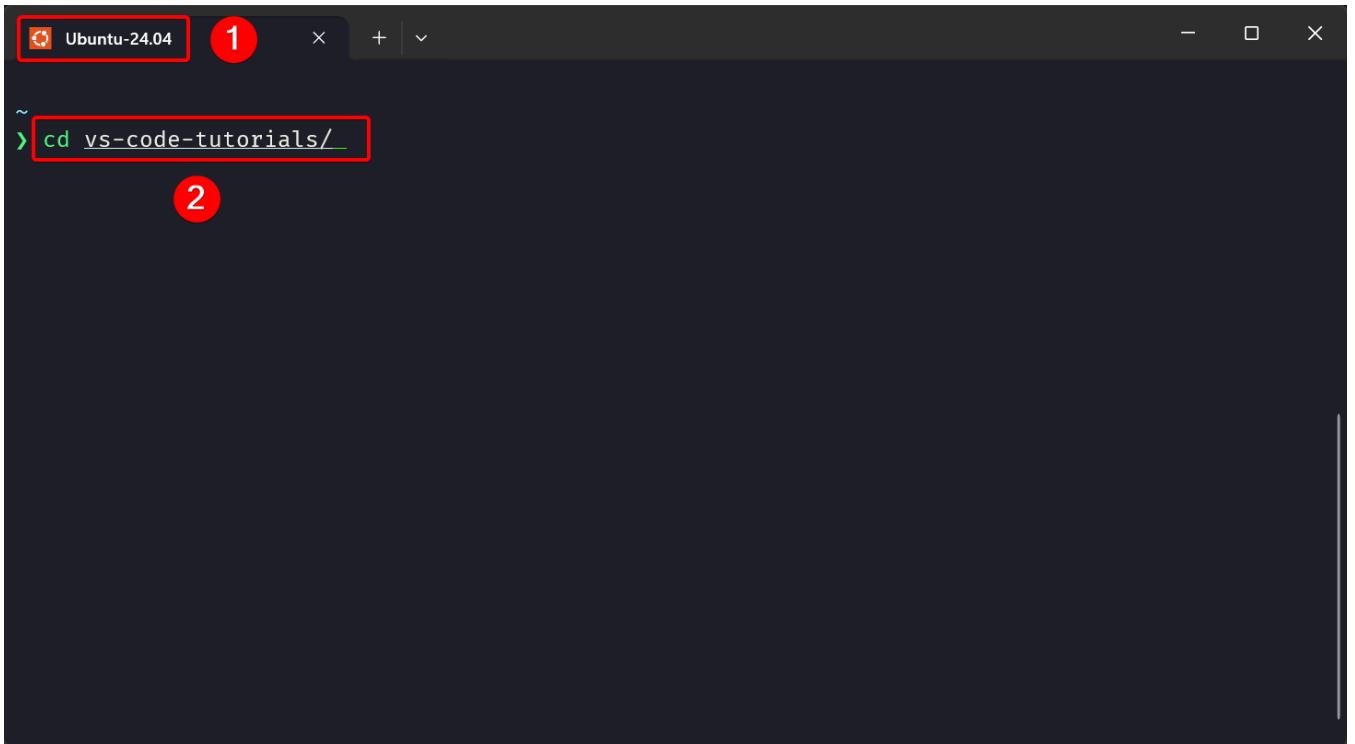








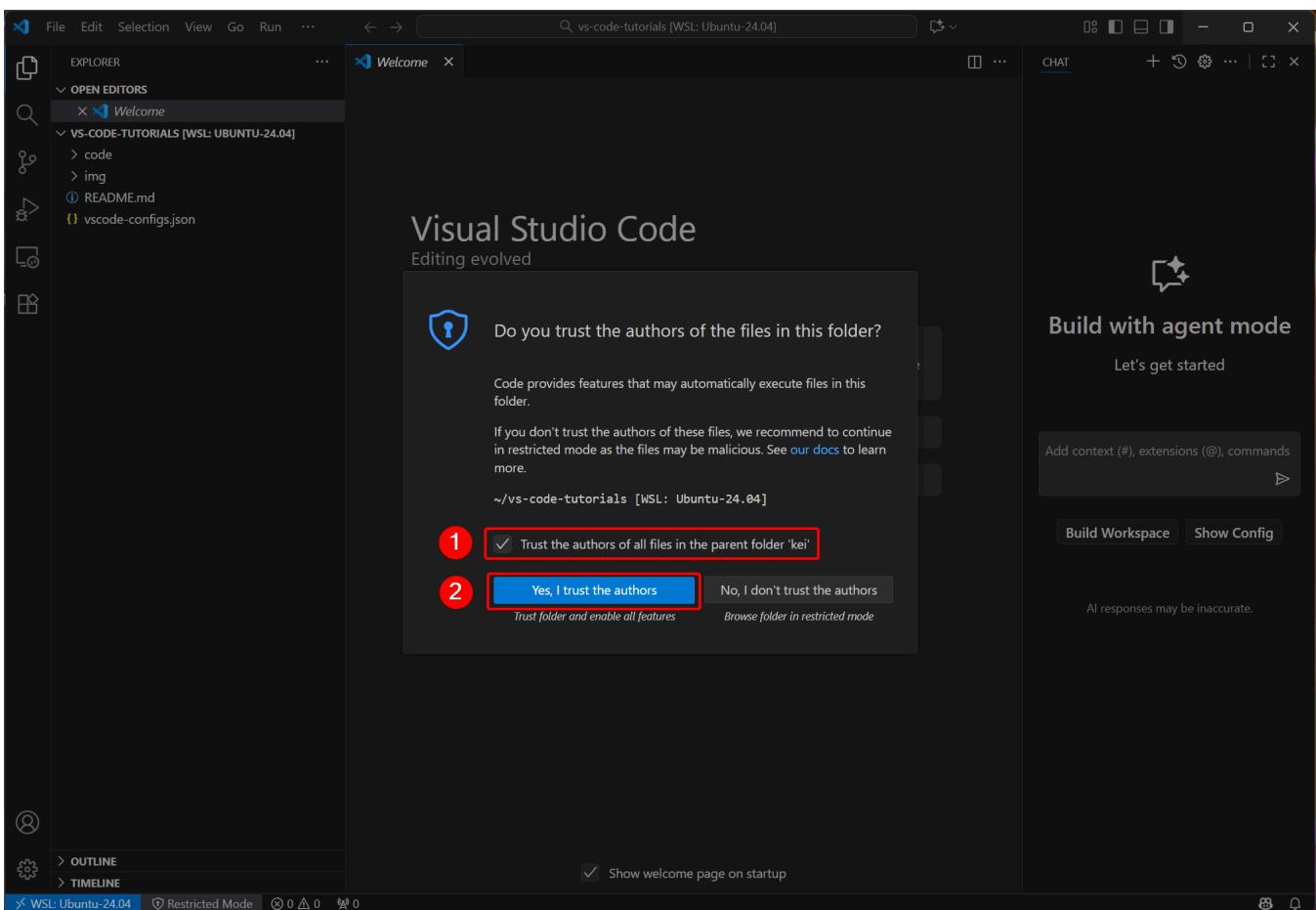
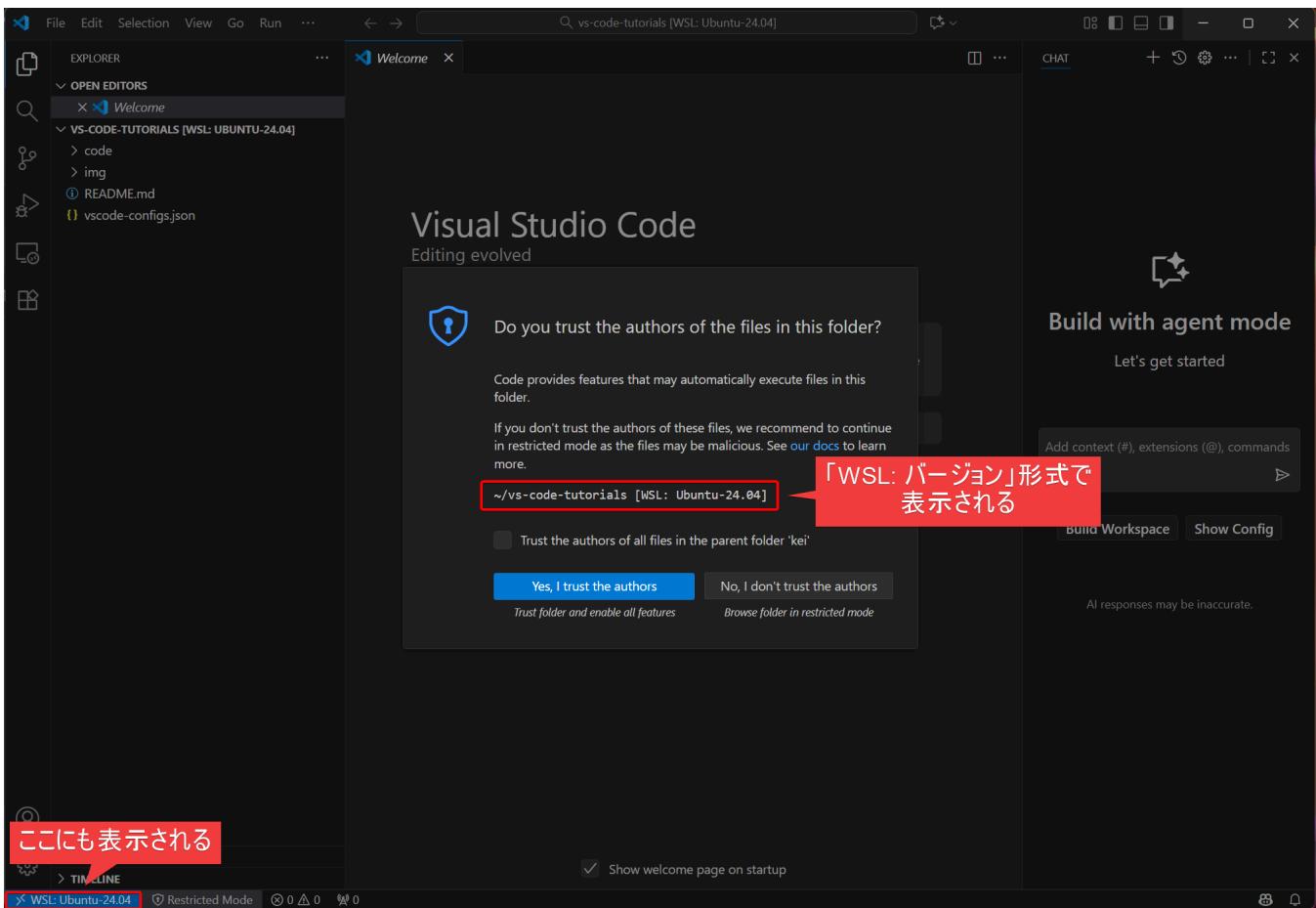
4. WSL Access

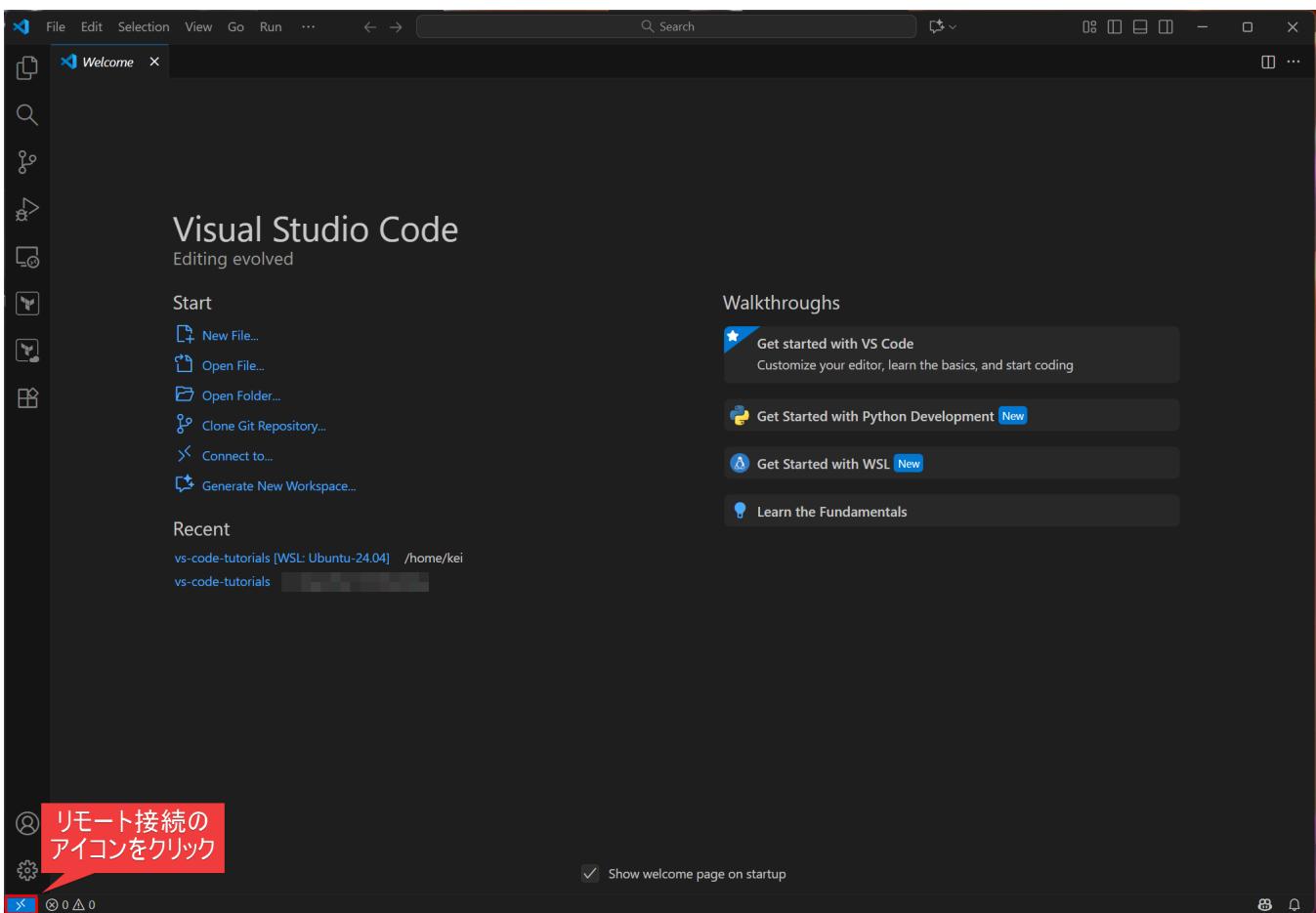
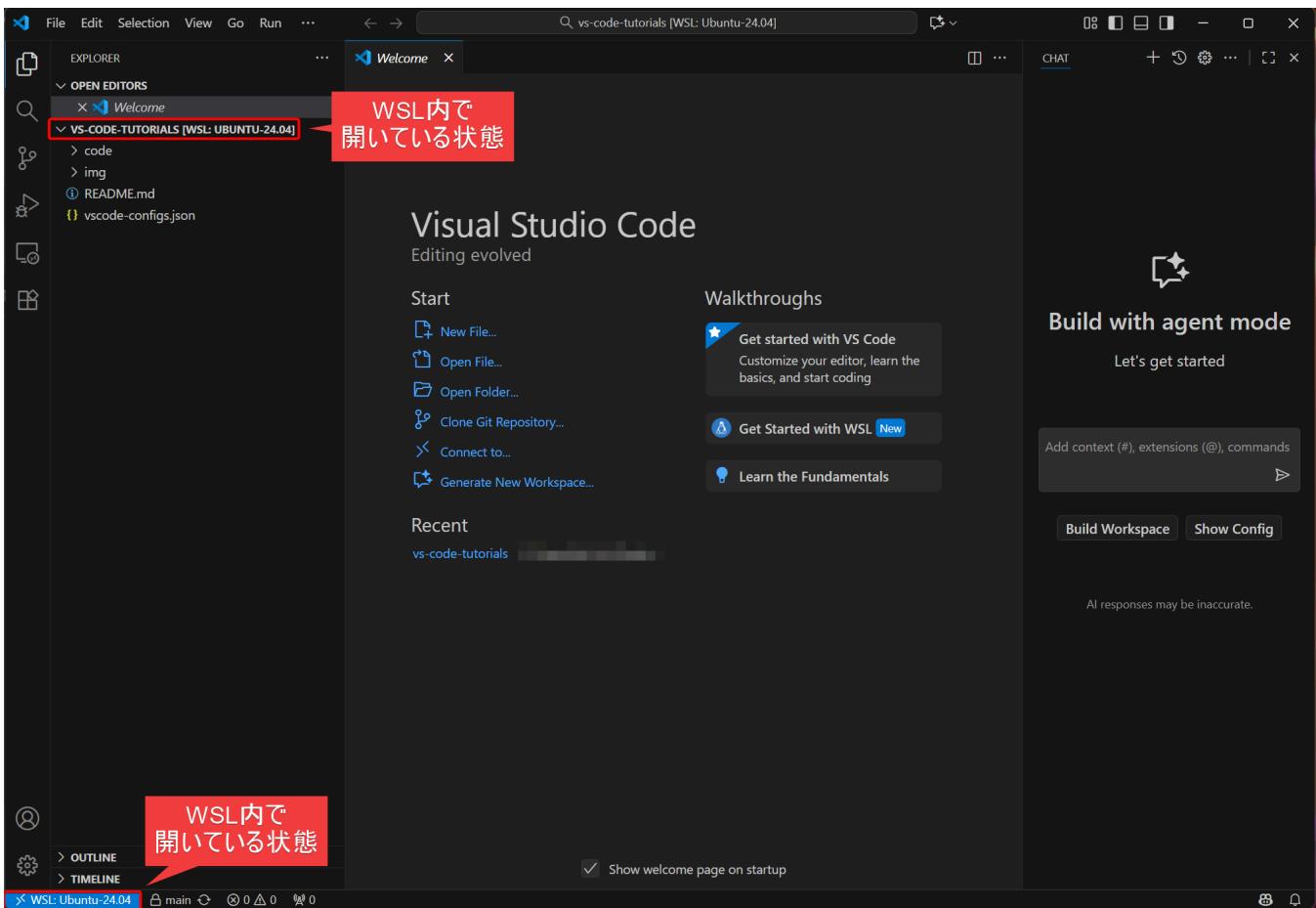


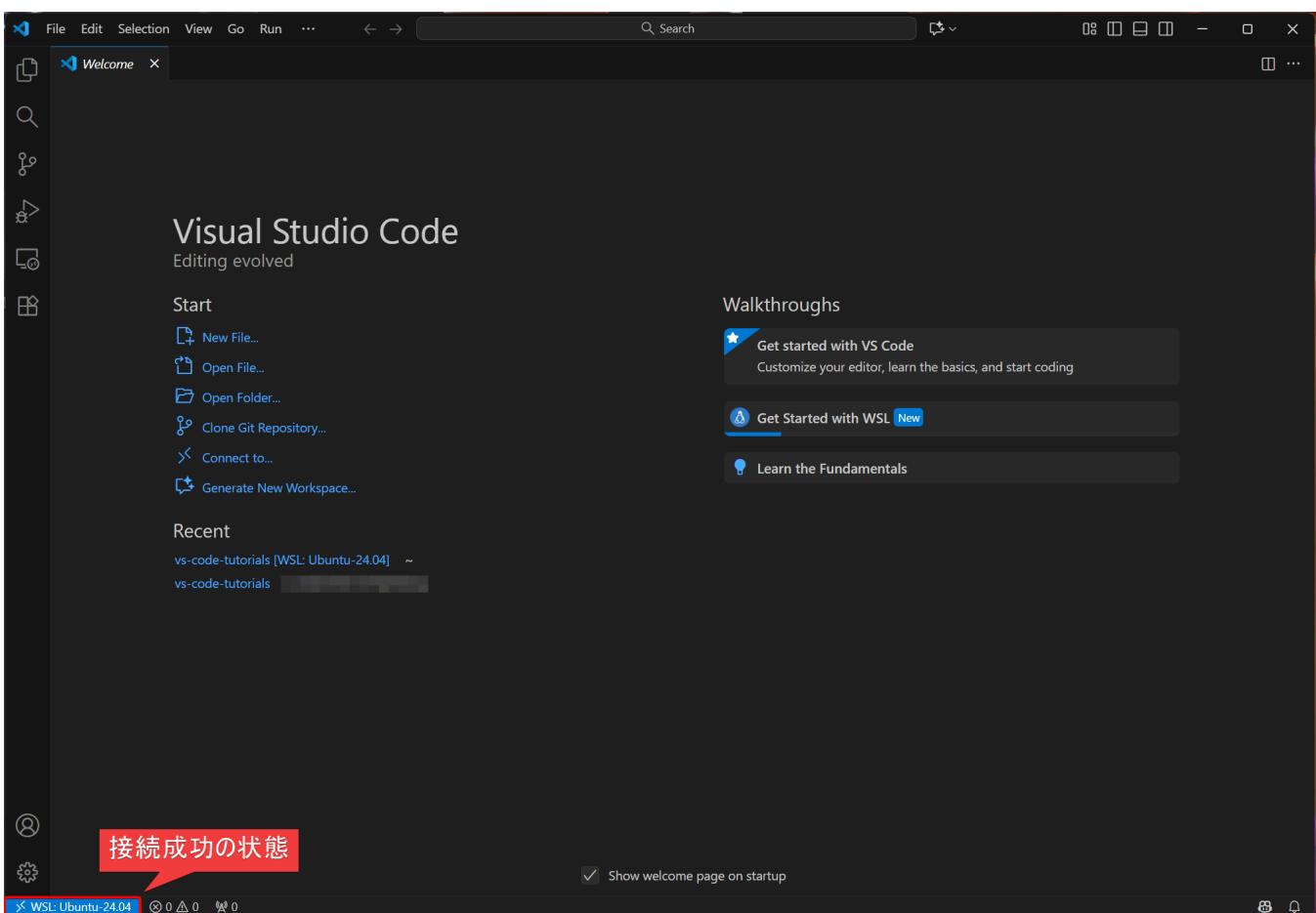
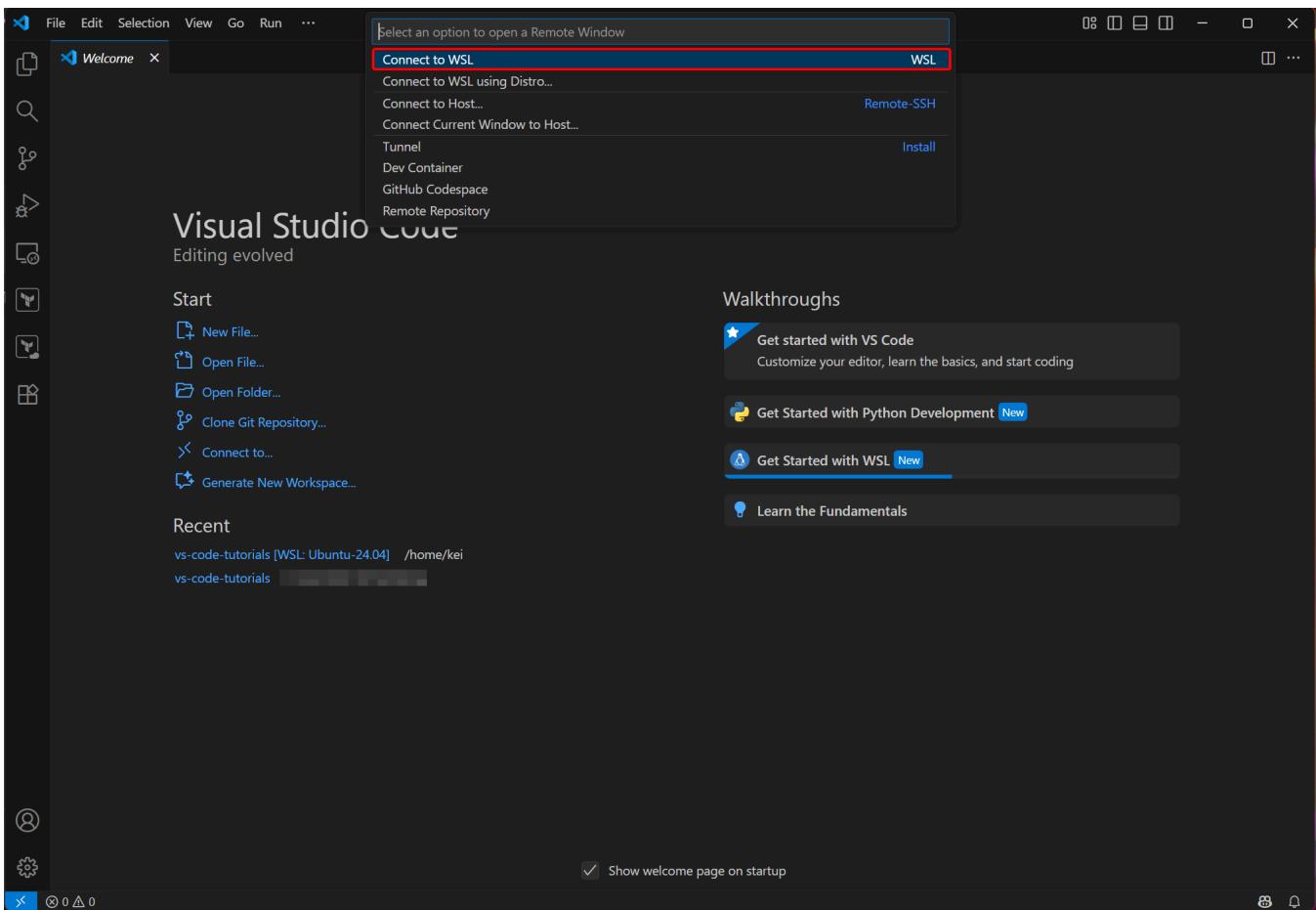
```
Ubuntu-24.04 x + v - □ ×
vs-code-tutorials on ✚ main
> code .
```

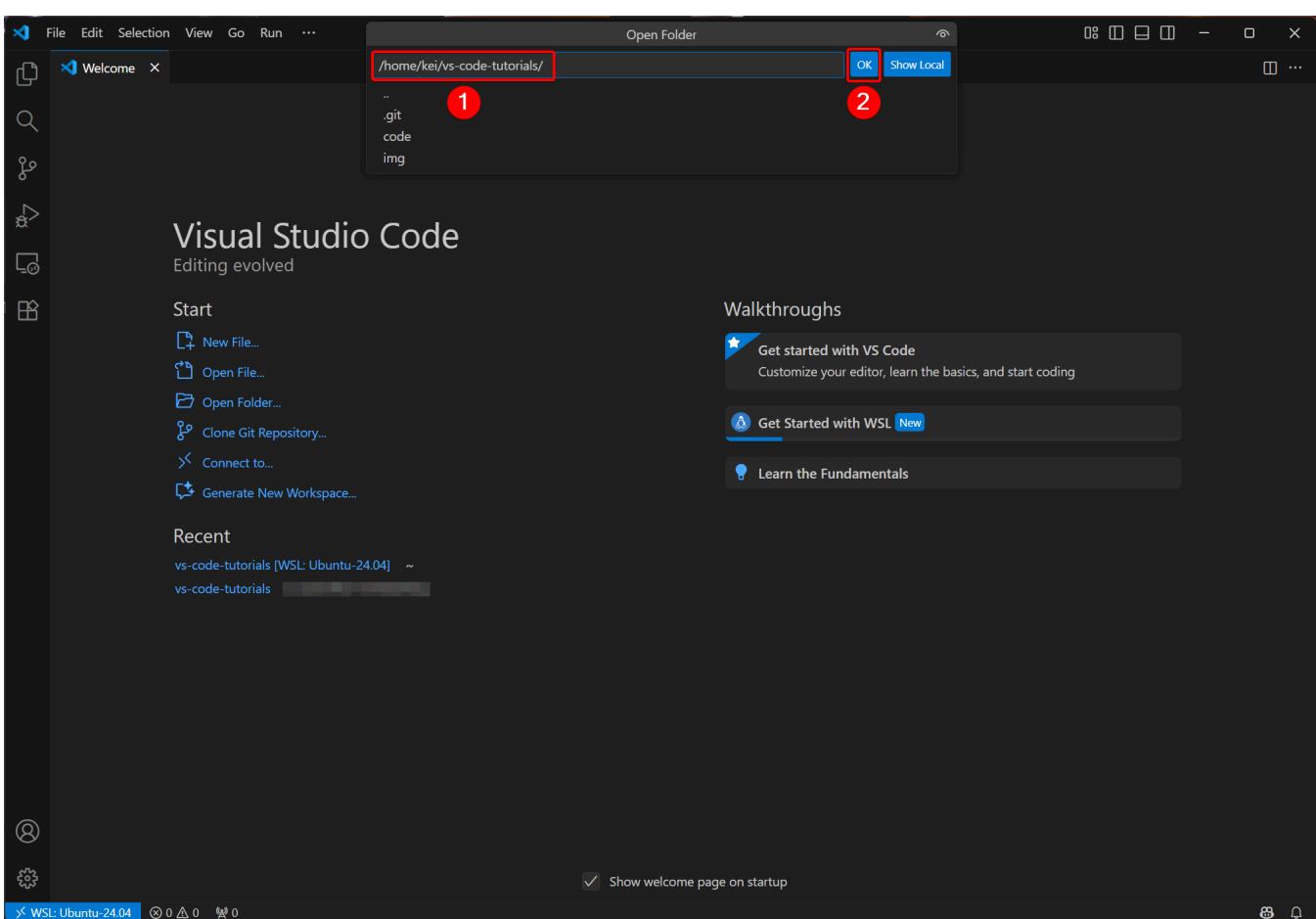
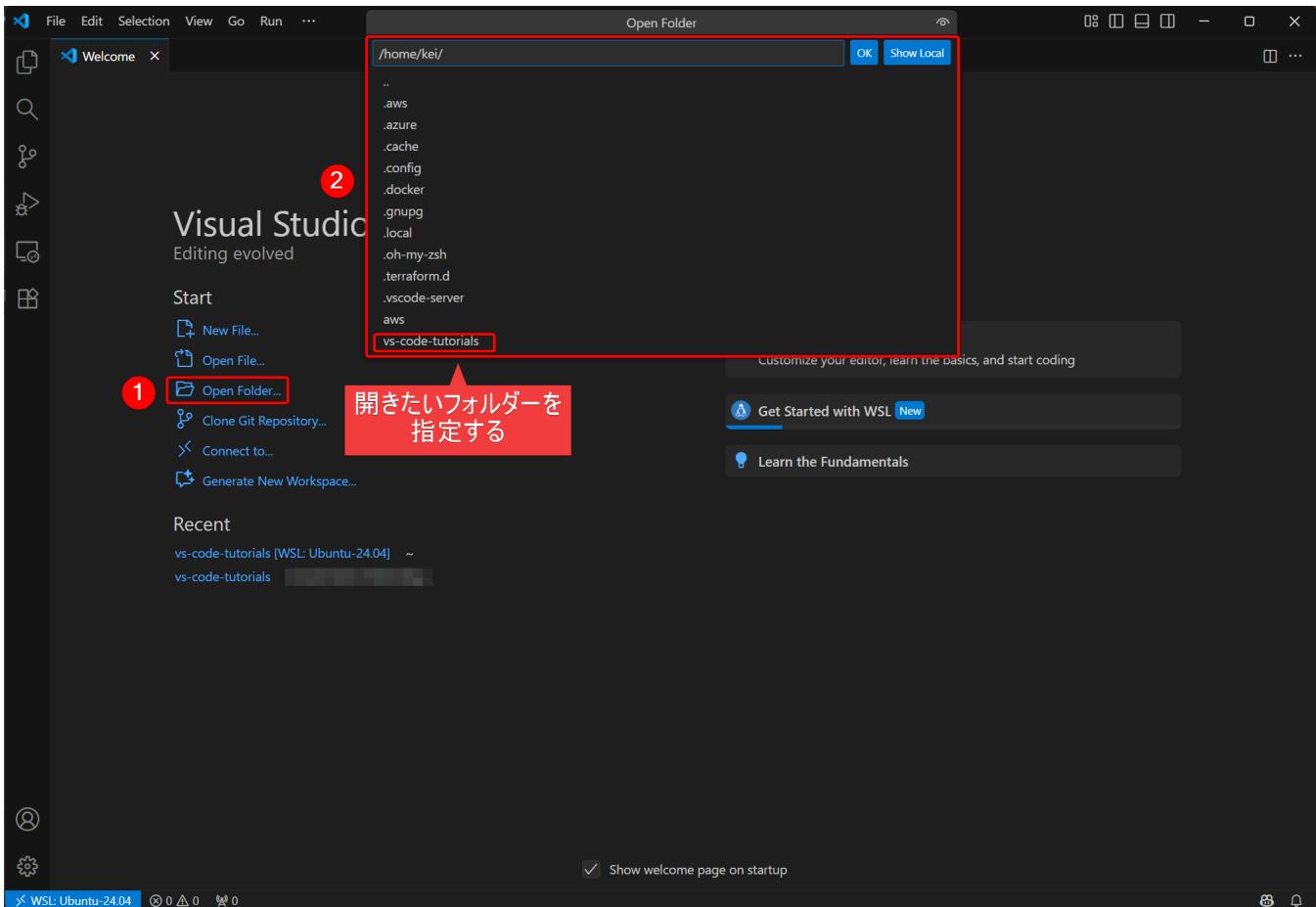
```
Ubuntu-24.04 x + v - □ ×
vs-code-tutorials on ✚ main
> code .
Installing VS Code Server for Linux x64 (7d842fb85a0275a4a8e4d7e040d2625abbf7f084)
Downloading: 100%
Unpacking: 100%
Unpacked 2265 files and folders to [REDACTED]/.vscode-server/bin/7d842fb85a0275a4a8e4d7e040d2625abbf7f084.
Looking for compatibility check script at [REDACTED]/.vscode-server/bin/7d842fb85a0275a4a8e4d7e040d2625abbf7f084/bin/helpers/check-requirements.sh
Running compatibility check script
Compatibility check successful (0)

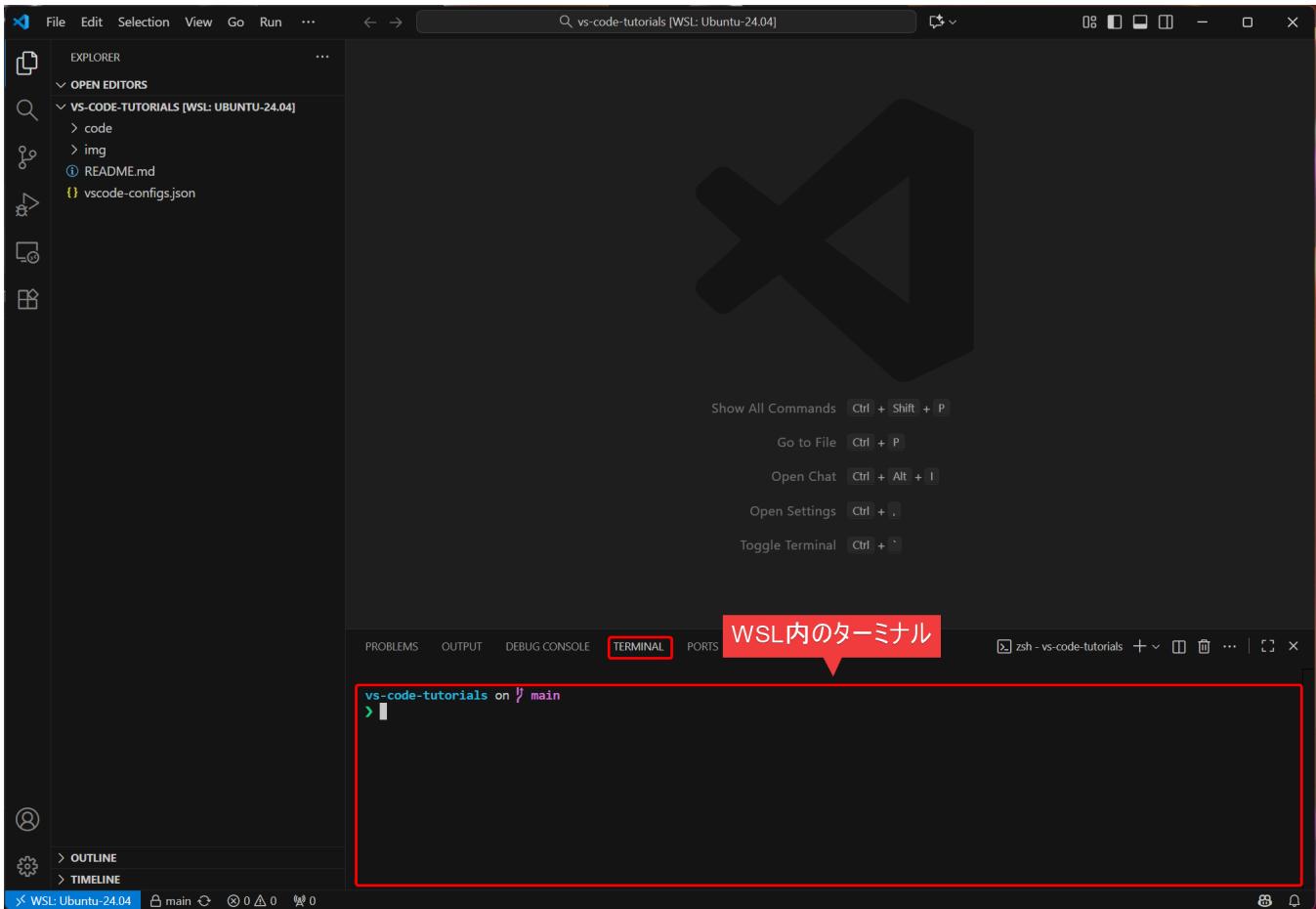
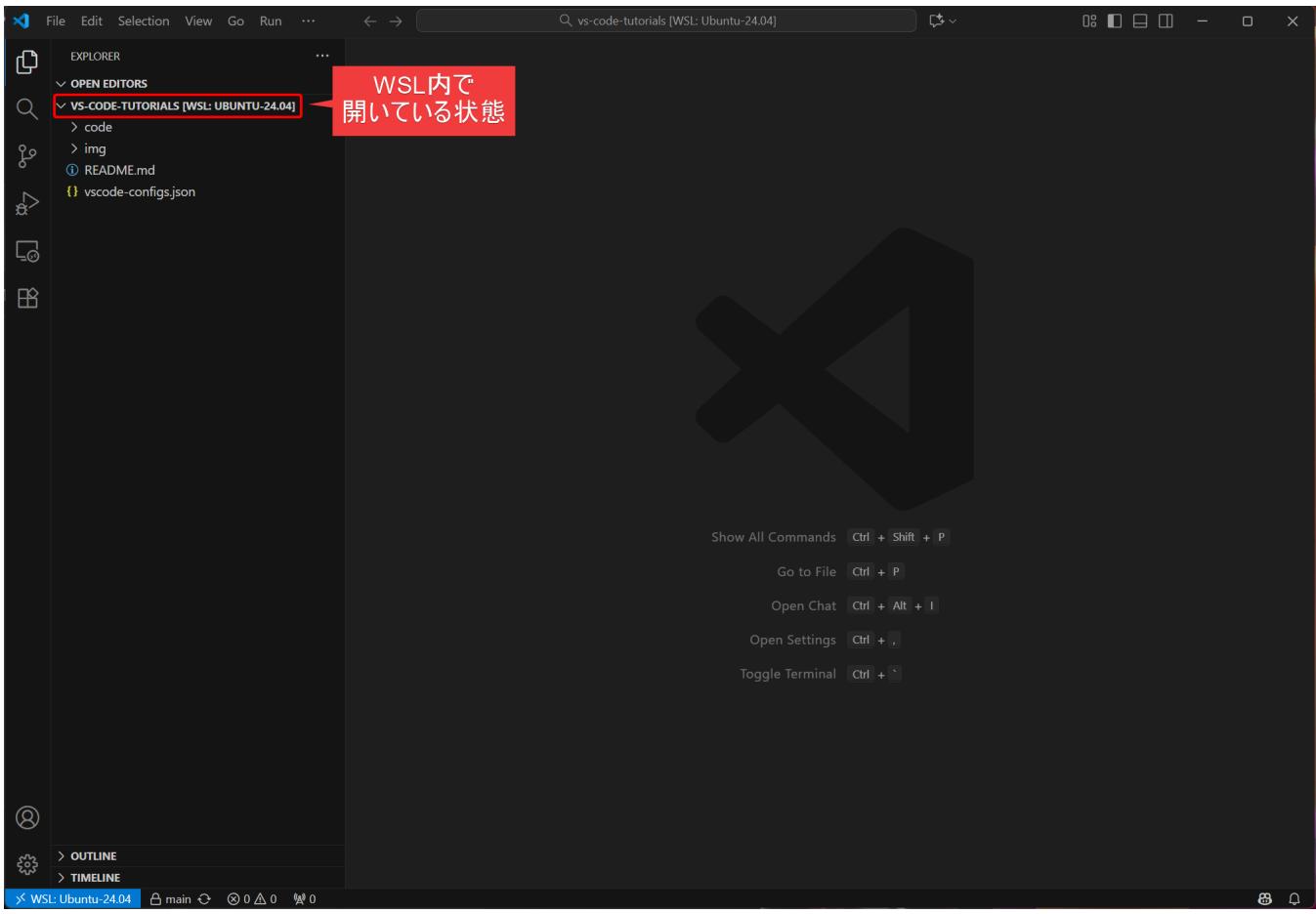
vs-code-tutorials on ✚ main took 5s
> _
```











```
main.tf
code > terraform > main.tf
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 data "aws_ami" "ubuntu" {
17   most_recent = true
18
19   filter {
20     name  = "name"
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
22   }
23
24   owners = ["099720109477"] # Canonical
25 }
26
27 resource "aws_instance" "app_server" {
28   ami           = data.aws_ami.ubuntu.id
29   instance_type = "t2.micro"
30
31   tags = {
32     Name = "learn-terraform"
33   }
34 }
35
36 data "aws_ami" "ubuntu" {
37   most_recent = true
38
39   filter {
40     name  = "name"
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42   }
43 }
```

拡張機能が機能しない

EXTENSIONS

Search Extensions in Marketplace

LOCAL - INSTALLED (13)

- Remote - SSH
- Remote - SSH: Editing Configuration
- Remote Explorer
- WSL
- HashiCorp Terraform

> WSL: UBUNTU-24.04 INSTALLED (RECOMMENDED)

- Dev Containers
- Container Tools
- Microsoft Terraform
- Microsoft Edge Tools ...
- GitHub Copilot

MCP SERVERS

main.tf

```
main.tf
code > terraform > main.tf
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 data "aws_ami" "ubuntu" {
17   most_recent = true
18
19   filter {
20     name  = "name"
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
22   }
23
24   owners = ["099720109477"] # Canonical
25 }
26
27 resource "aws_instance" "app_server" {
28   ami           = data.aws_ami.ubuntu.id
29   instance_type = "t2.micro"
30
31   tags = {
32     Name = "learn-terraform"
33   }
34 }
35
36 data "aws_ami" "ubuntu" {
37   most_recent = true
38
39   filter {
40     name  = "name"
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42   }
43 }
```

非表示する

WSLにインストールされていない拡張機能

```

1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }

15 data "aws_ami" "ubuntu" {
16   most_recent = true

17   filter {
18     name   = "name"
19     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
20   }
21
22   owners = ["099720109477"] # Canonical
23 }

24
25 resource "aws_instance" "app_server" {
26   ami           = data.aws_ami.ubuntu.id
27   instance_type = "t2.micro"
28
29   tags = {
30     Name = "learn-terraform"
31   }
32 }

33 data "aws_ami" "ubuntu" {
34   most_recent = true

35   filter {
36     name   = "name"
37     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
38   }
39
40   owners = ["099720109477"] # Canonical
41 }

42
43

```

Install in WSL:Ubuntu-24.04

WSL: UBUNTU-24.04 - INSTALLED

RECOMMENDED

MCP SERVERS

WSL: Ubuntu-24.04

Extension: HashiCorp Terraform

HashiCorp Terraform

HashiCorp [hashicorp.com](#) | ⚡ 5,680,797 | ★★★☆☆ (206)

Syntax highlighting and autocompletion for Terraform

Disable | Uninstall | Switch to Pre-Release Version | Auto Update

Extension is enabled on 'WSL: Ubuntu-24.04'.

Terraform Extension for Visual Studio Code

The HashiCorp Terraform Extension for Visual Studio Code (VS Code) with the Terraform Language Server adds editing features for Terraform, Terraform Stacks and Terraform Search files such as syntax highlighting, IntelliSense, code navigation, code formatting, module explorer and much more!

Quick Start

Get started writing Terraform configurations with VS Code in three steps:

- Step 1: If you haven't done so already, install Terraform
- Step 2: If you haven't done so already, install Terraform
- Step 3: To activate the extension, open any folder or VS Code workspace containing Terraform, Terraform Stacks or Terraform Search files. Once activated, the Terraform language indicator will appear in the bottom right corner of the window.

New to Terraform? Read the [Terraform Learning guides](#)

See [Usage](#) for more detailed getting started information.

Read the [Troubleshooting Guide](#) for answers to common questions.

Features

This extension provides Terraform, Terraform Stacks and Terraform Search language features. For most features Terraform Stacks and Terraform Search support is implied, specific

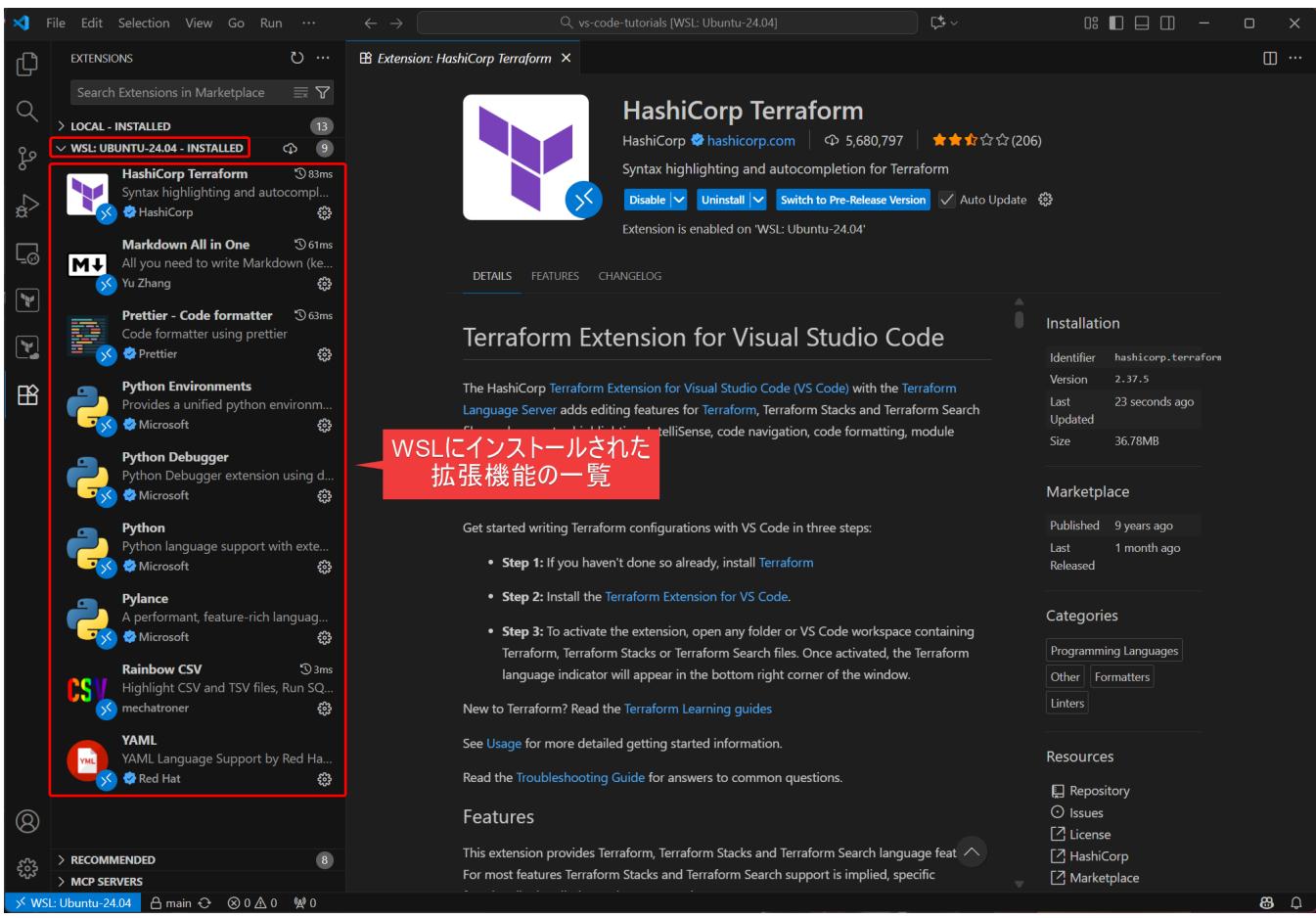
WSLに再インストールする

WSL: UBUNTU-24.04 - INSTALLED

RECOMMENDED

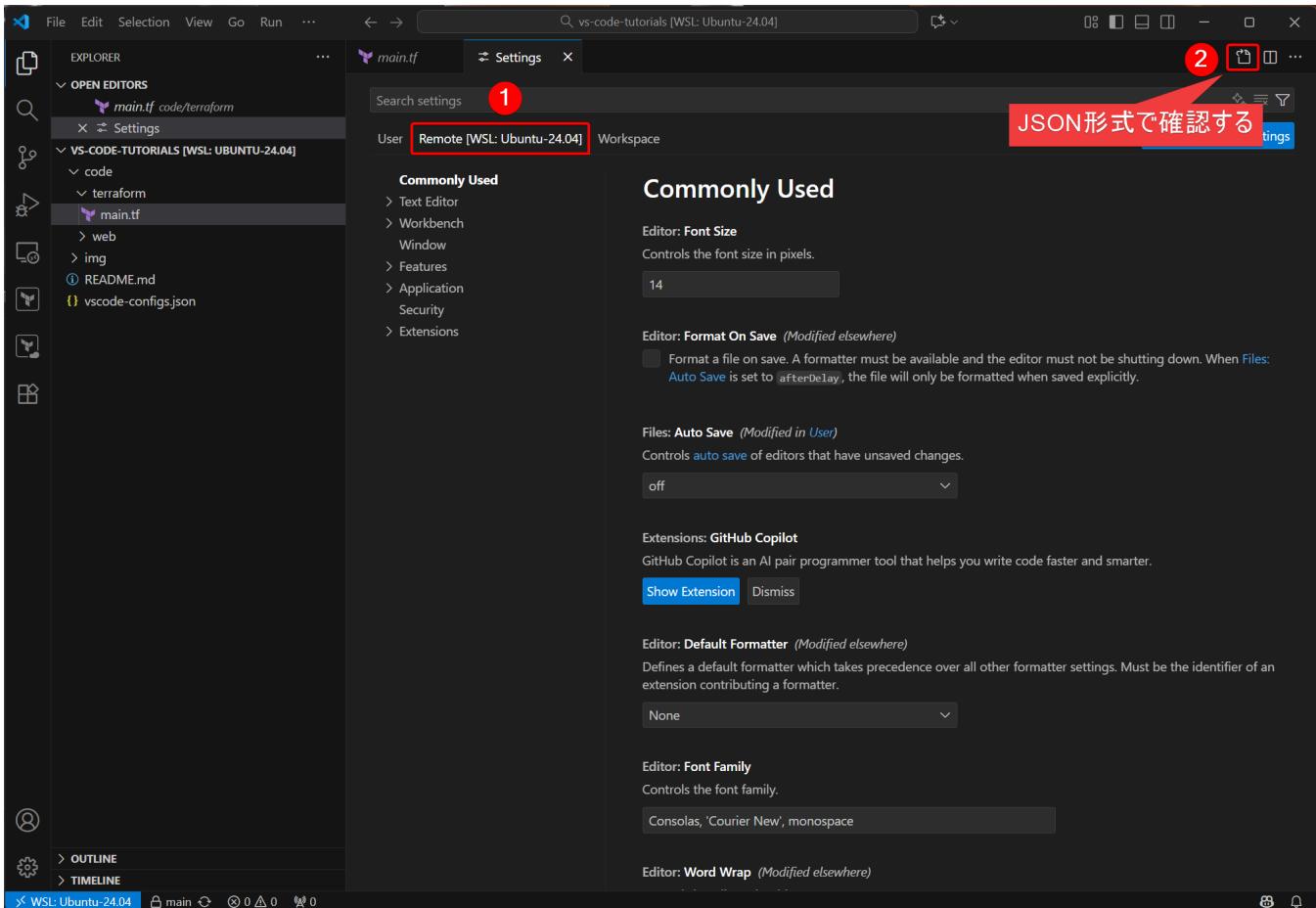
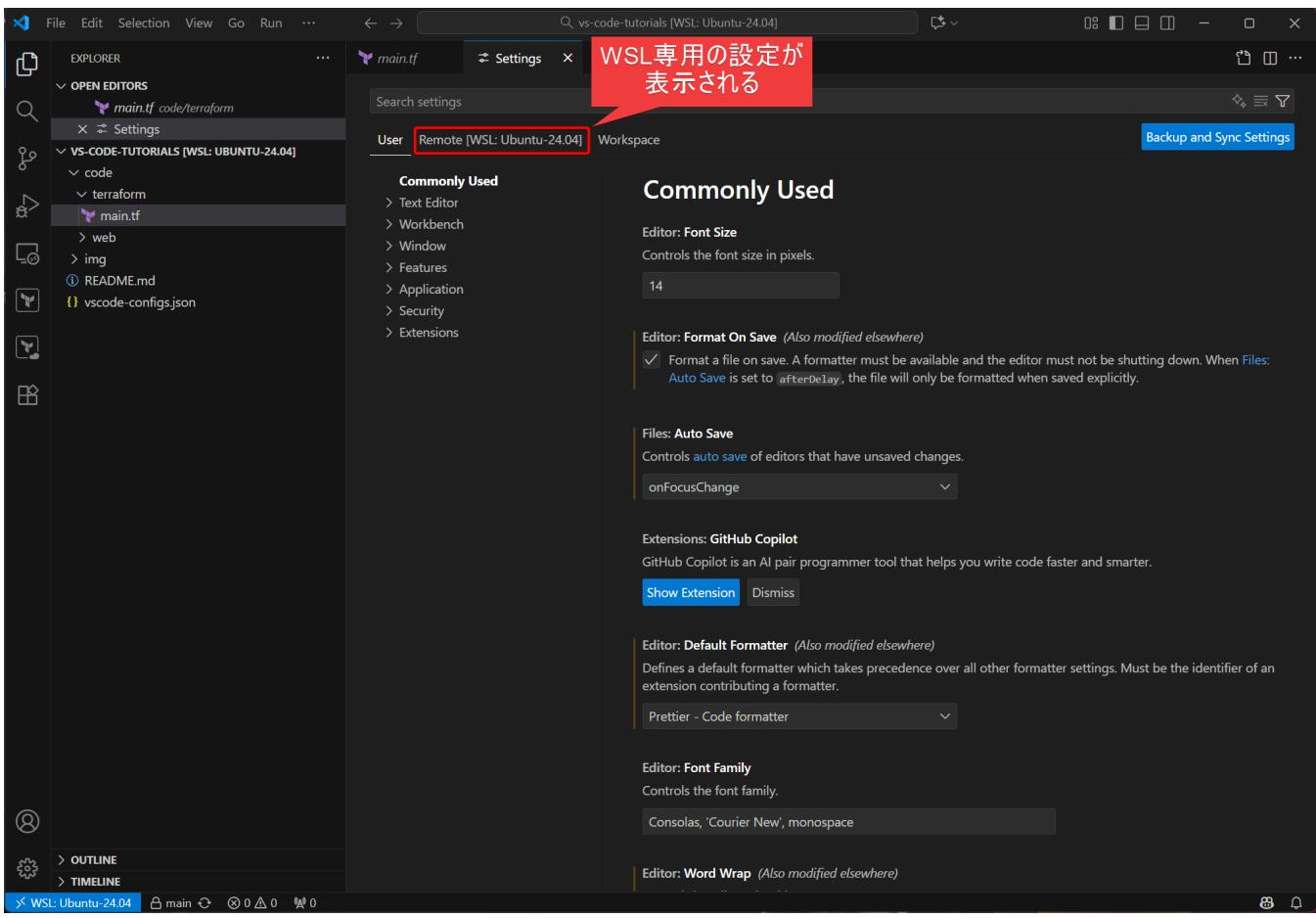
MCP SERVERS

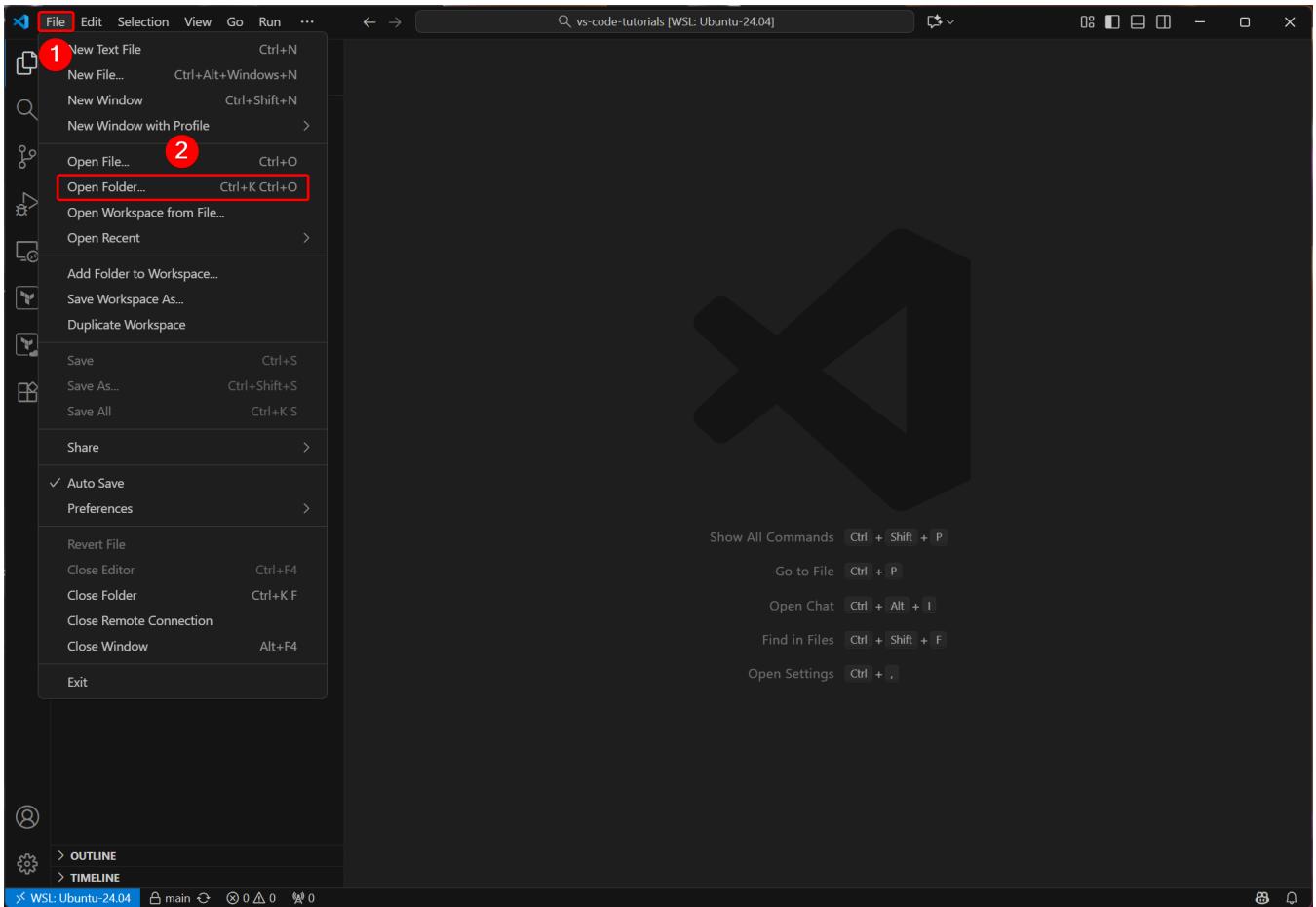
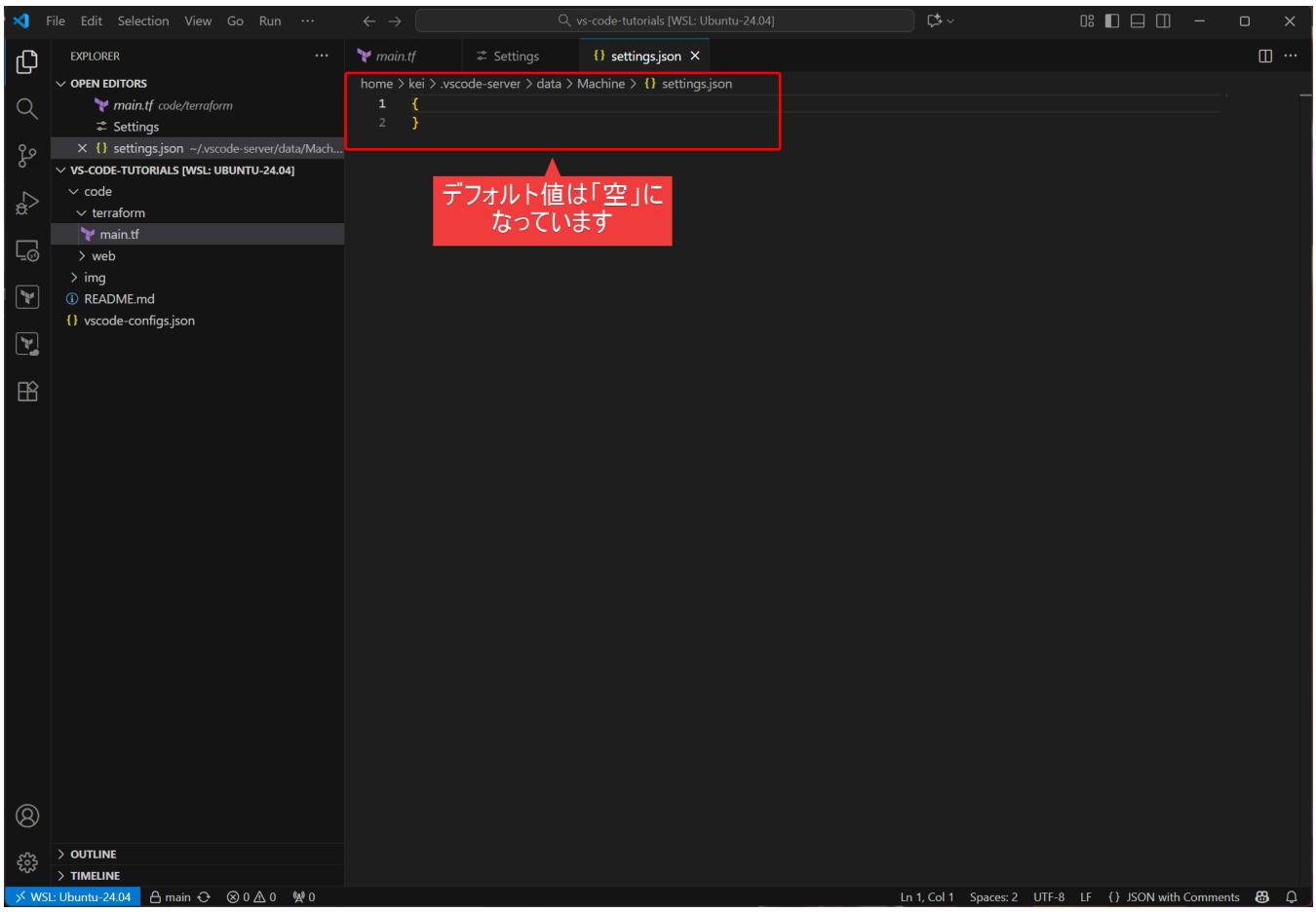
WSL: Ubuntu-24.04

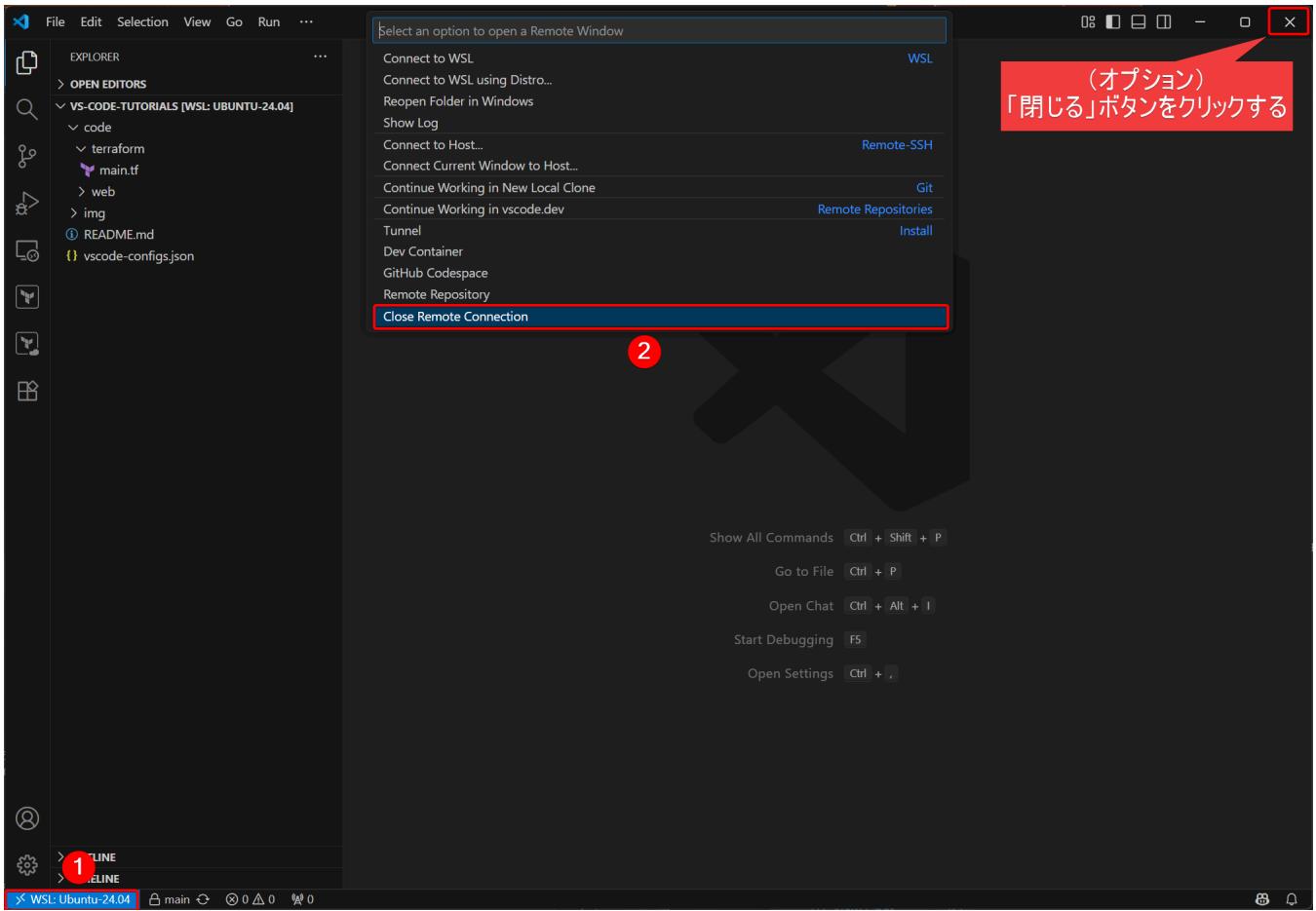
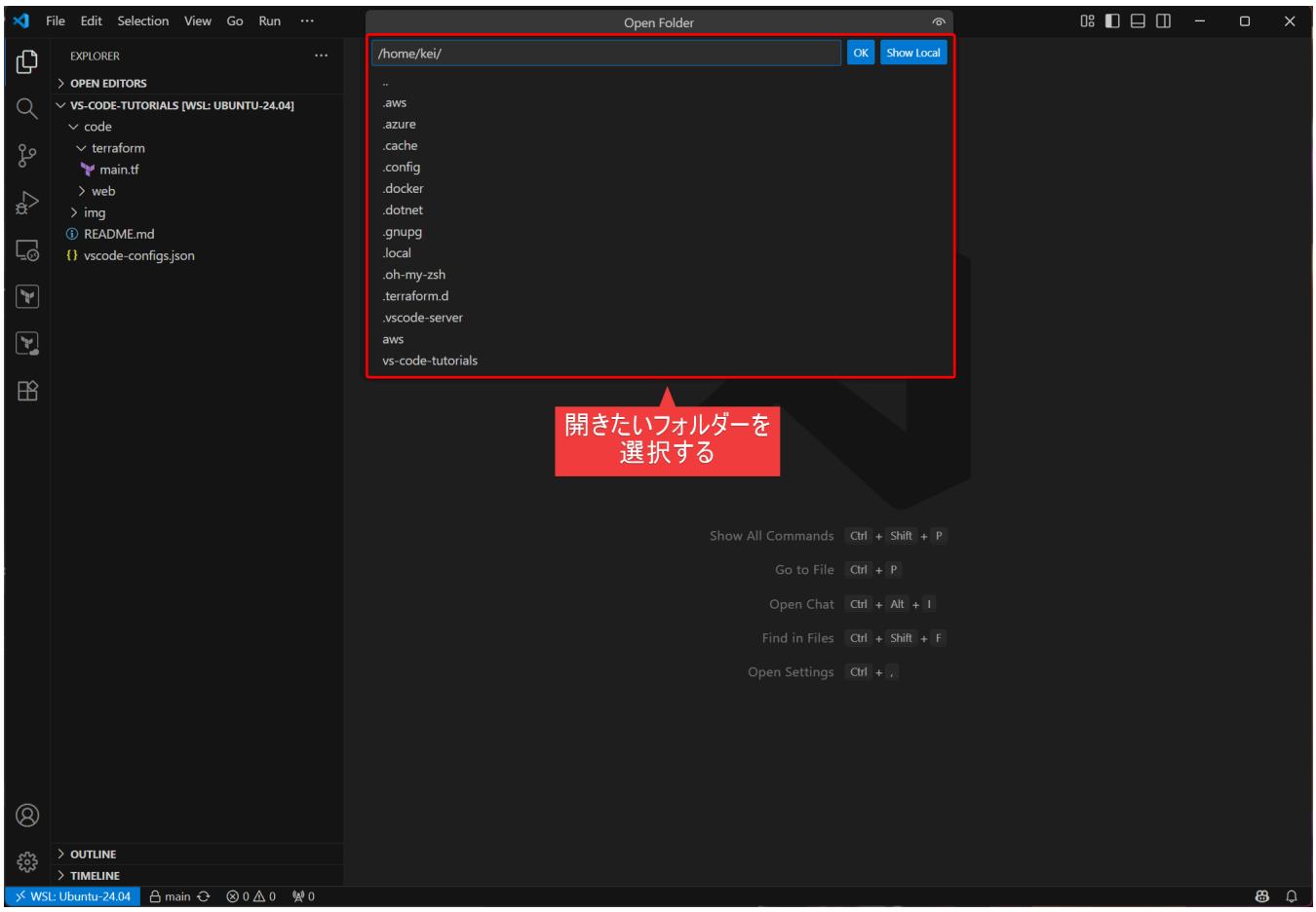


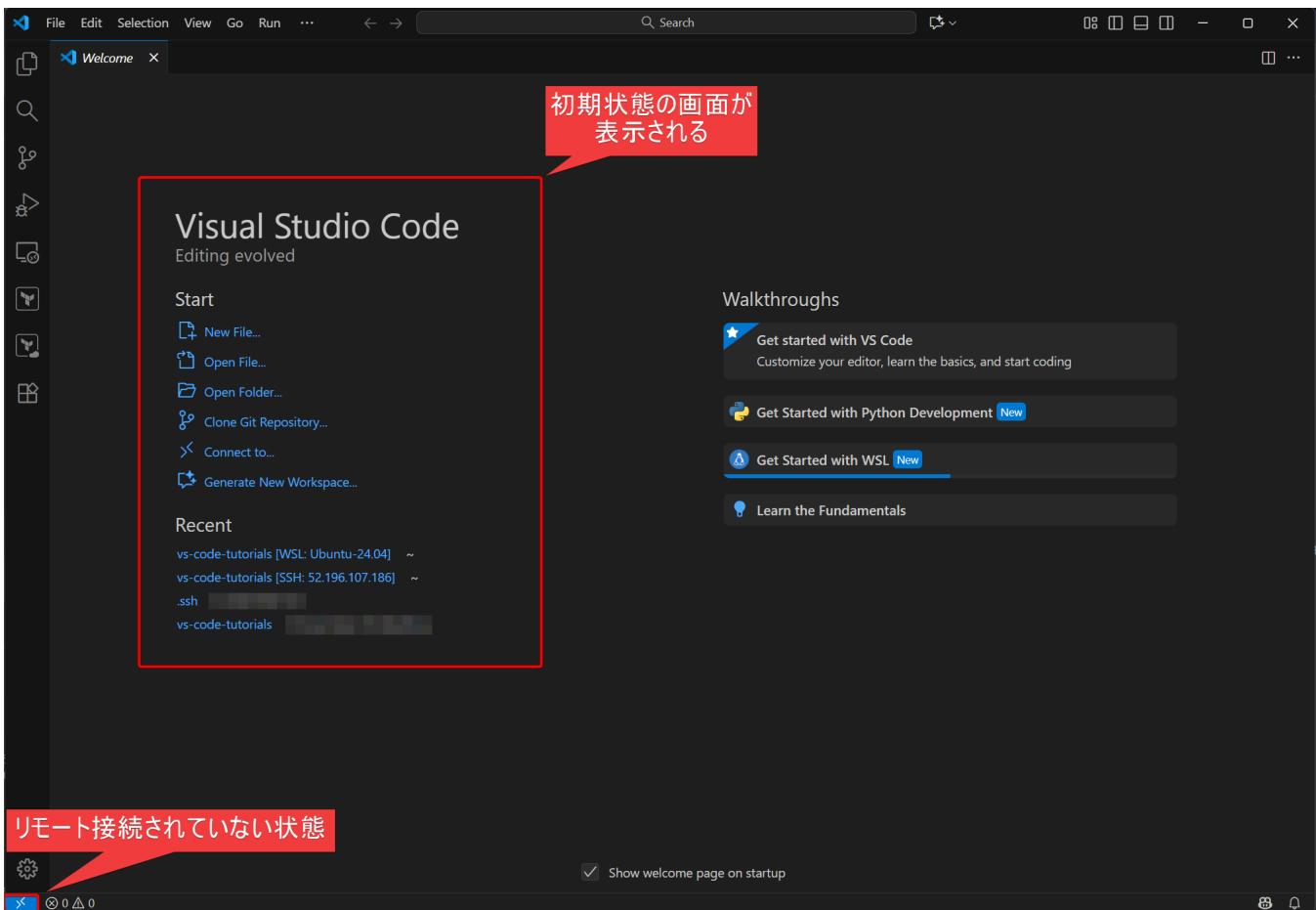
The screenshot shows the Visual Studio Code interface with the 'main.tf' file open in the editor. A red box highlights the 'main.tf' file in the Explorer sidebar. The code editor displays the Terraform configuration:

```
1 terraform {  
2   required_providers {  
3     aws = {  
4       source  = "hashicorp/aws"  
5       version = "~> 5.92"  
6     }  
7   }  
8  
9   required_version = ">= 1.2"  
10 }  
11  
12 provider "aws" {  
13   region = "us-west-2"  
14 }  
15  
16 data "aws_ami" "ubuntu" {  
17   most_recent = true  
18  
19   filter {  
20     name  = "name"  
21     values = ["ubuntu/images/hvm-ssd(gp3)/ubuntu-noble-24.04-amd64-server-*"]  
22   }  
23  
24   owners = ["099720109477"] # Canonical  
25 }  
26  
27 resource "aws_instance" "app_server" {  
28   ami           = data.aws_ami.ubuntu.id  
29   instance_type = "t2.micro"  
30  
31   tags = {  
32     Name = "learn-terraform"  
33   }  
34 }  
35  
36 data "aws_ami" "ubuntu" {  
37   most_recent = true  
38  
39   filter {  
40     name  = "name"  
41     values = ["ubuntu/images/hvm-ssd(gp3)/ubuntu-noble-24.04-amd64-server-*"]  
42   }  
43 }
```

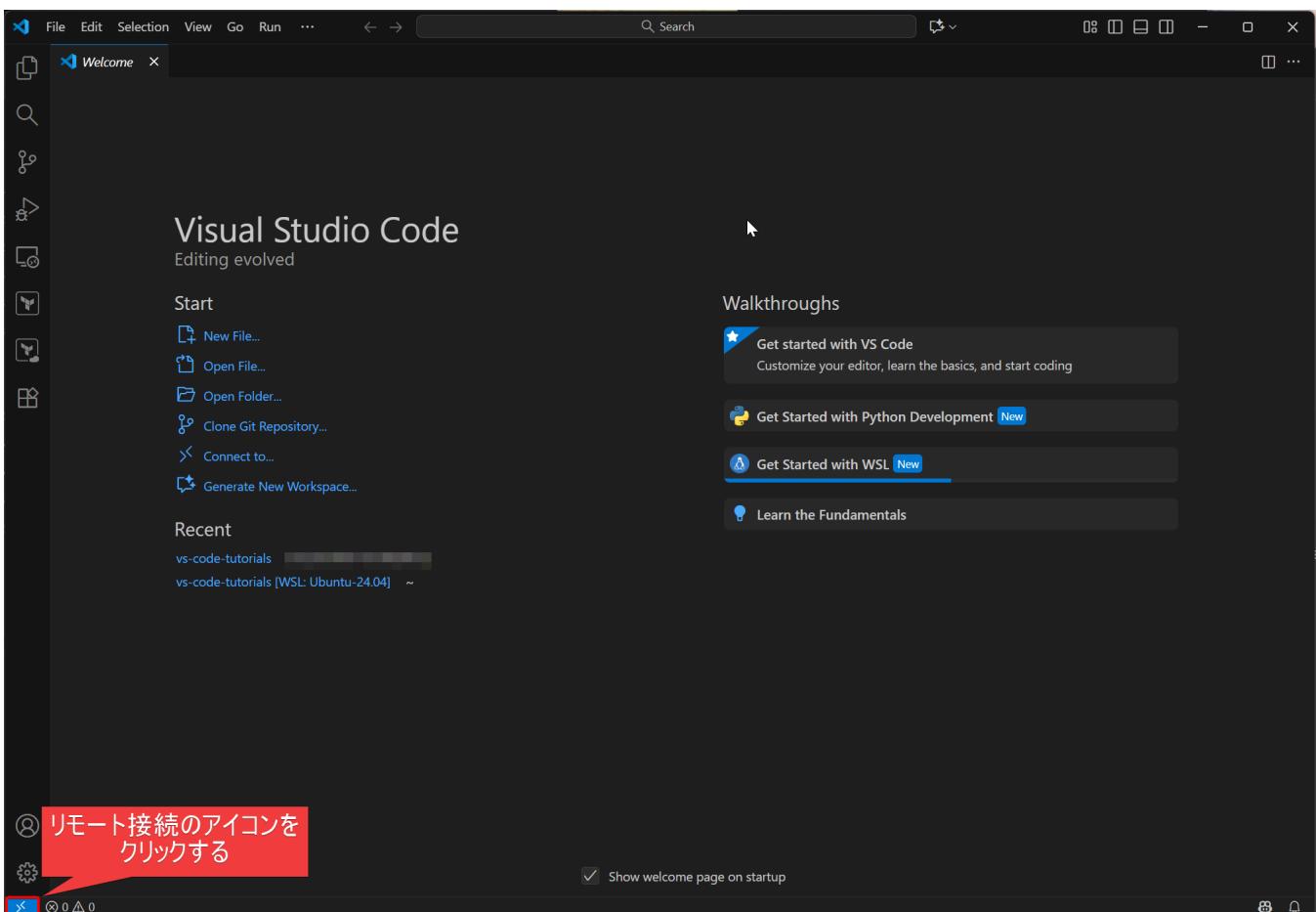


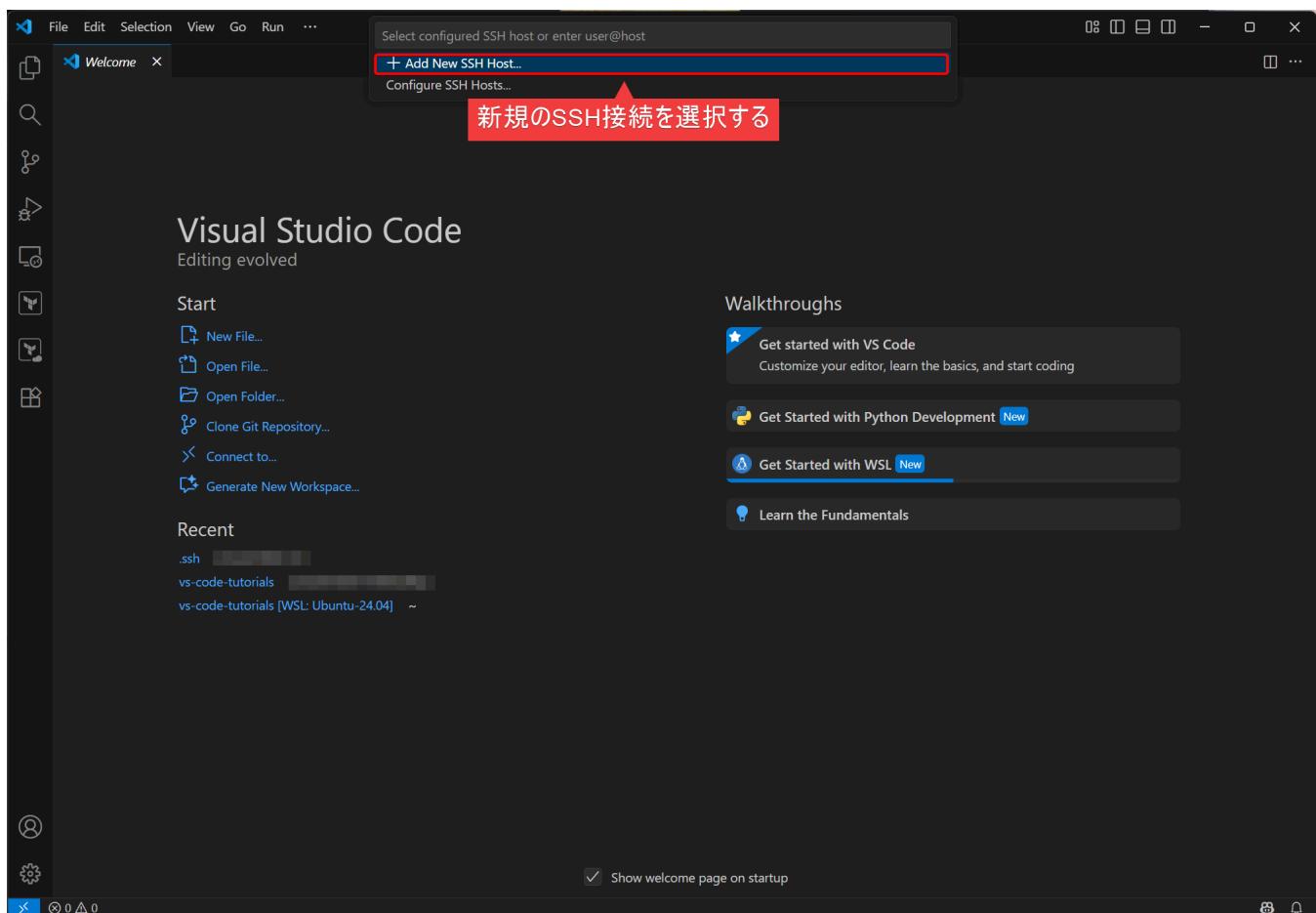
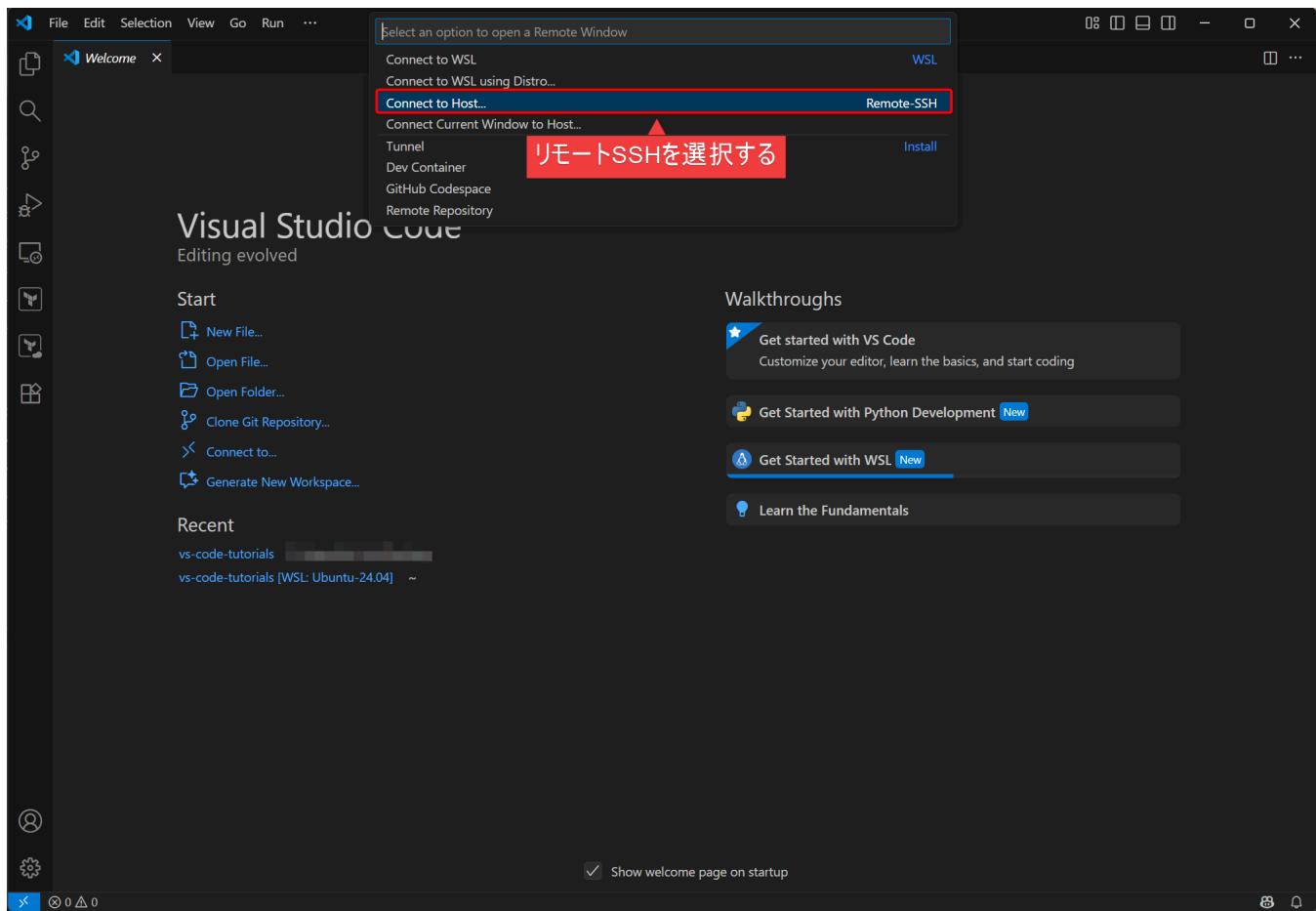


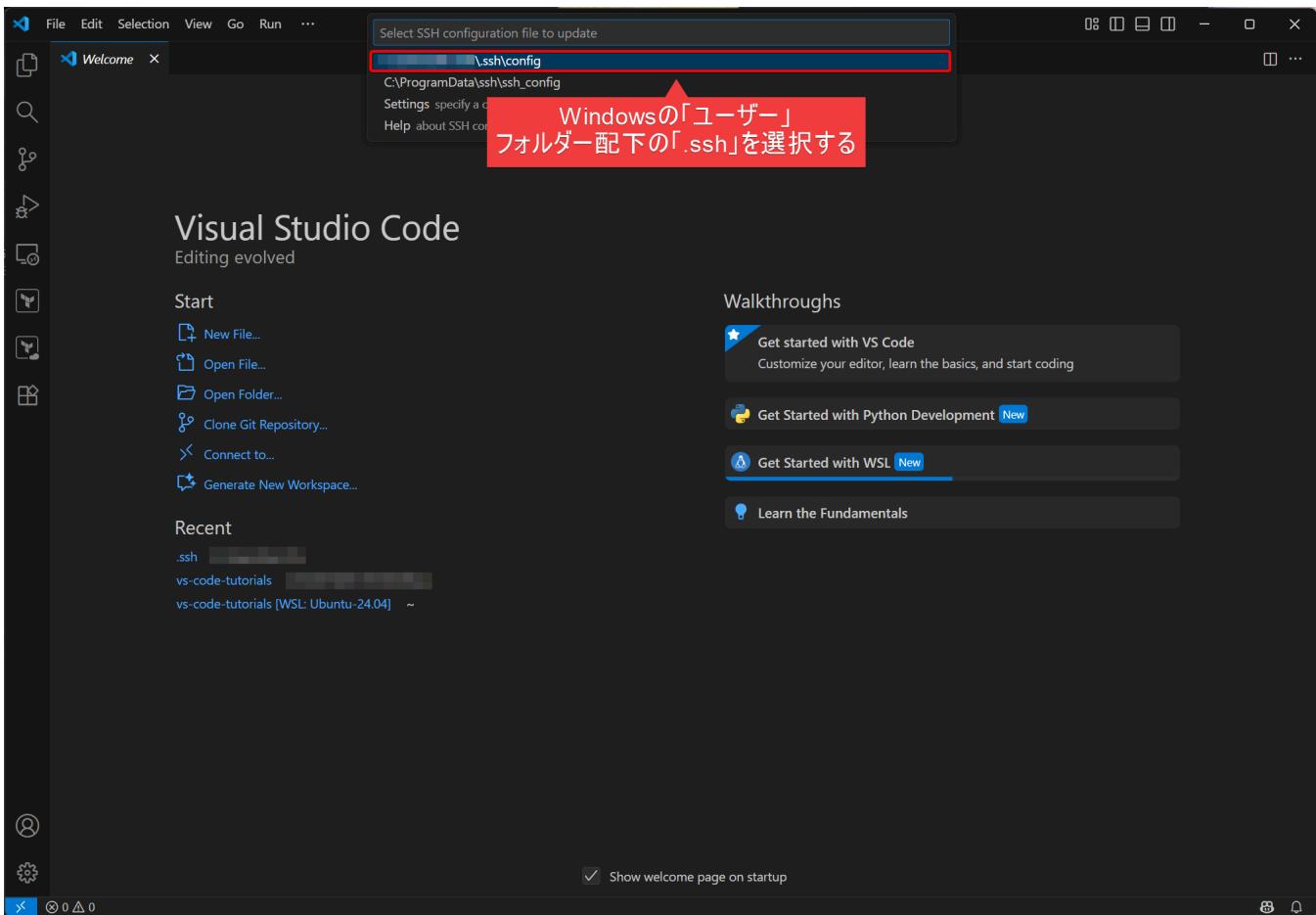
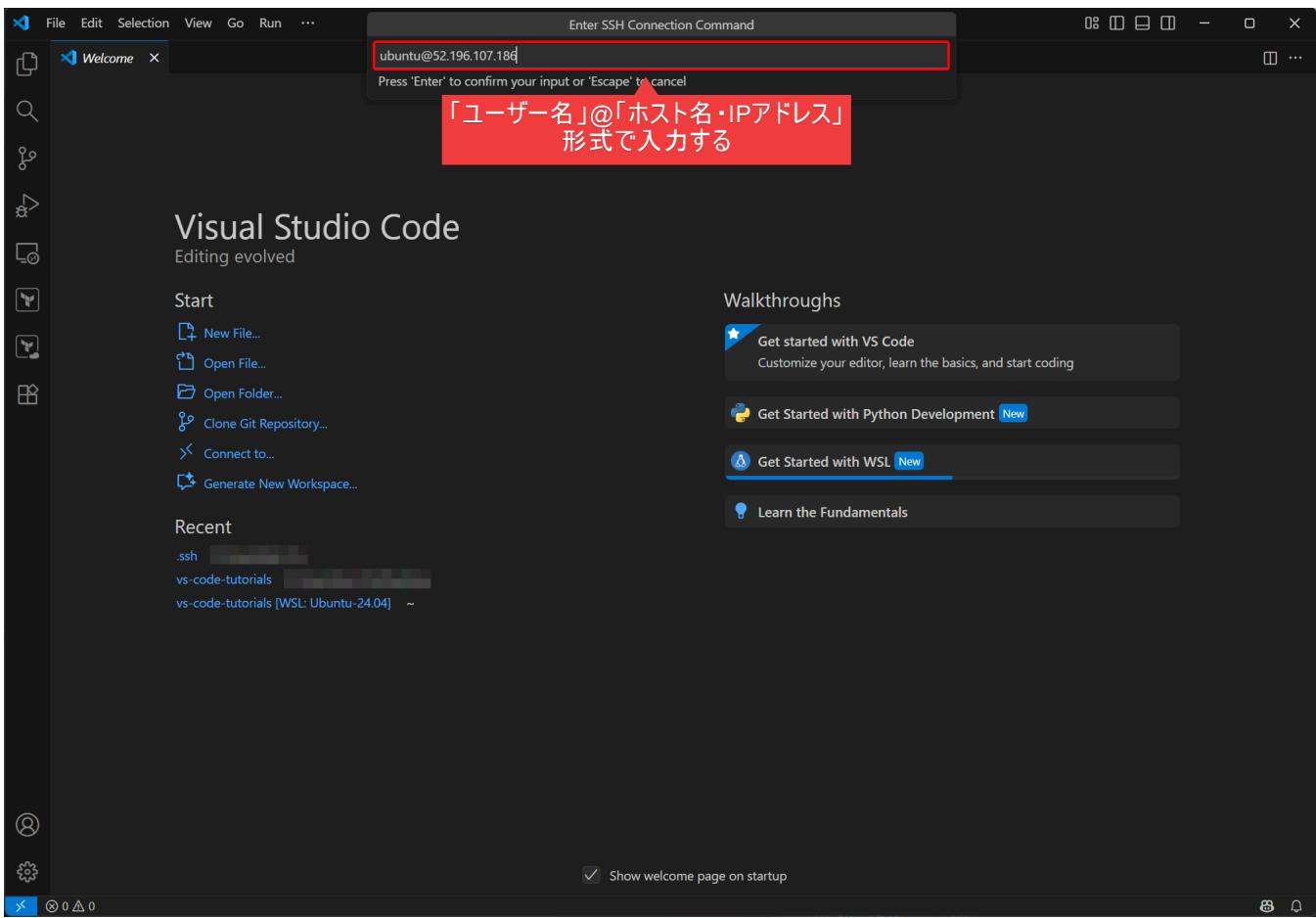


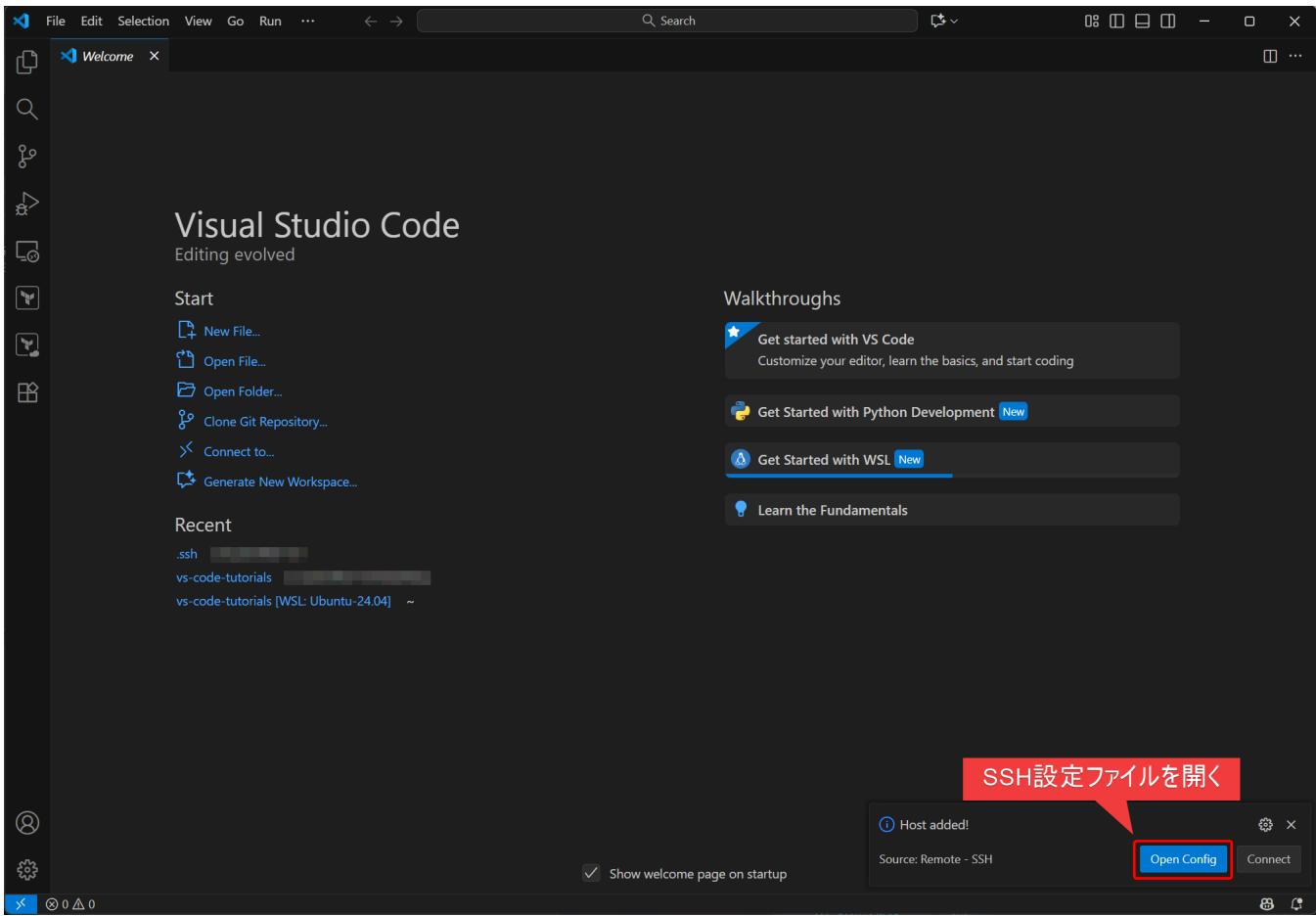


5. SSH access





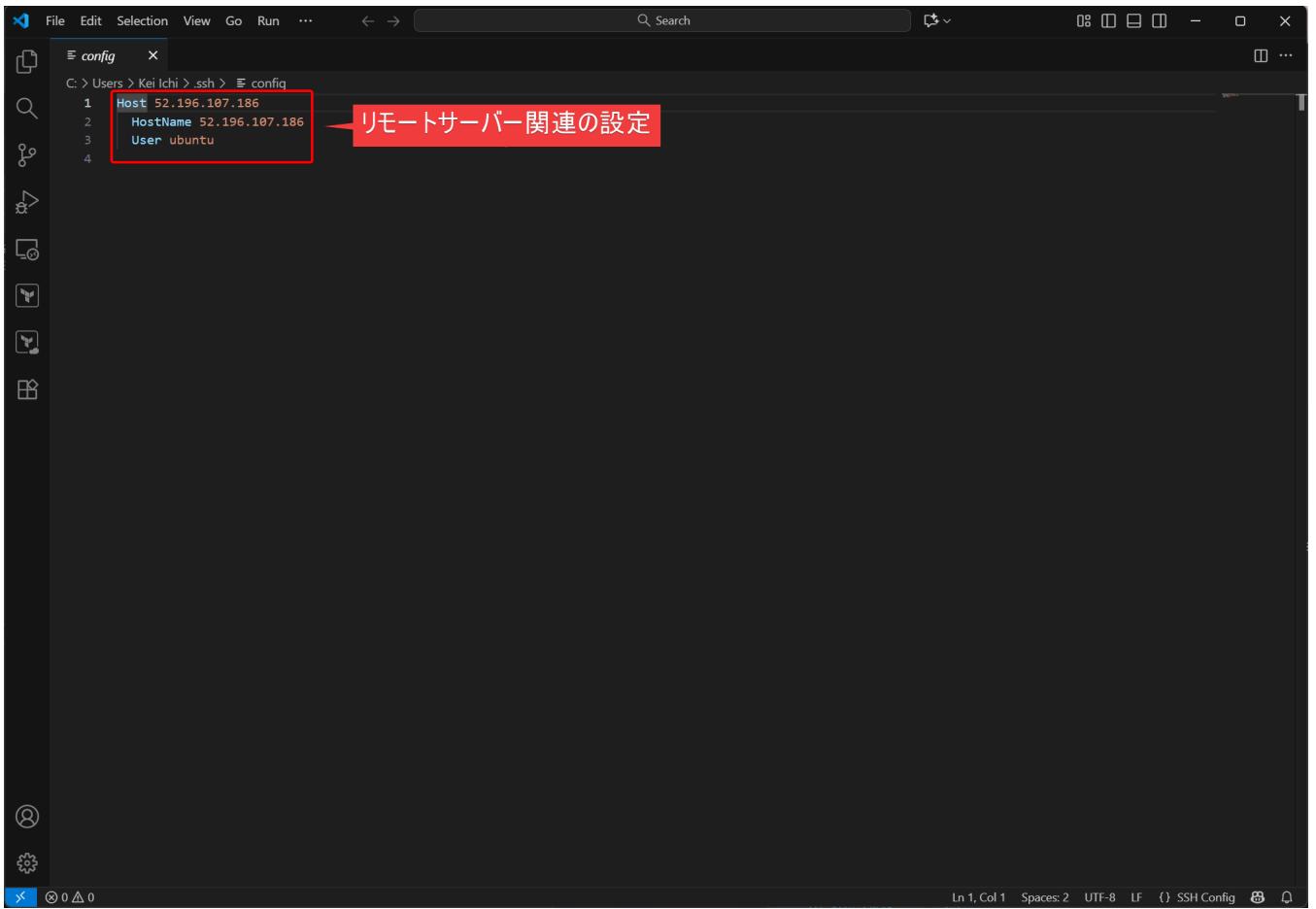




The screenshot shows a Windows Terminal window with two tabs: "Ubuntu-24.04" and "Windows PowerShell". The "Ubuntu-24.04" tab is active and shows a terminal session. The command `\.ssh> ls` is run, and the output is displayed in a table:

Mode	LastWriteTime	Length	Name
-a---	2025-11-12 18:29	110	config
-ar---	2025-11-11 20:26	387	vscode-tutorial-instance-key.pem

A red box highlights the directory path `Directory: \.ssh`. Another red box highlights the file `vscode-tutorial-instance-key.pem`. A red callout bubble points to this file with the text "SSHのキーを配置する" (Configure the SSH key).



File Edit Selection View Go Run ... ⏪ ⏩ Search 08 □ □ - □ ×

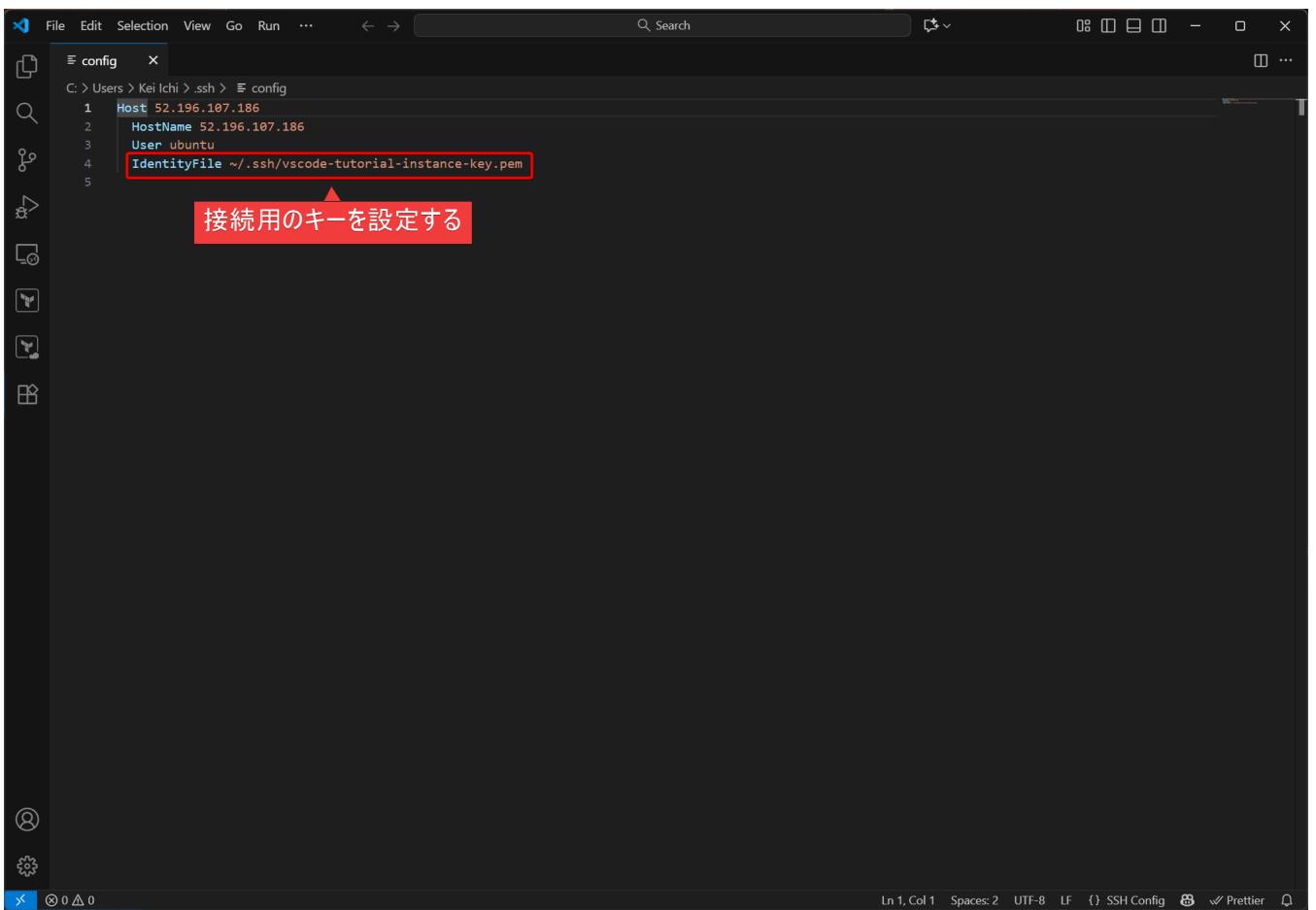
config x

C: > Users > Kei Ichi > ssh > config

```
1 Host 52.196.187.186
2   HostName 52.196.187.186
3   User ubuntu
4
```

リモートサーバー関連の設定

Ln 1, Col 1 Spaces: 2 UTF-8 LF { } SSH Config ⚙️ 🔍



File Edit Selection View Go Run ... ⏪ ⏩ Search 08 □ □ - □ ×

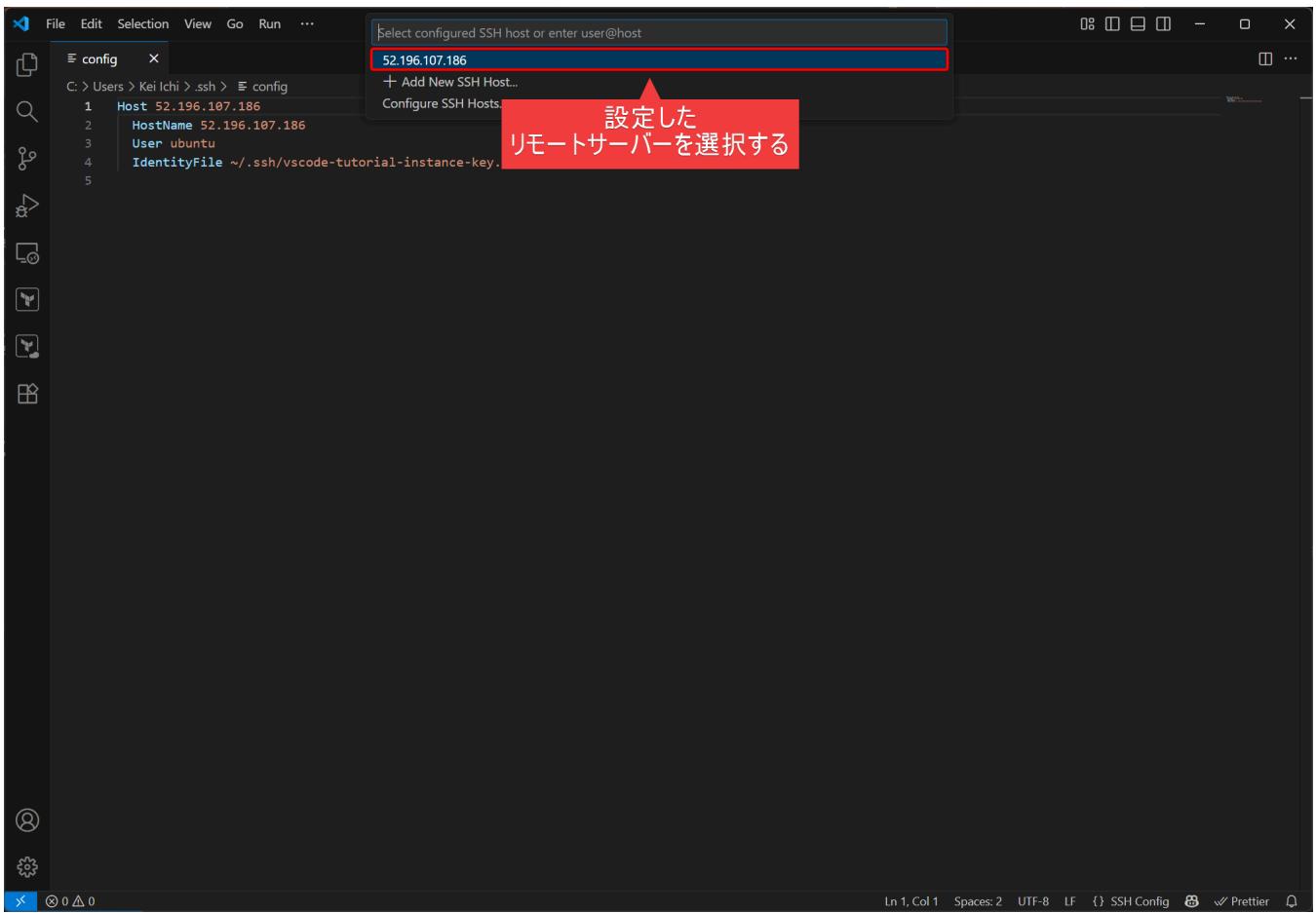
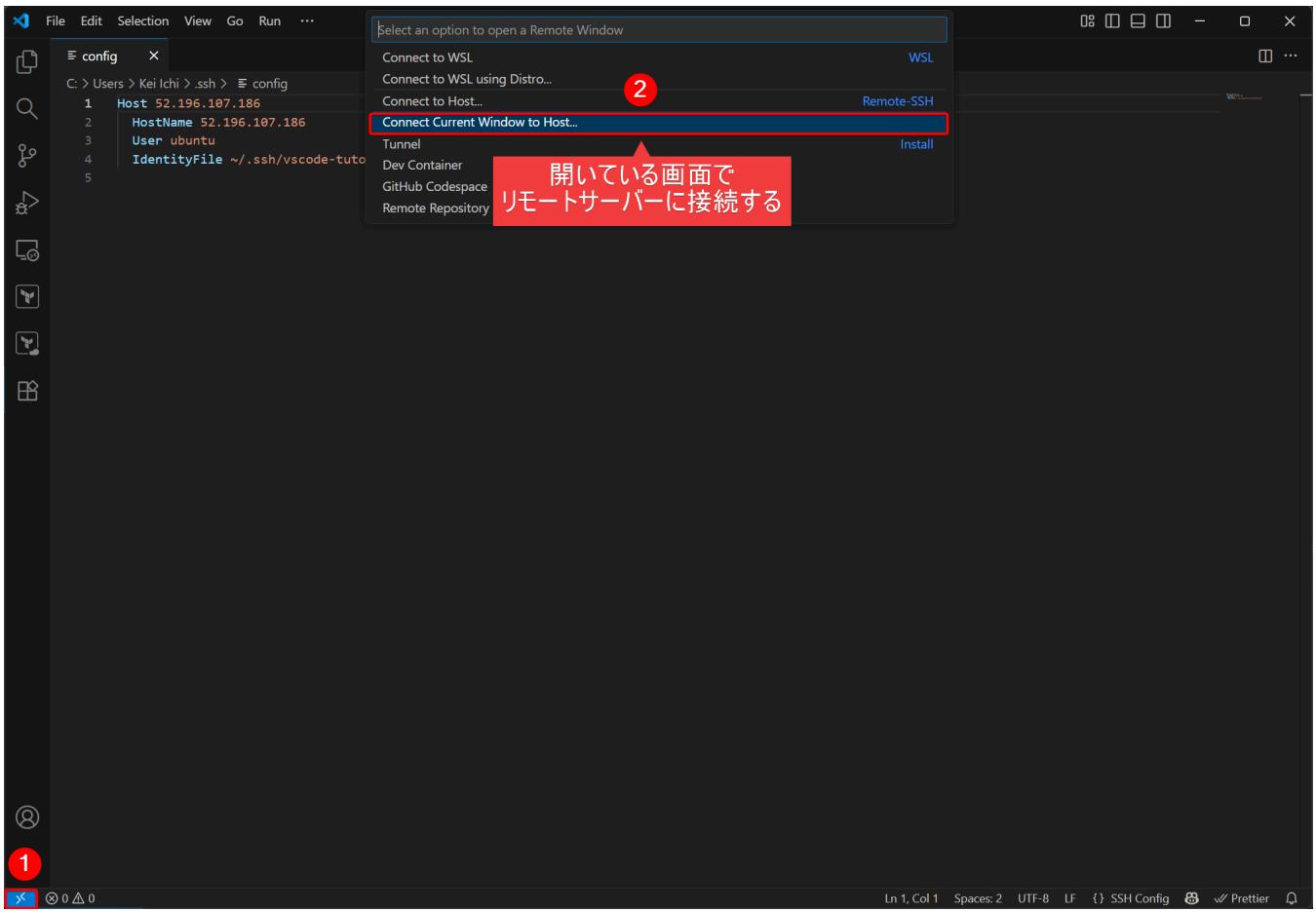
config x

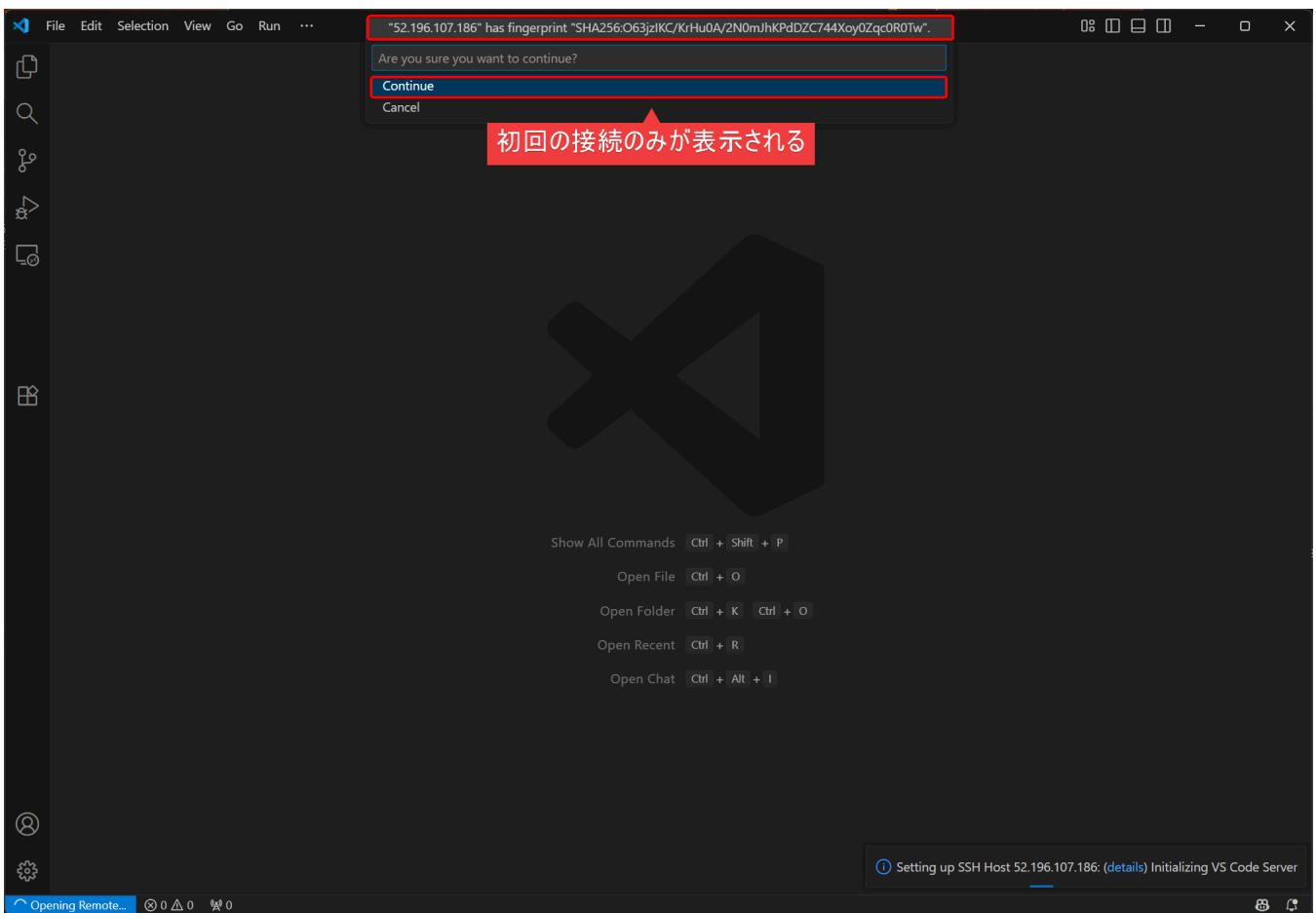
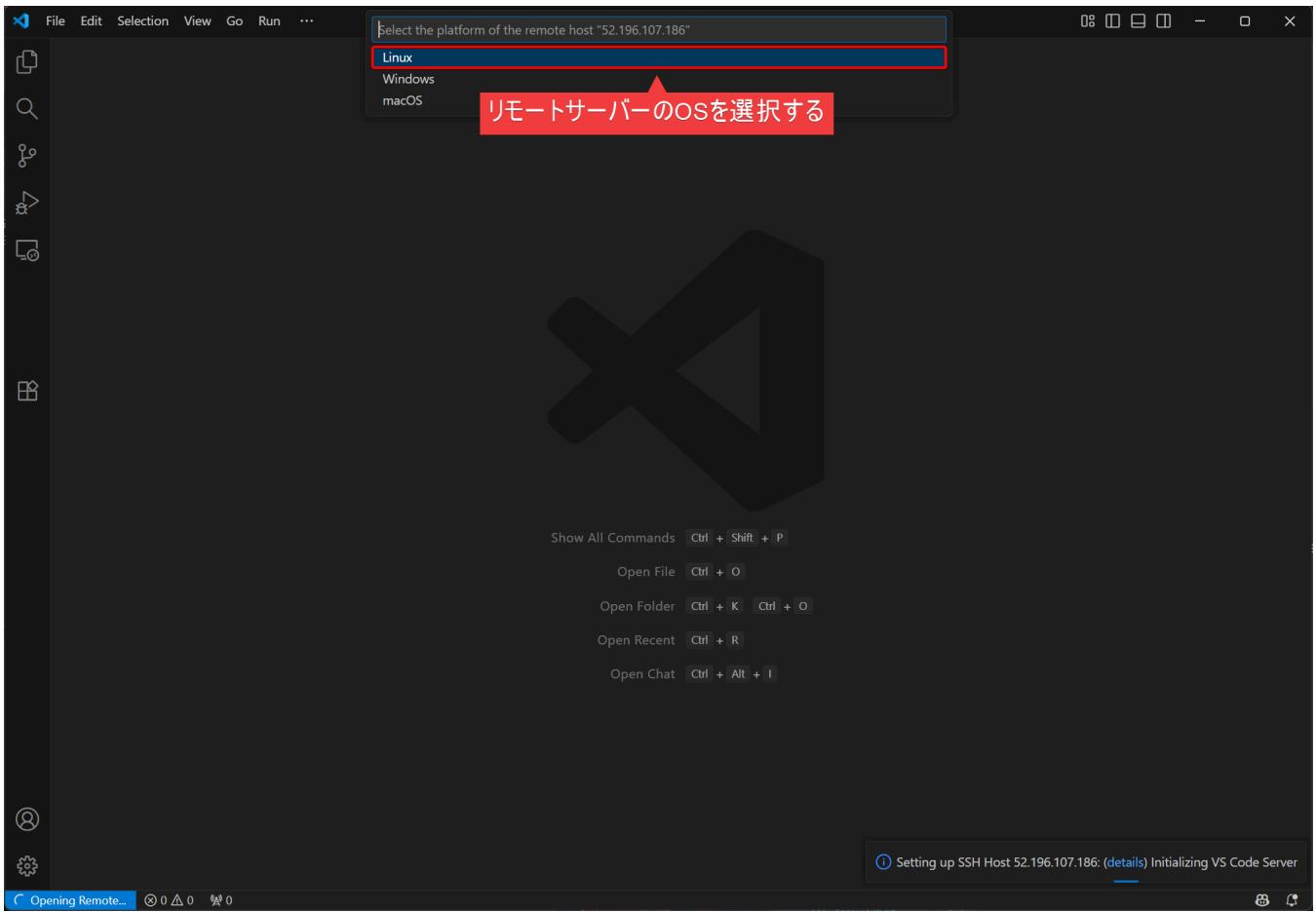
C: > Users > Kei Ichi > ssh > config

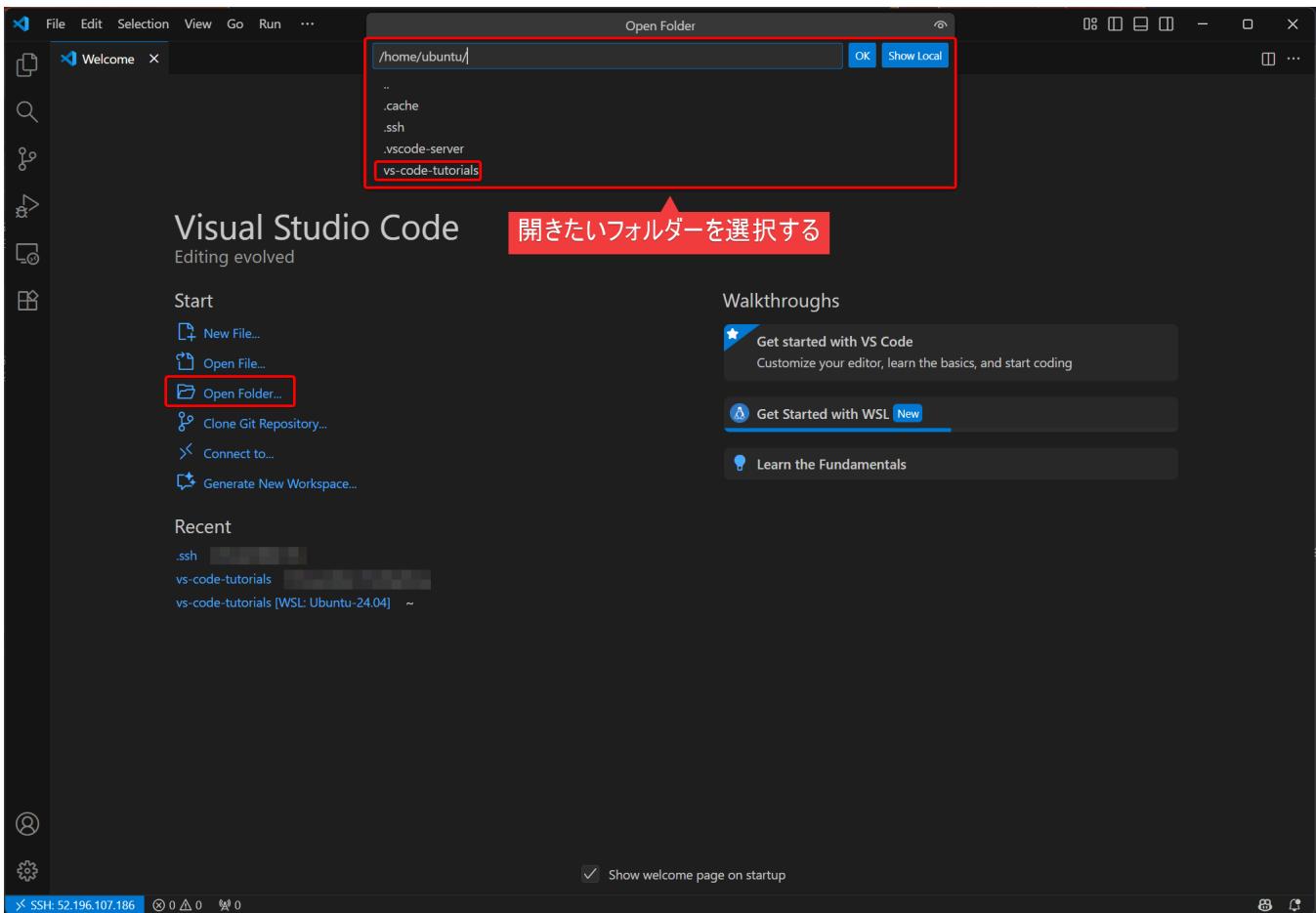
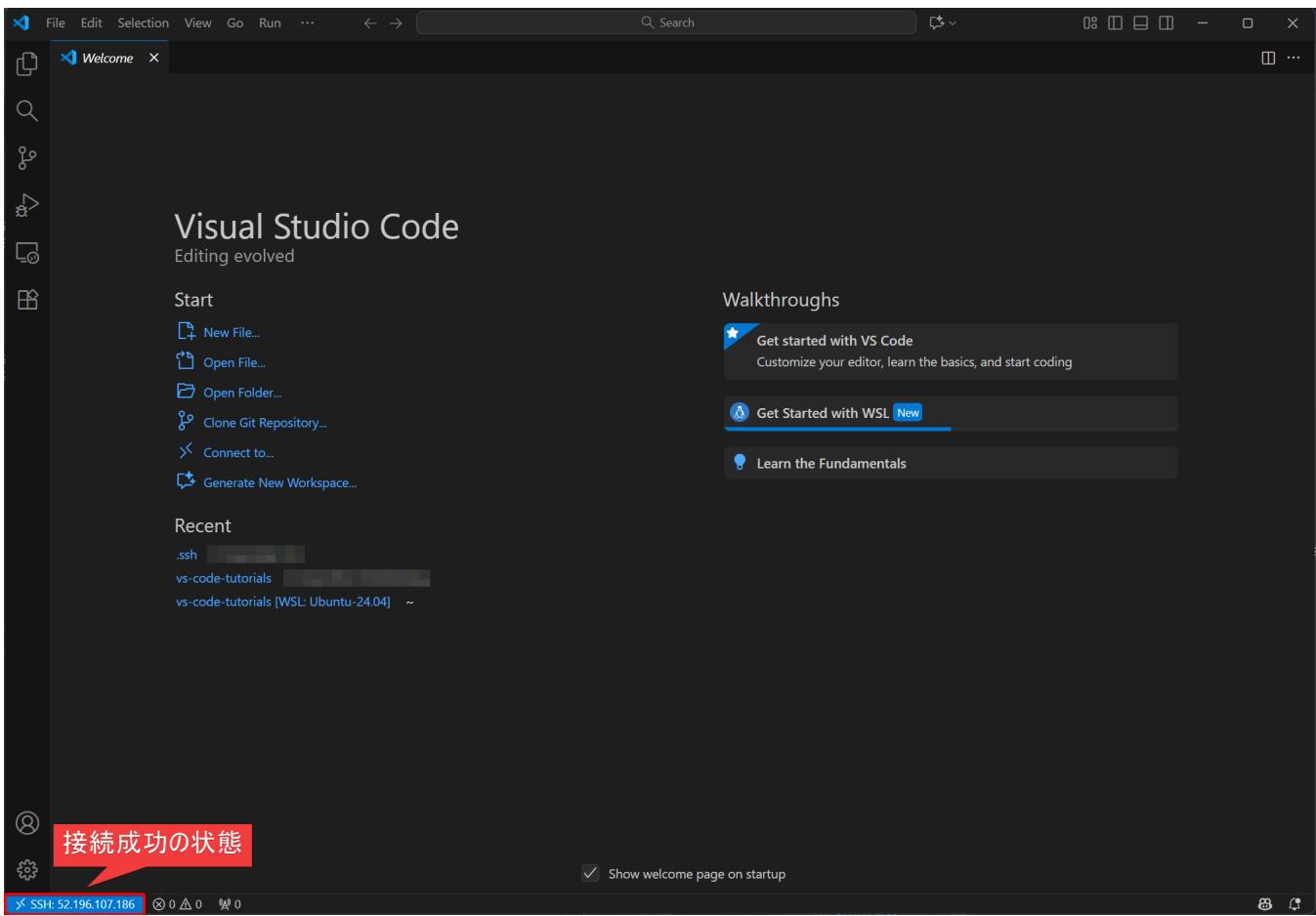
```
1 Host 52.196.187.186
2   HostName 52.196.187.186
3   User ubuntu
4   IdentityFile ~/.ssh/vscode-tutorial-instance-key.pem
5
```

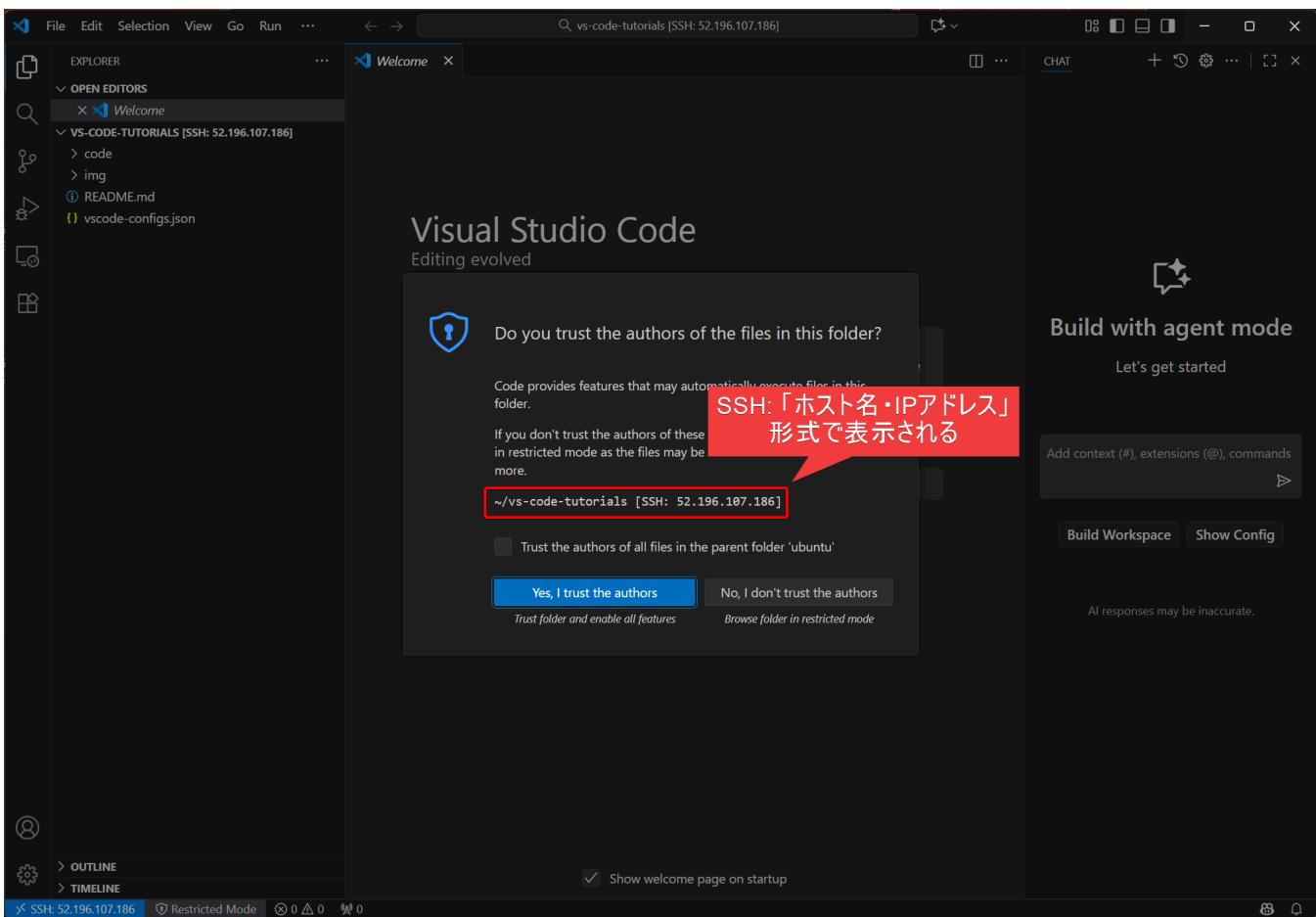
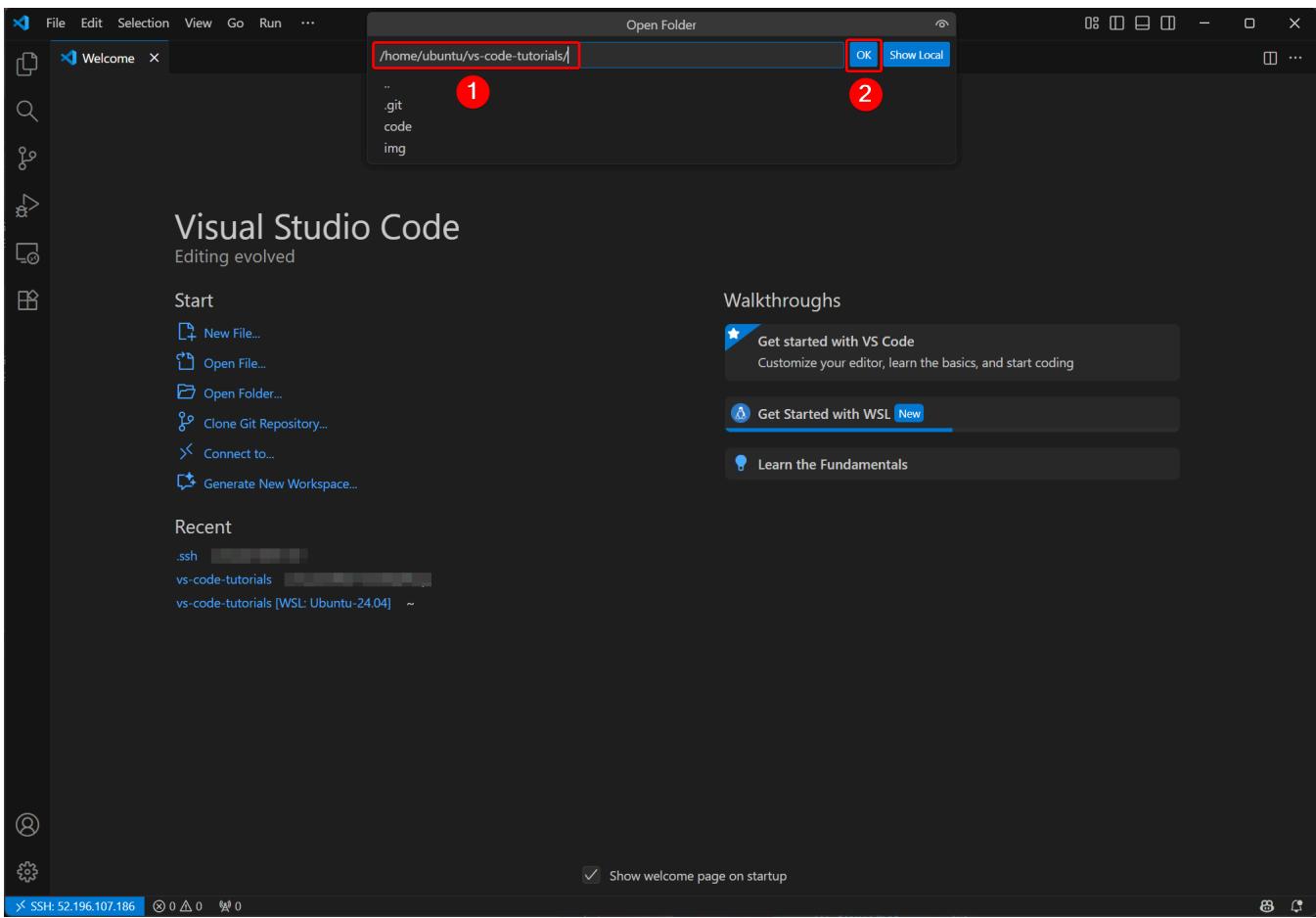
接続用のキーを設定する

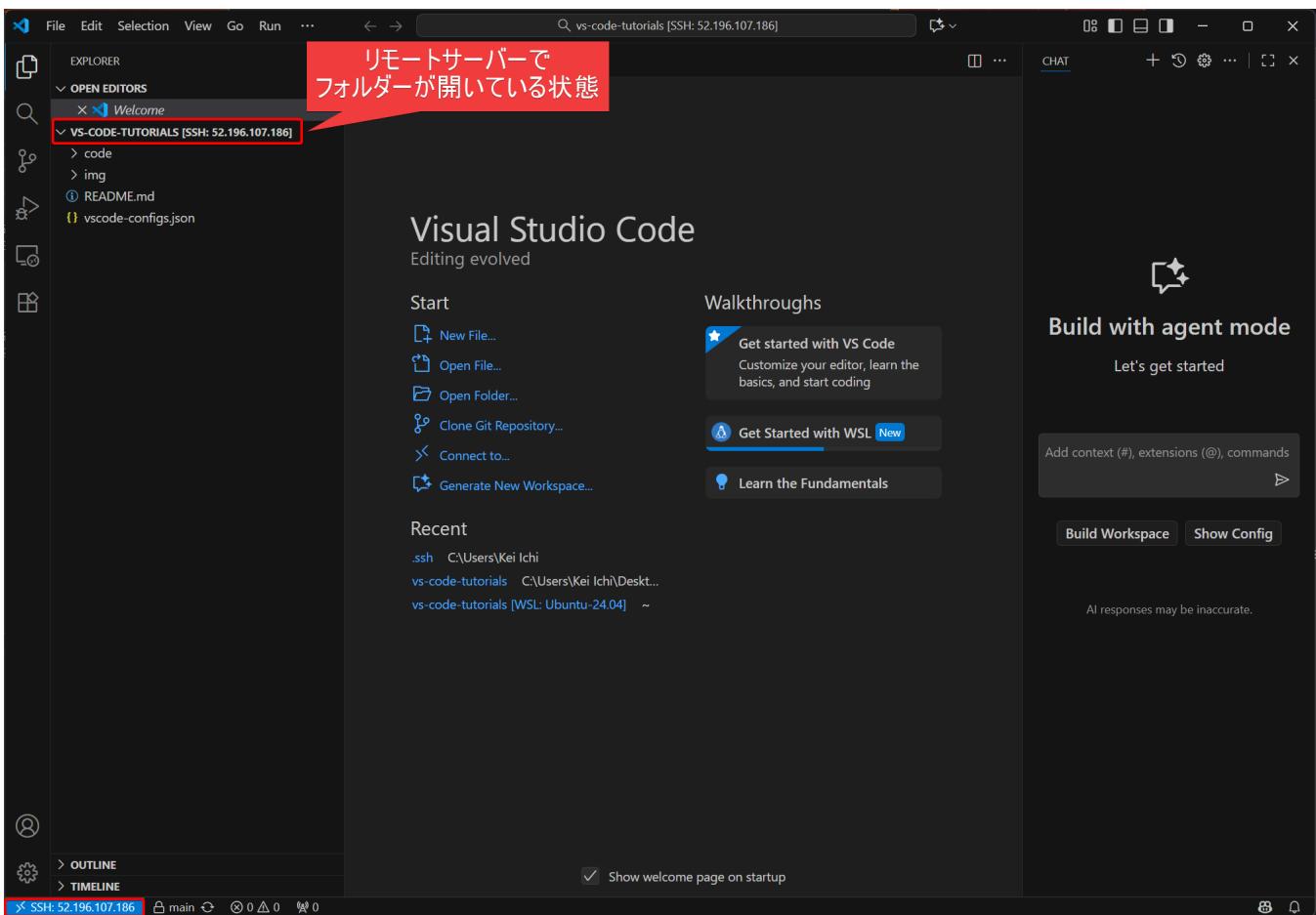
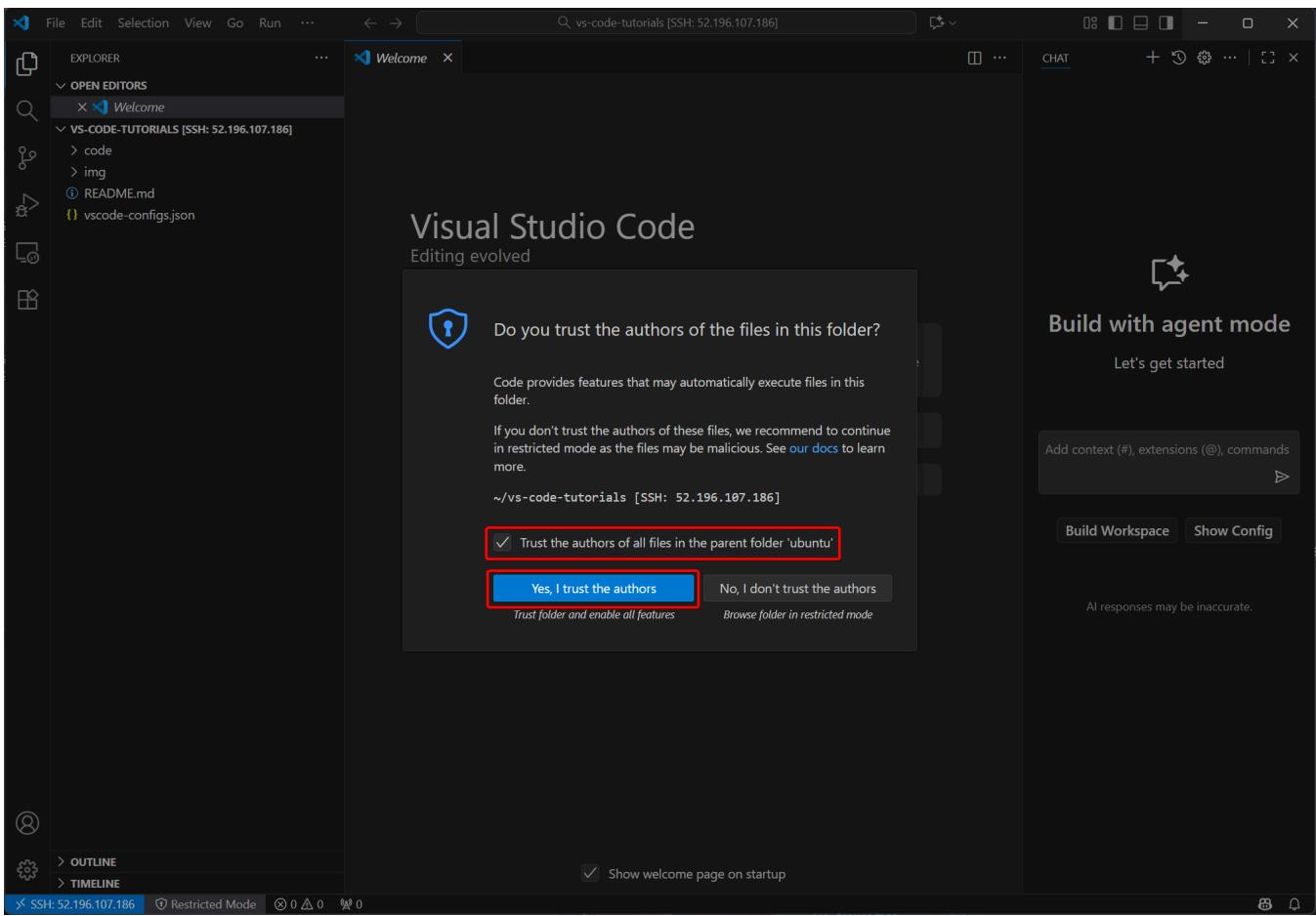
Ln 1, Col 1 Spaces: 2 UTF-8 LF { } SSH Config ⚙️ 🔍











```
main.tf
code > terraform > main.tf
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 data "aws_ami" "ubuntu" {
17   most_recent = true
18
19   filter {
20     name  = "name"
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
22   }
23
24   owners = ["099720109477"] # Canonical
25 }
26
27 resource "aws_instance" "app_server" {
28   ami           = data.aws_ami.ubuntu.id
29   instance_type = "t2.micro"
30
31   tags = {
32     Name = "learn-terraform"
33   }
34 }
35
36 data "aws_ami" "ubuntu" {
37   most_recent = true
38
39   filter {
40     name  = "name"
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42   }
43 }
```

SSH: 52.196.107.186

EXTENSIONS

Search Extensions in Marketplace

LOCAL - INSTALLED

- HashiCorp Terraform
- Markdown All in One
- Prettier - Code formatter
- PyLance
- Python
- Python Debugger
- Python Environments
- Rainbow CSV
- YAML

RECOMMENDED

MCP SERVERS

SSH: 52.196.107.186 - INSTALLED

RECOMMENDED

MCP SERVERS

```
main.tf
code > terraform > main.tf
1 terraform {
2   required_providers {
3     aws = {
4       source  = "hashicorp/aws"
5       version = "~> 5.92"
6     }
7   }
8
9   required_version = ">= 1.2"
10 }
11
12 provider "aws" {
13   region = "us-west-2"
14 }
15
16 data "aws_ami" "ubuntu" {
17   most_recent = true
18
19   filter {
20     name  = "name"
21     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
22   }
23
24   owners = ["099720109477"] # Canonical
25 }
26
27 resource "aws_instance" "app_server" {
28   ami           = data.aws_ami.ubuntu.id
29   instance_type = "t2.micro"
30
31   tags = {
32     Name = "learn-terraform"
33   }
34 }
35
36 data "aws_ami" "ubuntu" {
37   most_recent = true
38
39   filter {
40     name  = "name"
41     values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
42   }
43 }
```

SSH: 52.196.107.186

This screenshot shows the Visual Studio Code interface connected to a remote server via SSH (52.196.107.186). The left sidebar displays a list of installed extensions, including HashiCorp Terraform, Markdown All in One, Prettier - Code formatter, Pylance, Python, Python Debugger, Python Environments, Rainbow CSV, and YAML. A red callout box highlights the 'HashiCorp Terraform' extension. Another red callout box highlights the 'SSH: 52.196.107.186 - INSTALLED' entry in the sidebar, with the text 'リモートサーバーのみにインストールされた拡張機能一覧' (List of extensions installed only on the remote server).

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 5.92"
    }
  }

  required_version = ">= 1.2"
}

provider "aws" {
  region = "us-west-2"
}

data "aws_ami" "ubuntu" {
  most_recent = true

  filter {
    name   = "name"
    values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
  }
}

resource "aws_instance" "app_server" {
  ami           = data.aws_ami.ubuntu.id
  instance_type = "t2.micro"

  tags = {
    Name = "learn-terraform"
  }
}

data "aws_ami" "ubuntu" {
  most_recent = true
```

This screenshot shows the Visual Studio Code interface connected to a remote server via SSH (52.196.107.186). The left sidebar shows the same list of installed extensions. A red callout box highlights the '(オプション)' (Optional) link in the top right corner of the 'Select an option to open a Remote Window' dropdown menu. Another red callout box highlights the 'Close Remote Connection' option in the same menu, with the number '2' indicating it is the second item.

Select an option to open a Remote Window

- Connect to Host... Remote SSH
- Connect Current Window to Host... WSL
- Connect to WSL Git
- Connect to WSL using Distro... Remote Repositories
- Continue Working in New Local Clone
- Continue Working in vscode.dev
- Tunnel Install
- Dev Container
- GitHub Codespace
- Remote Repository

Close Remote Connection 2

```
provider "aws" {
  region = "us-west-2"
}

data "aws_ami" "ubuntu" {
  most_recent = true

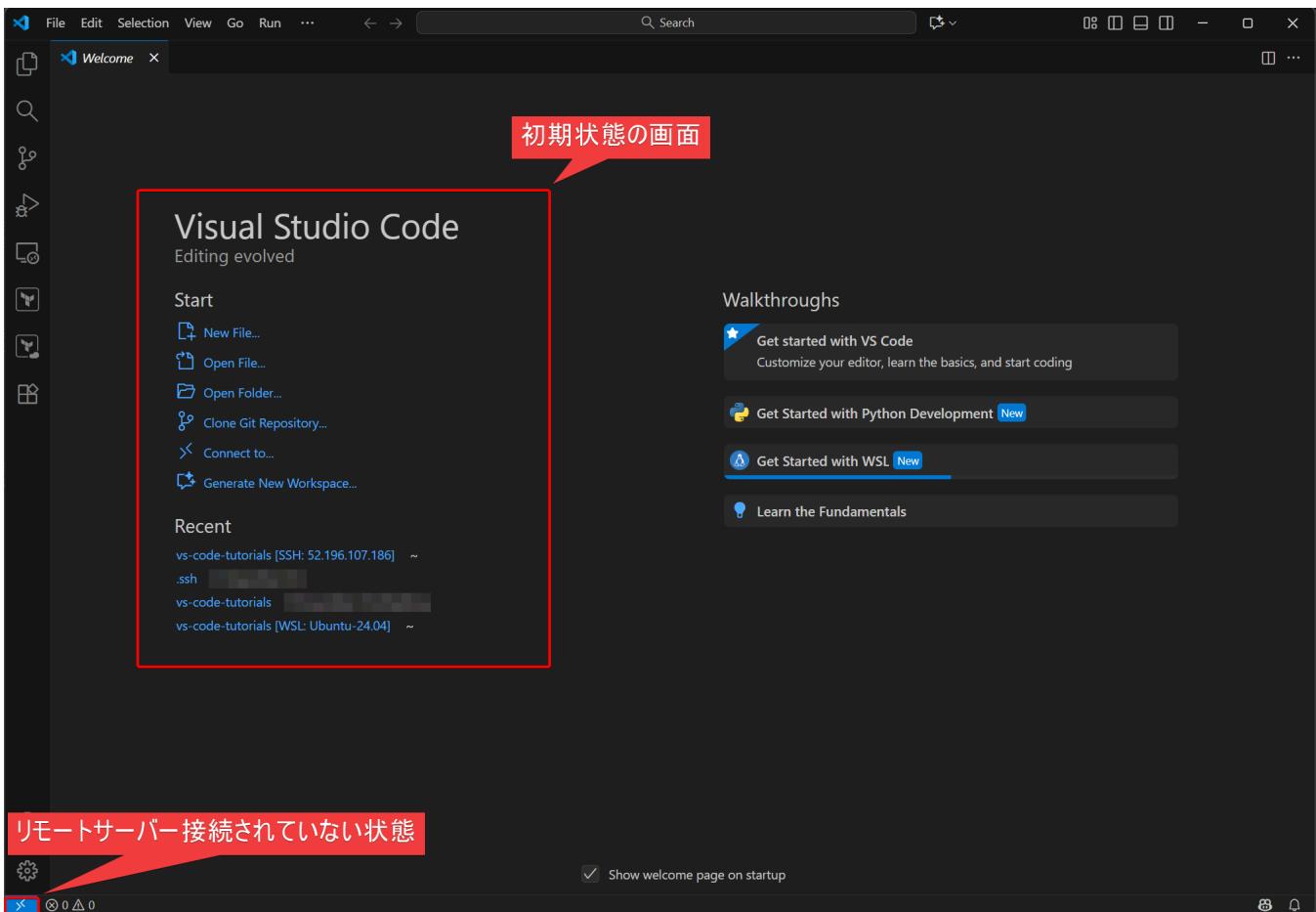
  filter {
    name   = "name"
    values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
  }
}

resource "aws_instance" "app_server" {
  ami           = data.aws_ami.ubuntu.id
  instance_type = "t2.micro"

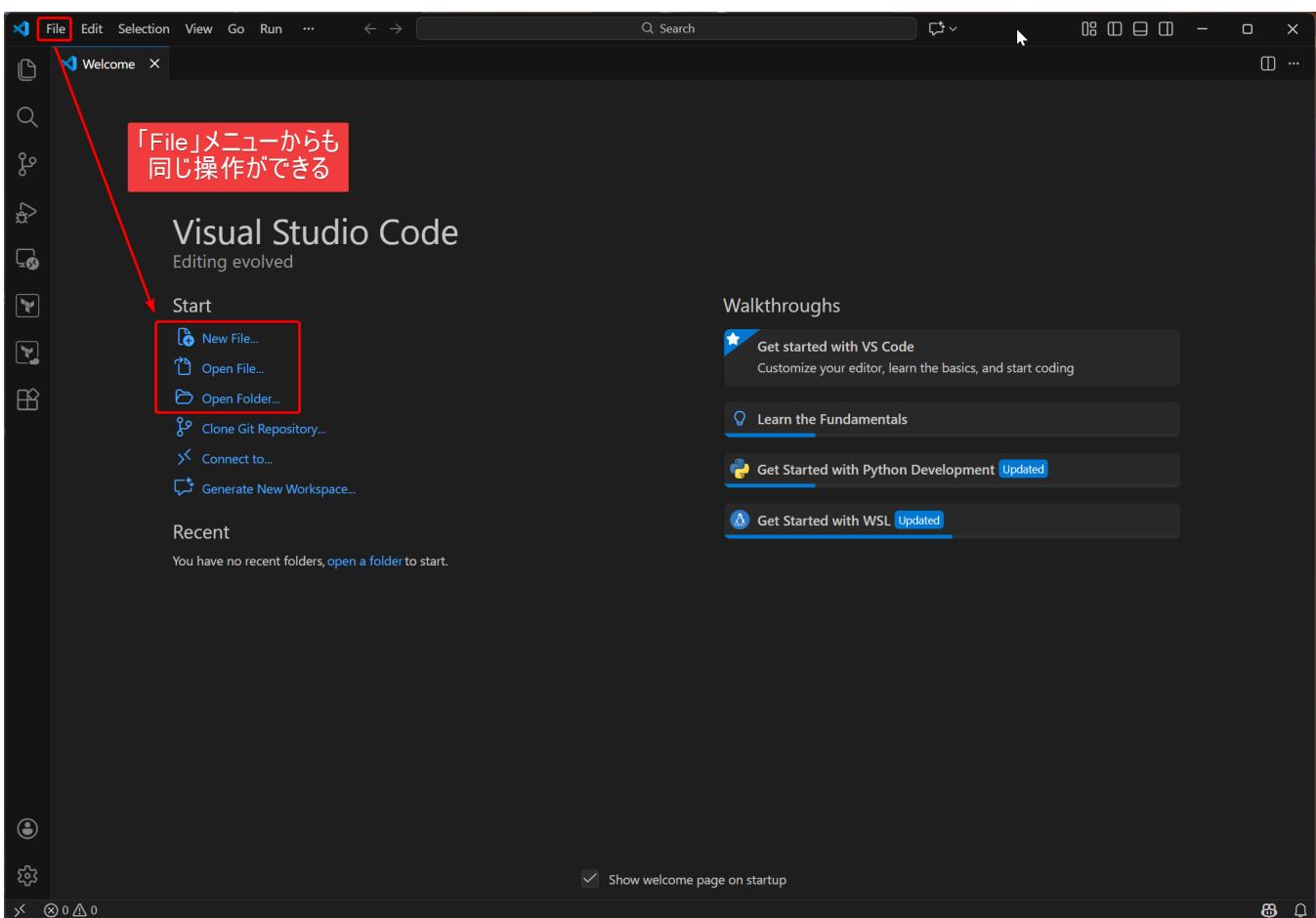
  tags = {
    Name = "learn-terraform"
  }
}

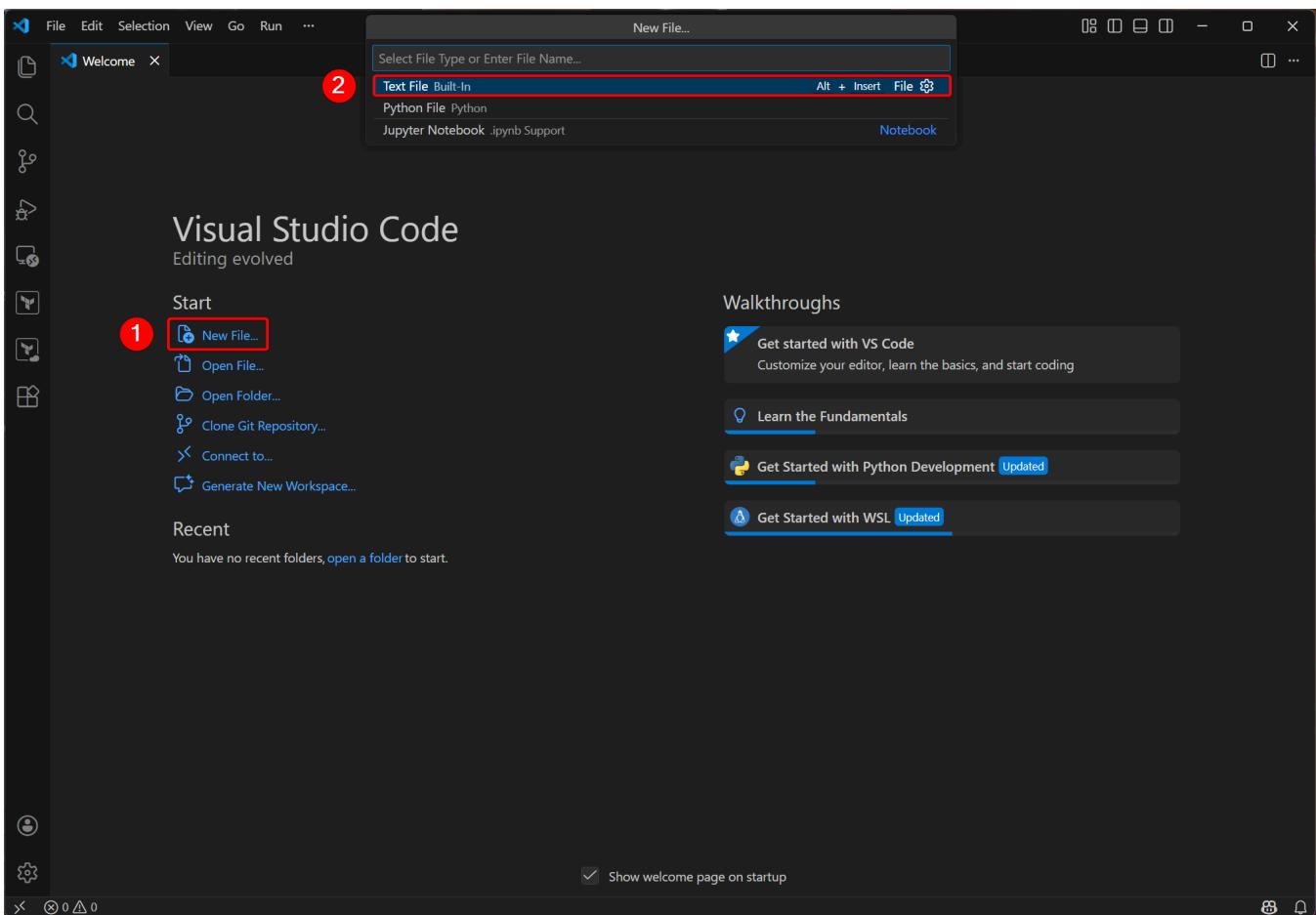
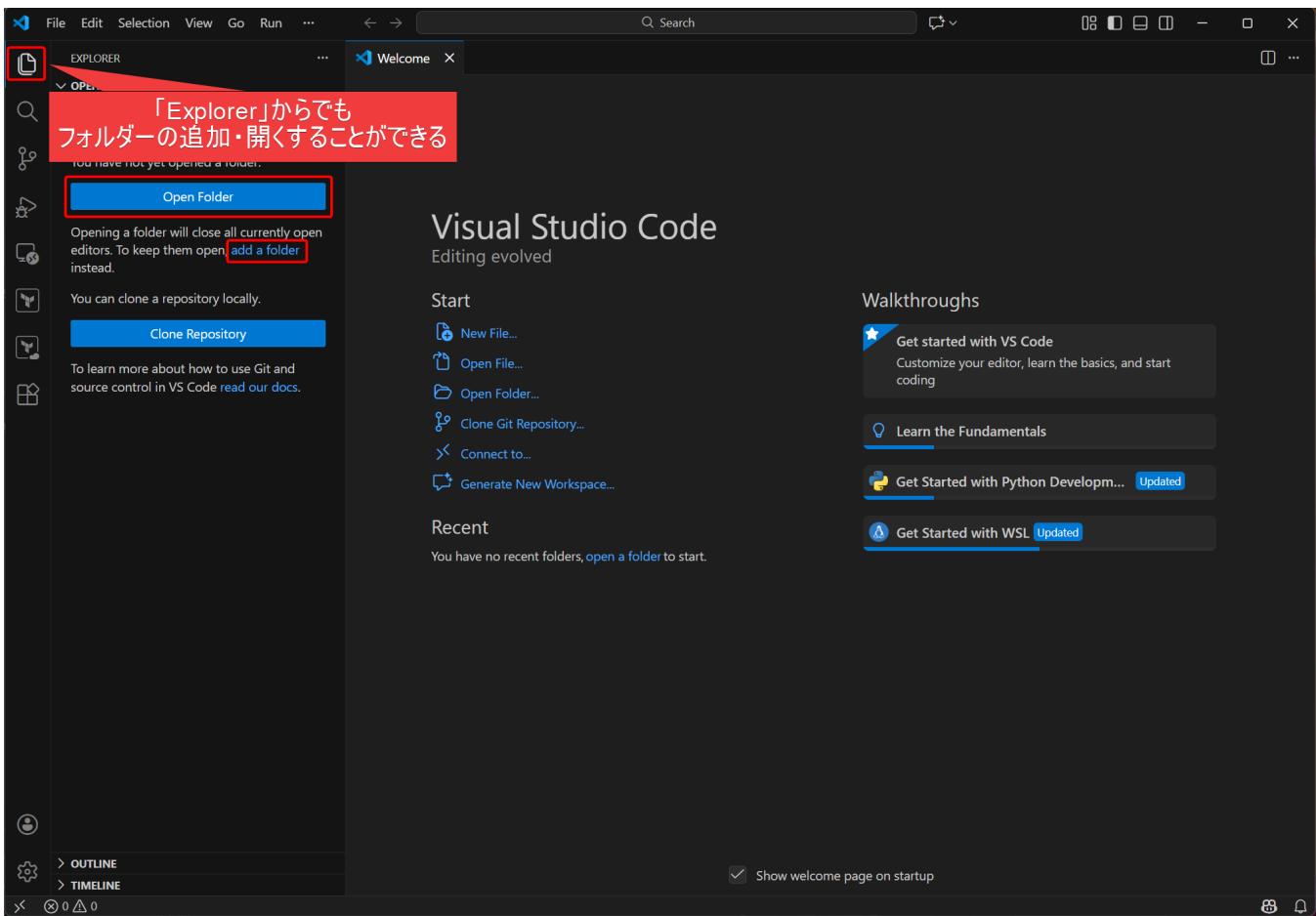
data "aws_ami" "ubuntu" {
  most_recent = true

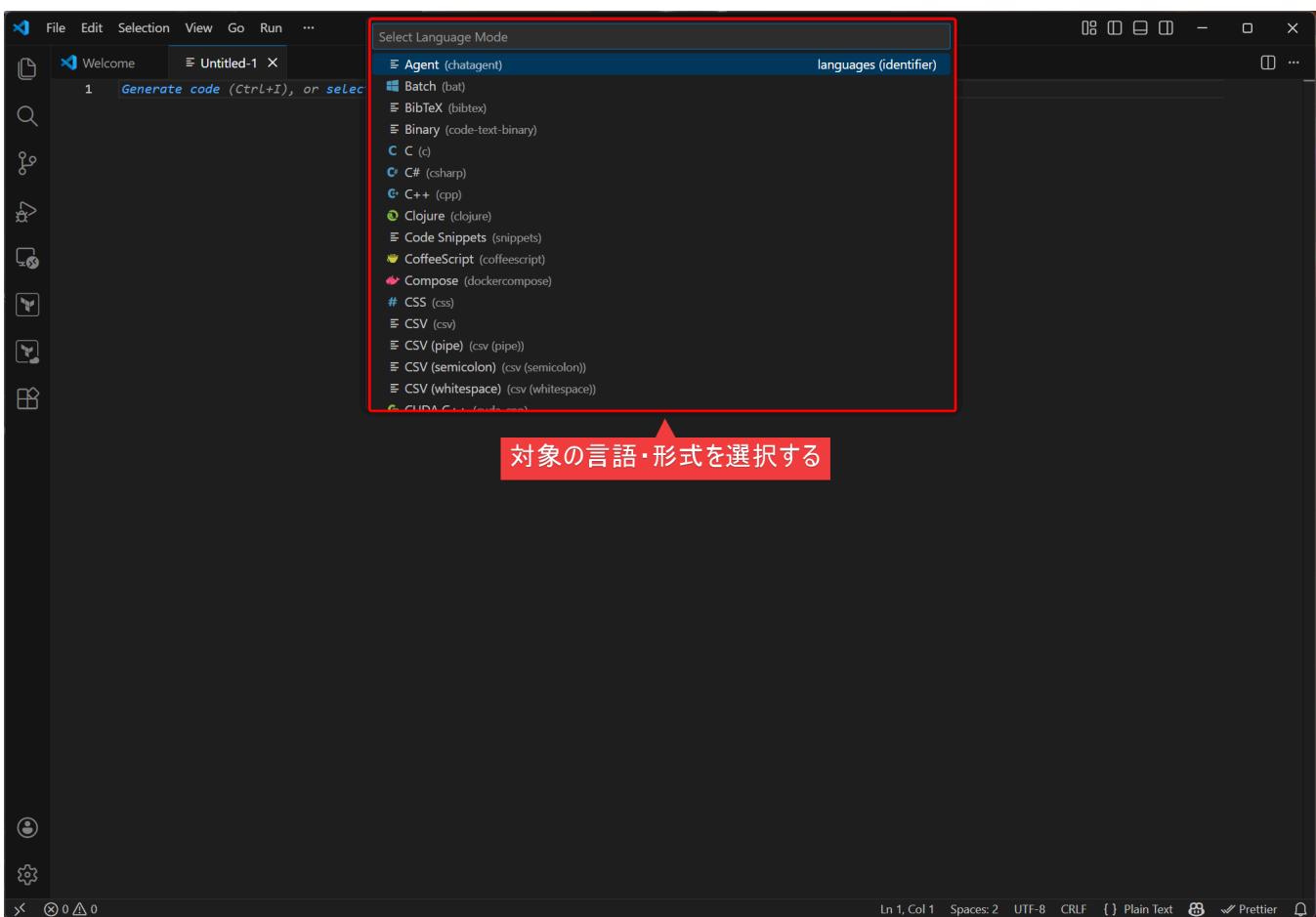
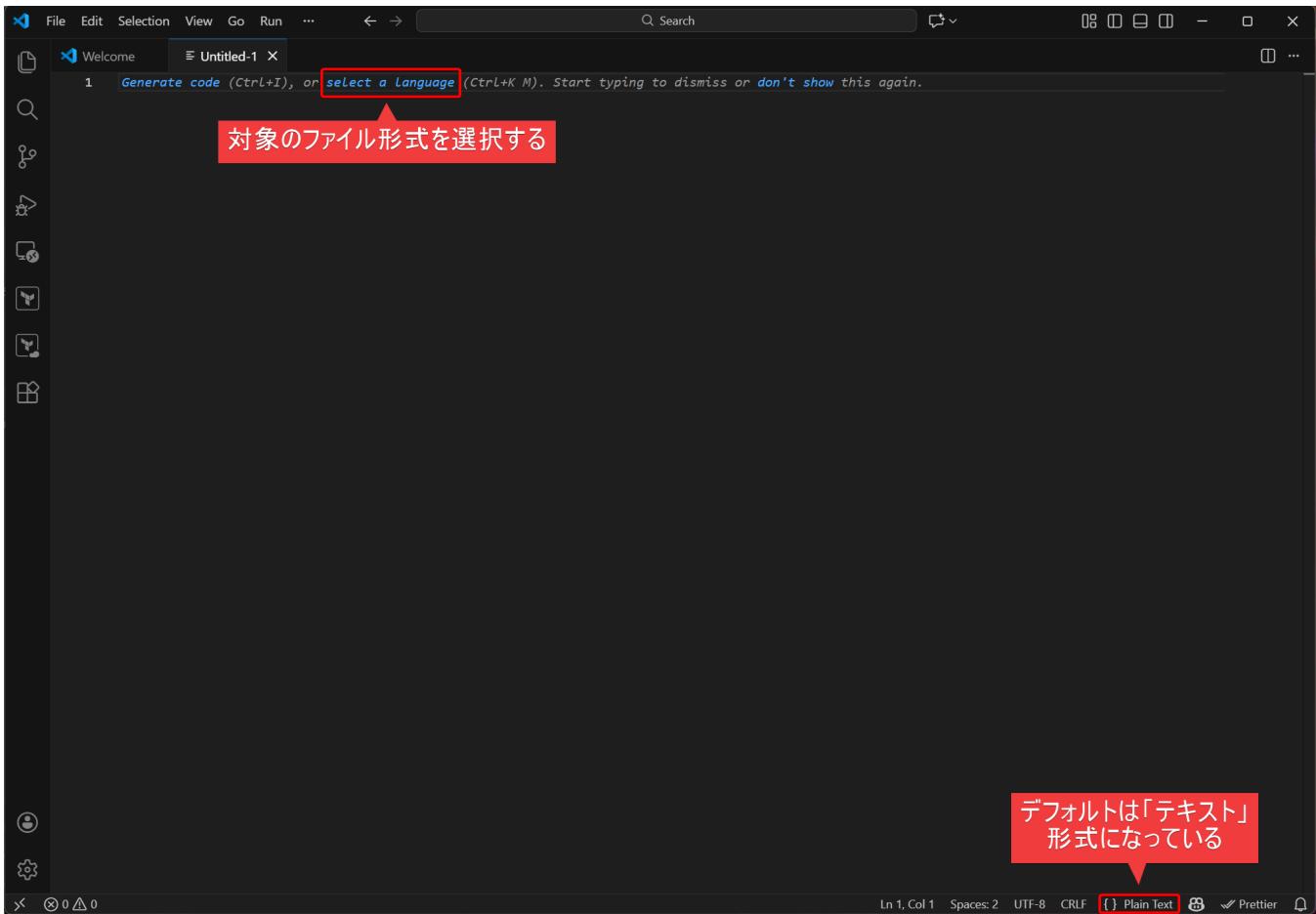
  filter {
    name   = "name"
    values = ["ubuntu/images/hvm-ssd-gp3/ubuntu-noble-24.04-amd64-server-*"]
```

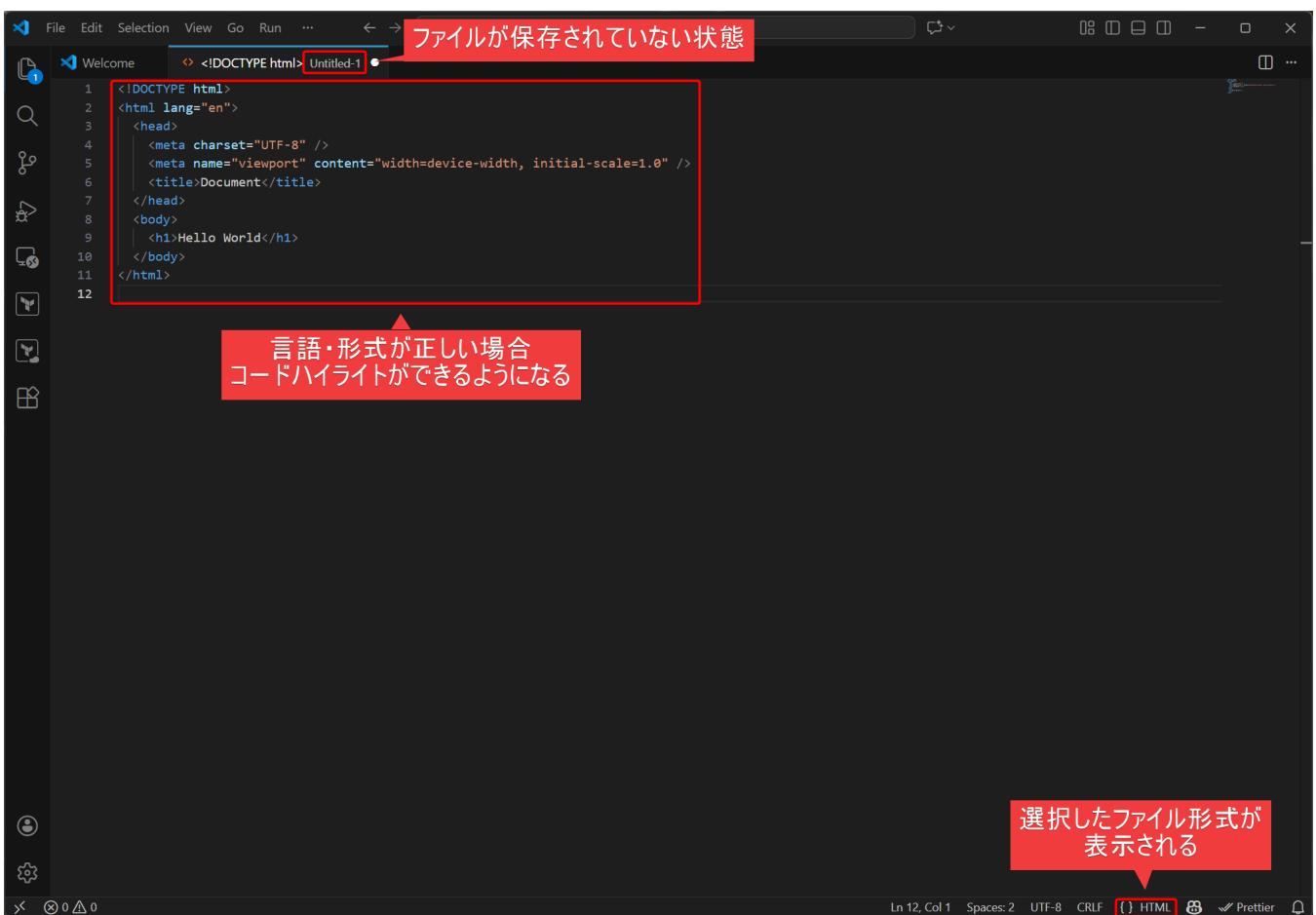
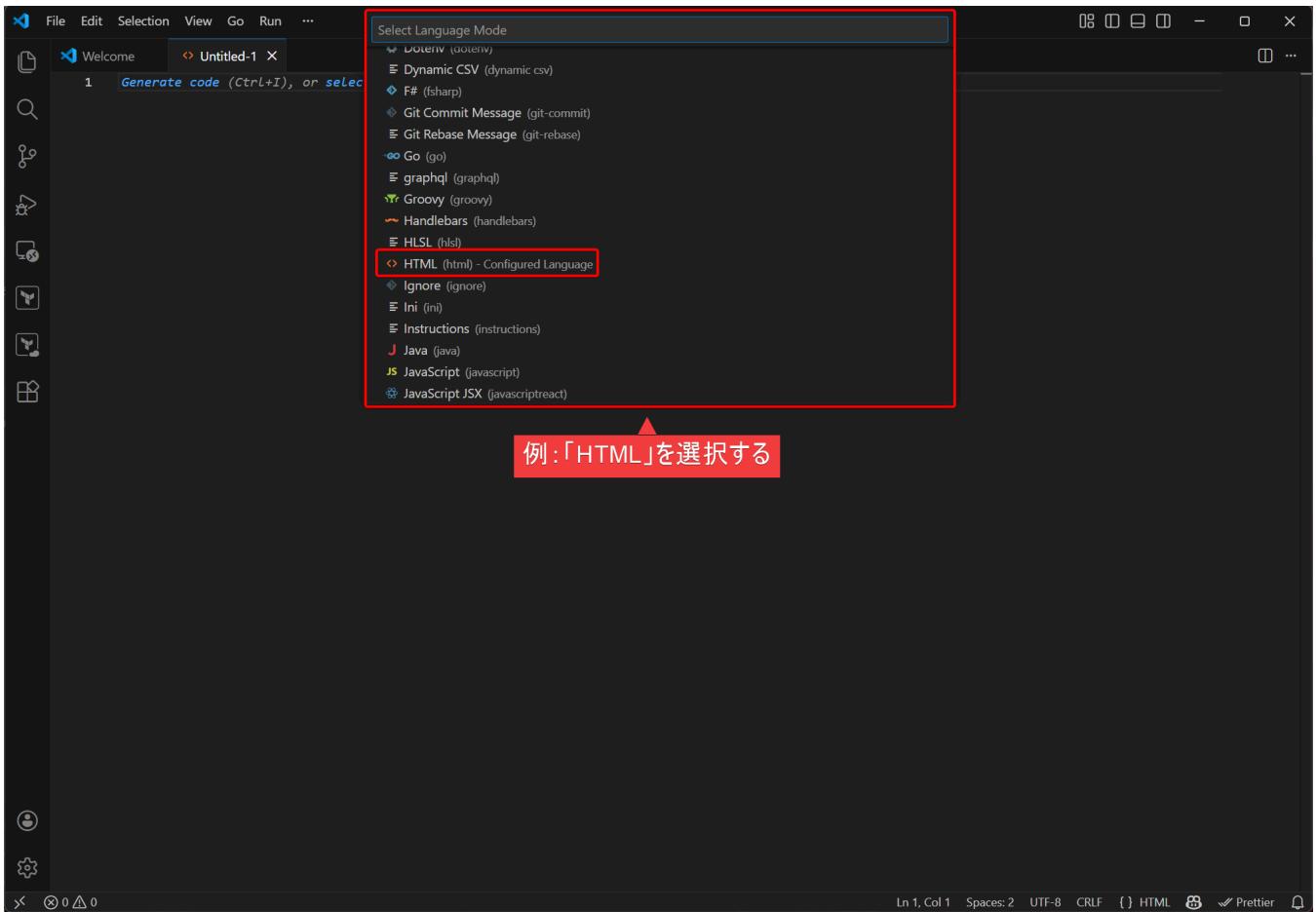


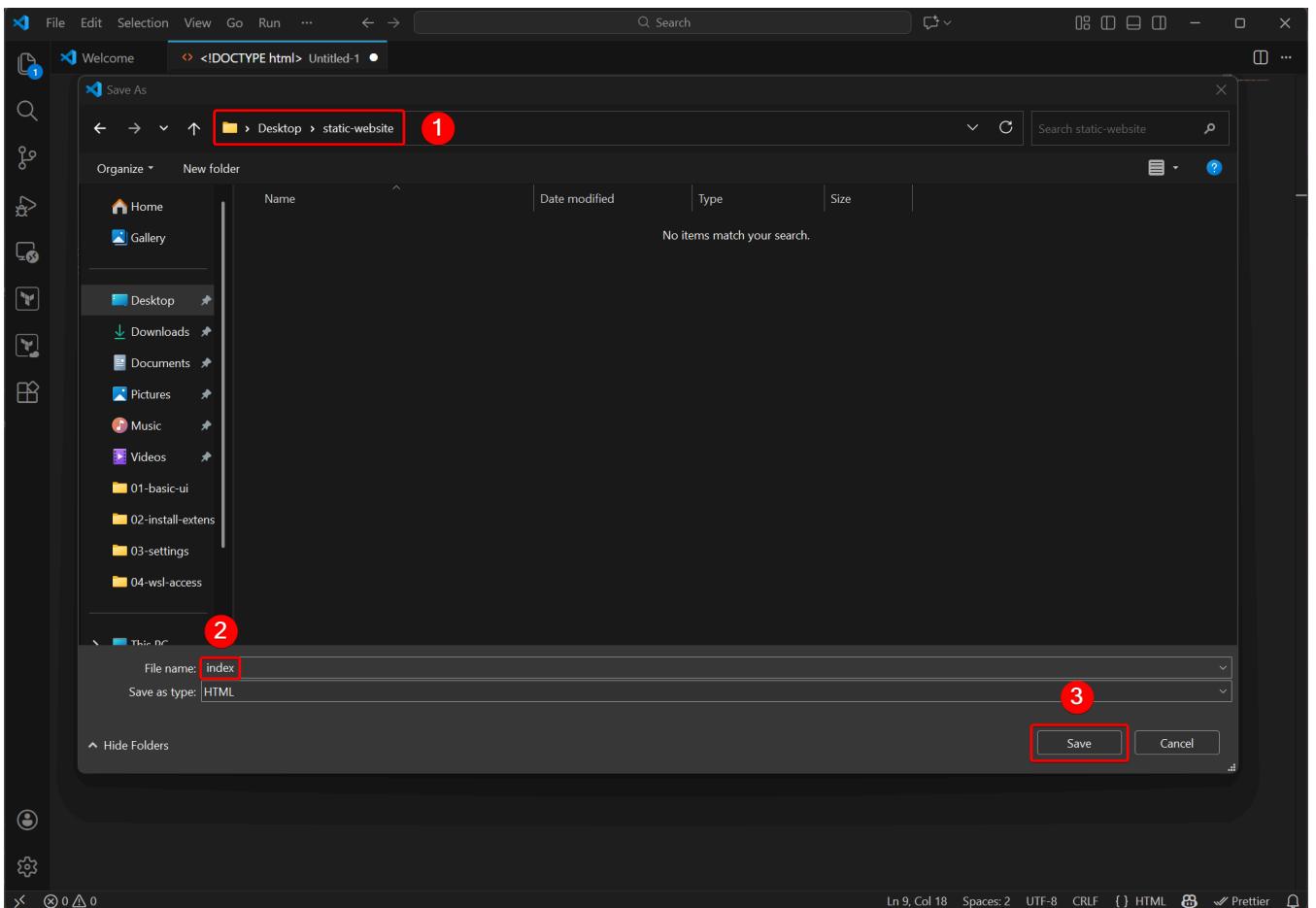
6. Edit code







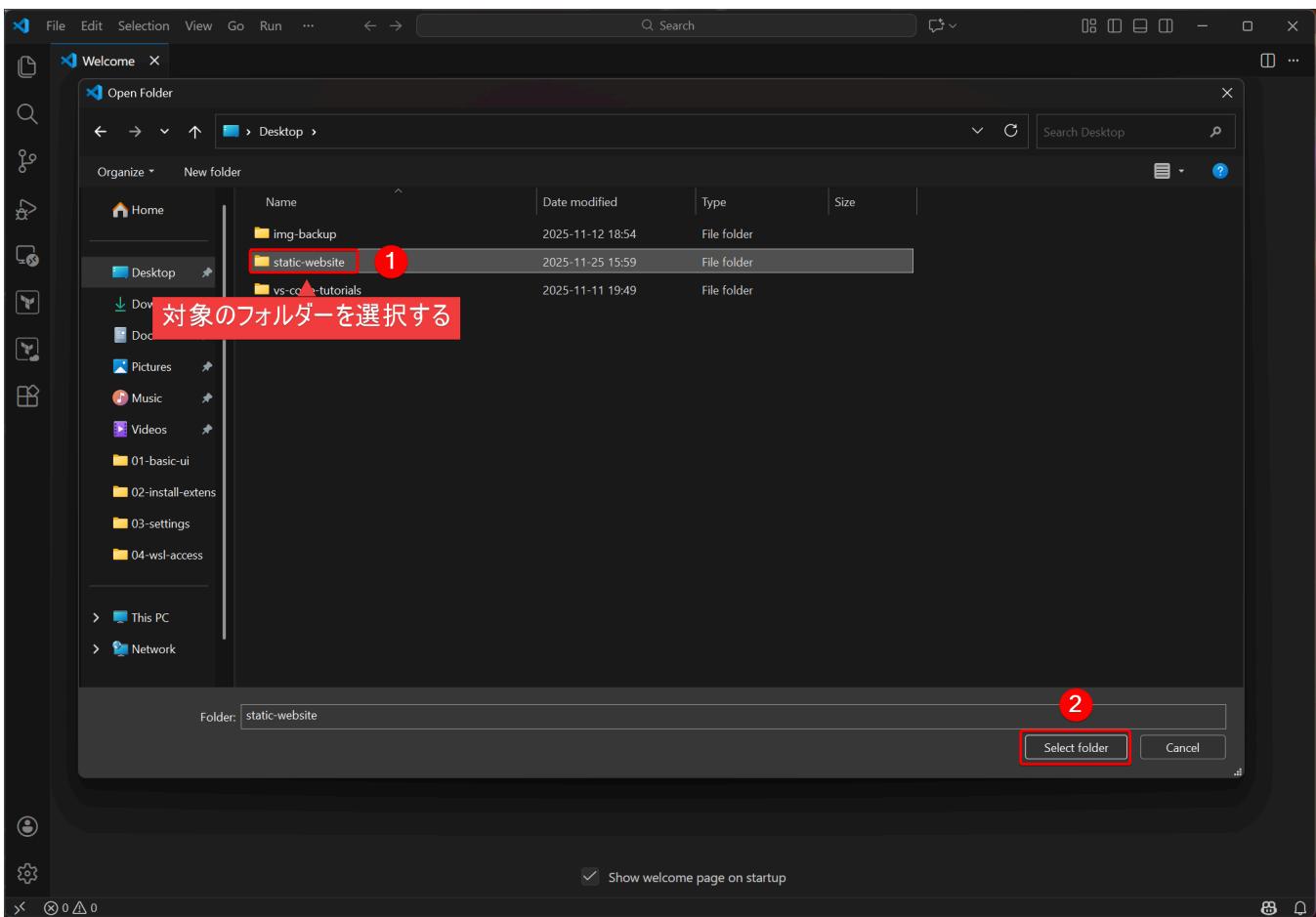
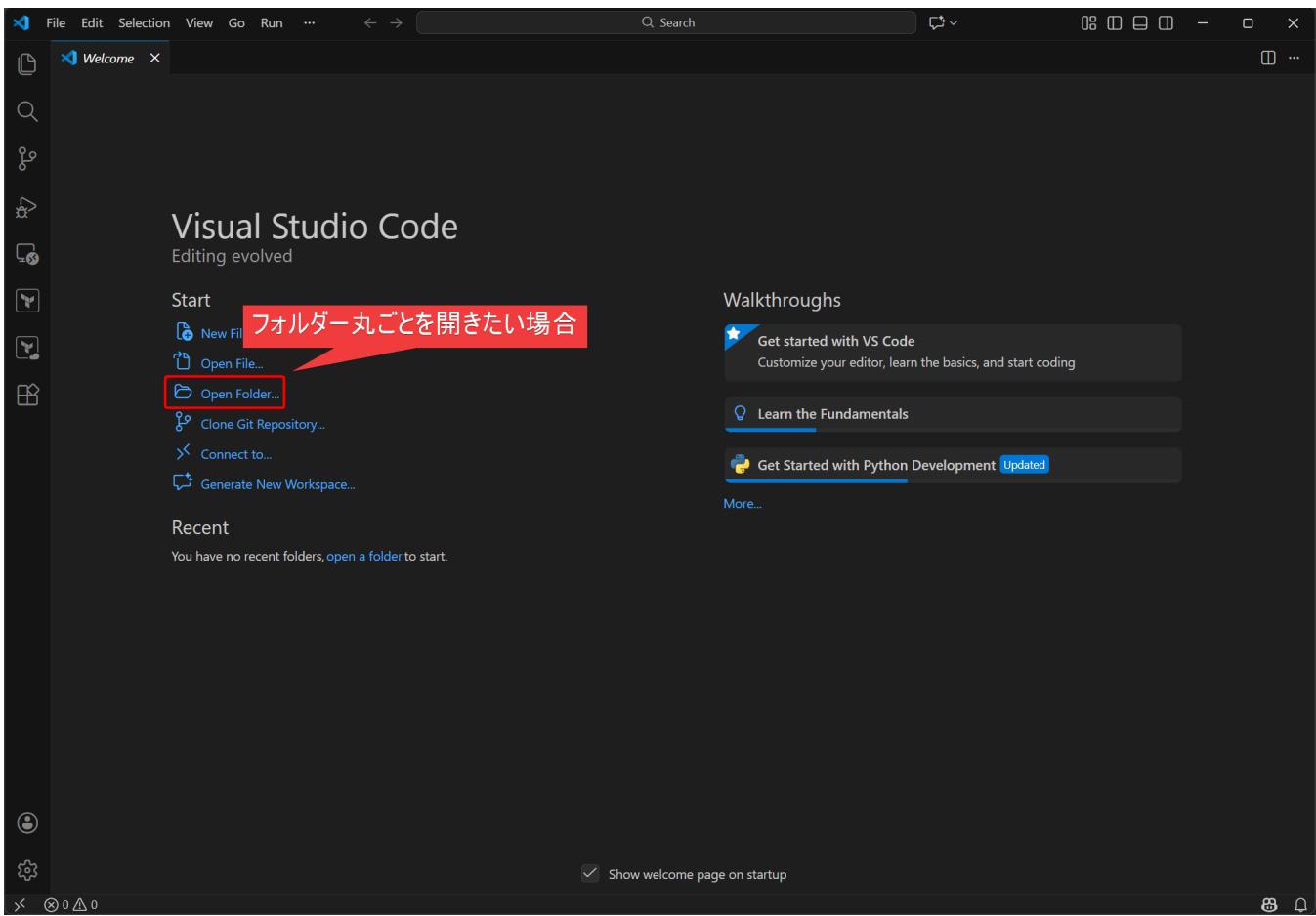


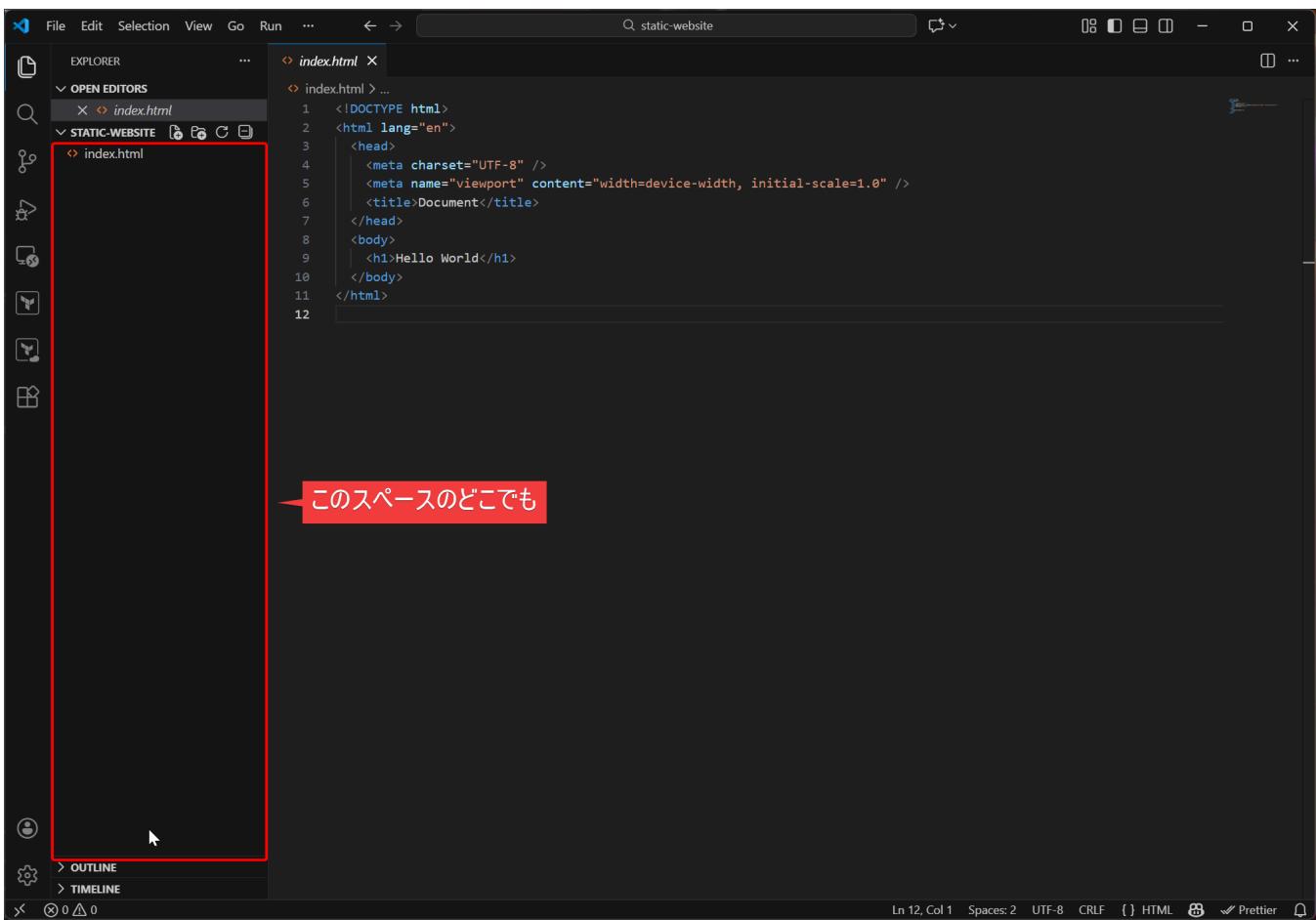
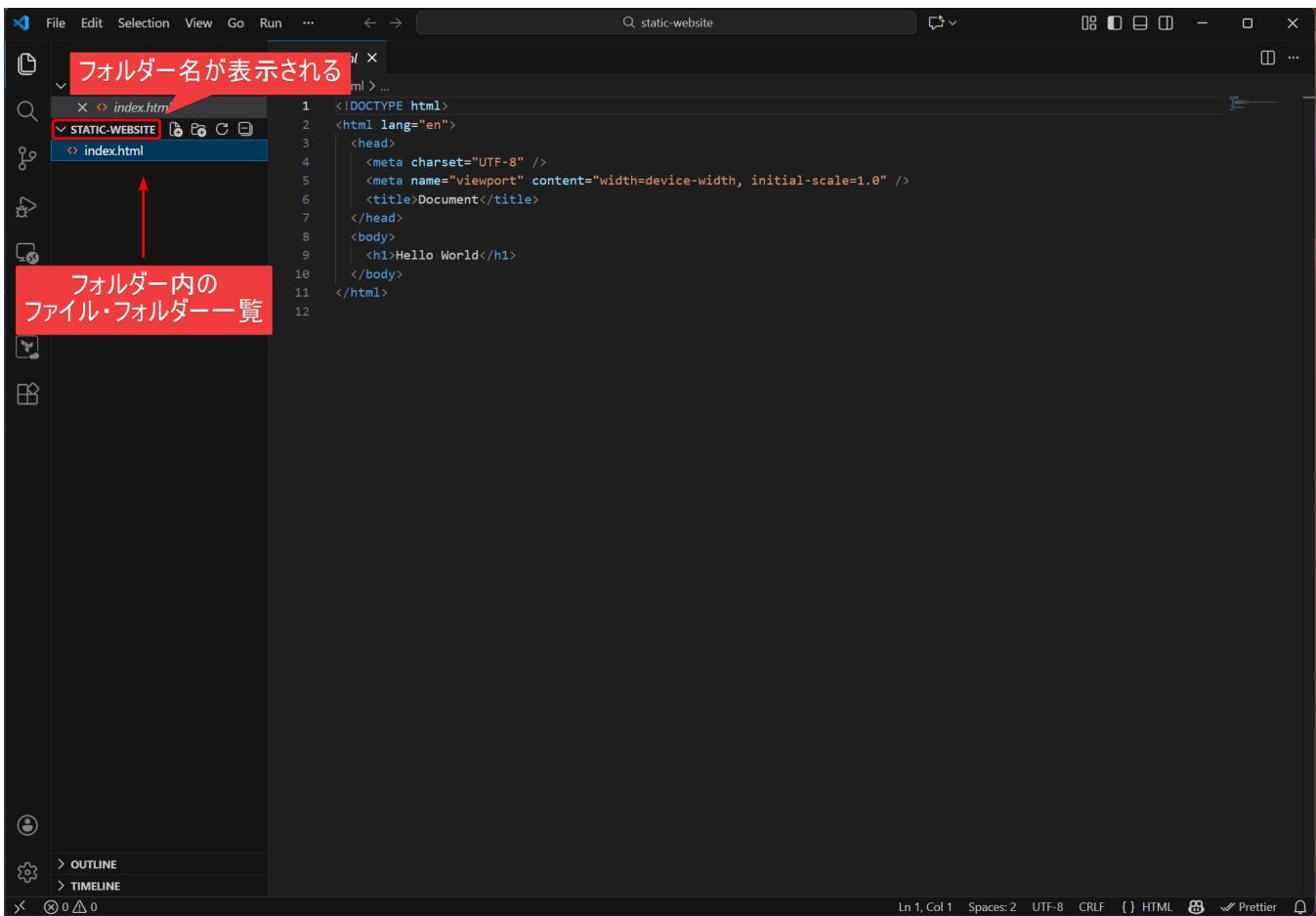


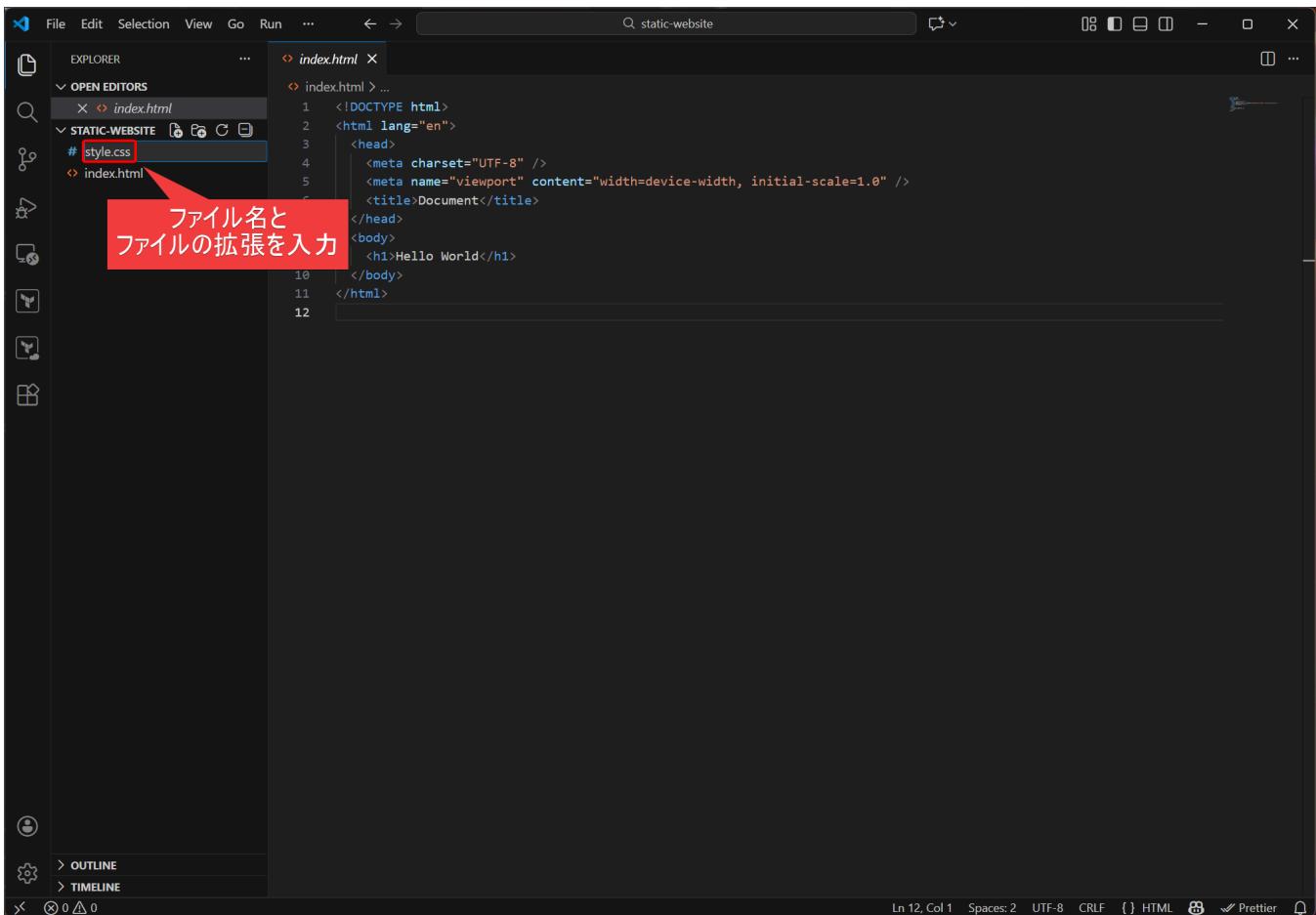
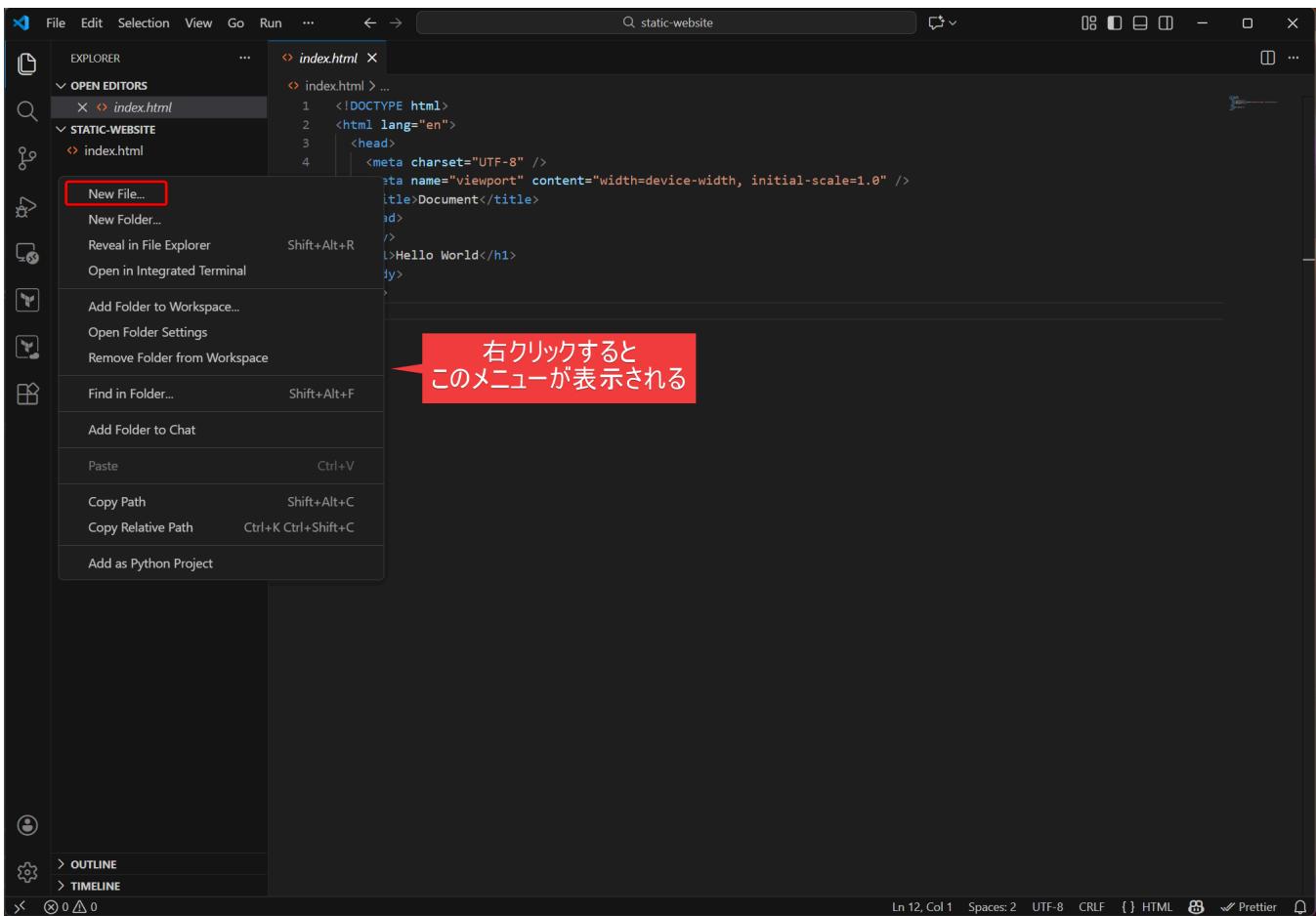
```
1 保存したファイル名で  
2 表示される  
3  
4  
5 <meta name="viewport" content="width=device-width, initial-scale=1.0" />  
6 <title>Document</title>  
7 </head>  
8 <body>  
9 | <h1>Hello World</h1>  
10 | </body>  
11 </html>  
12
```

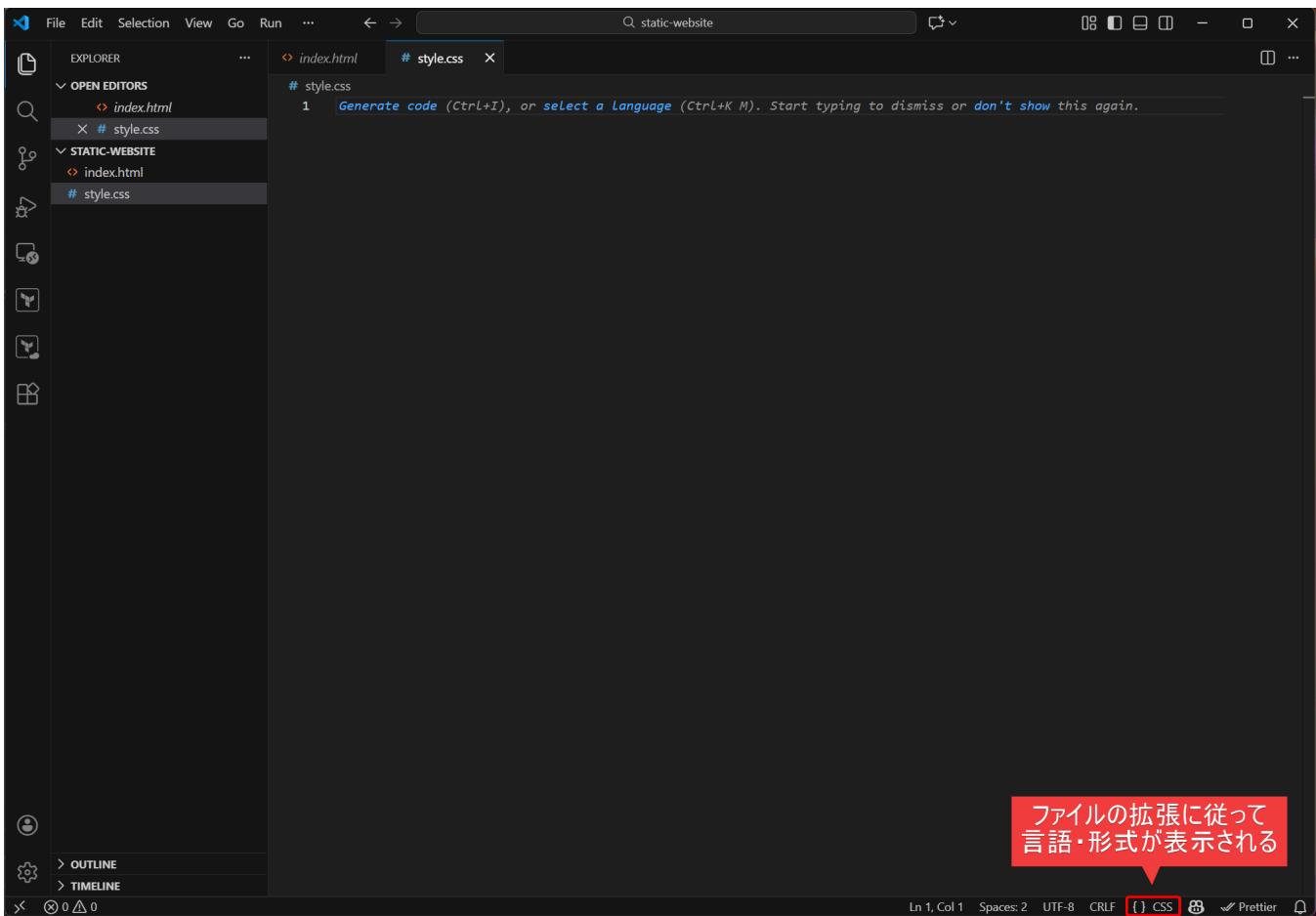
The code editor shows the saved file "index.html". The content of the file is displayed in the main pane, starting with the text "保存したファイル名で" followed by "表示される". The code itself is a simple HTML document with a single

element.

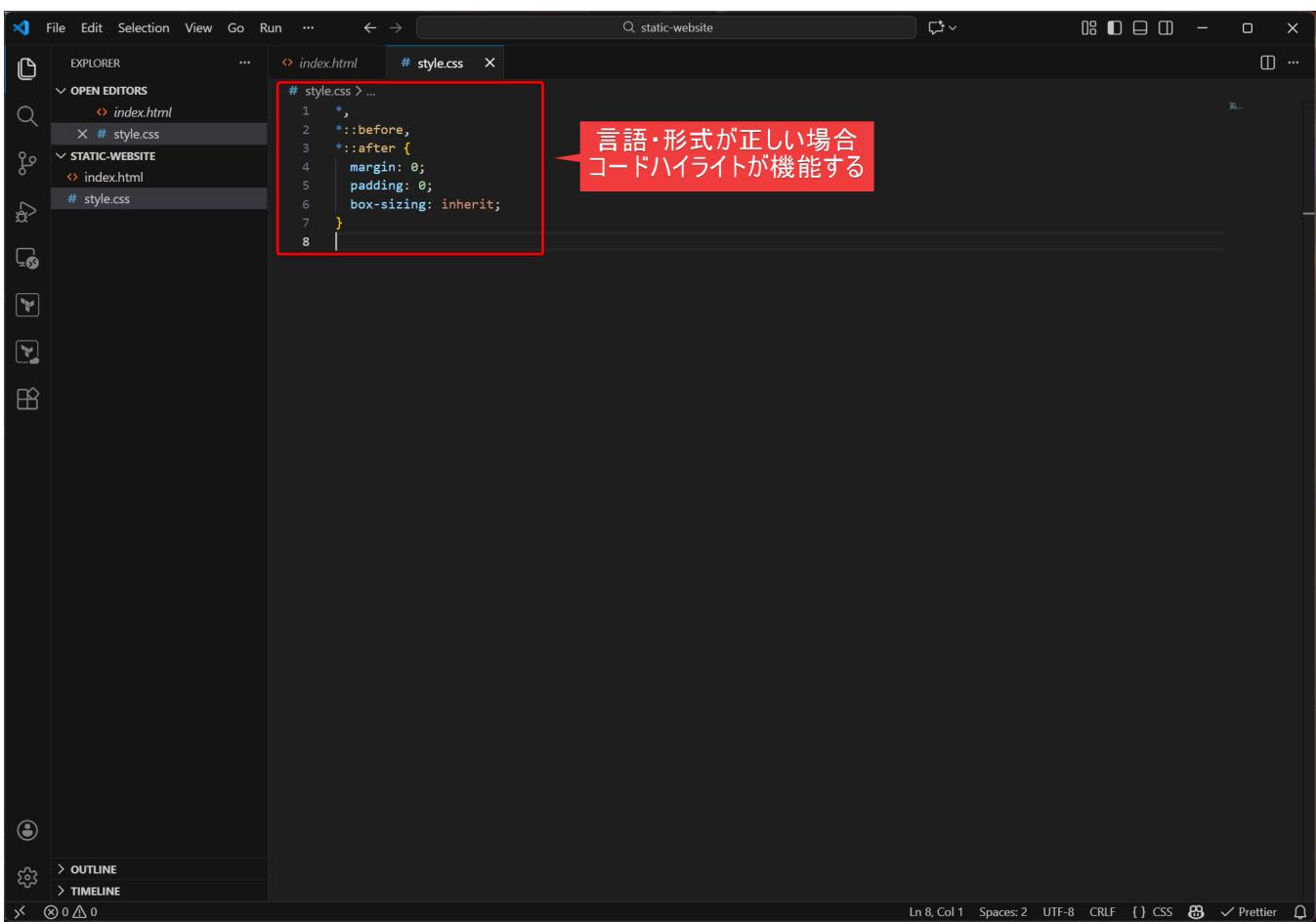




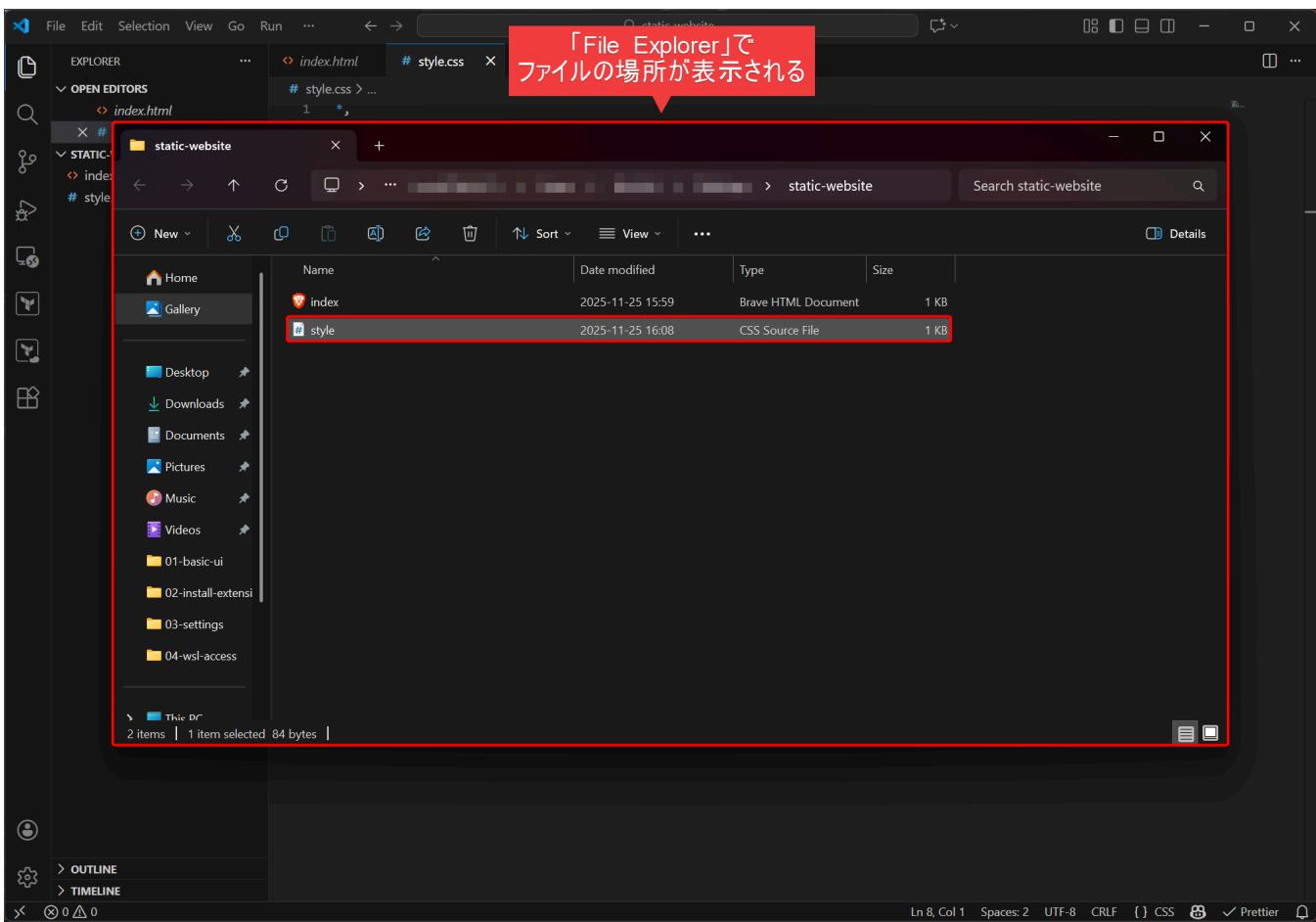
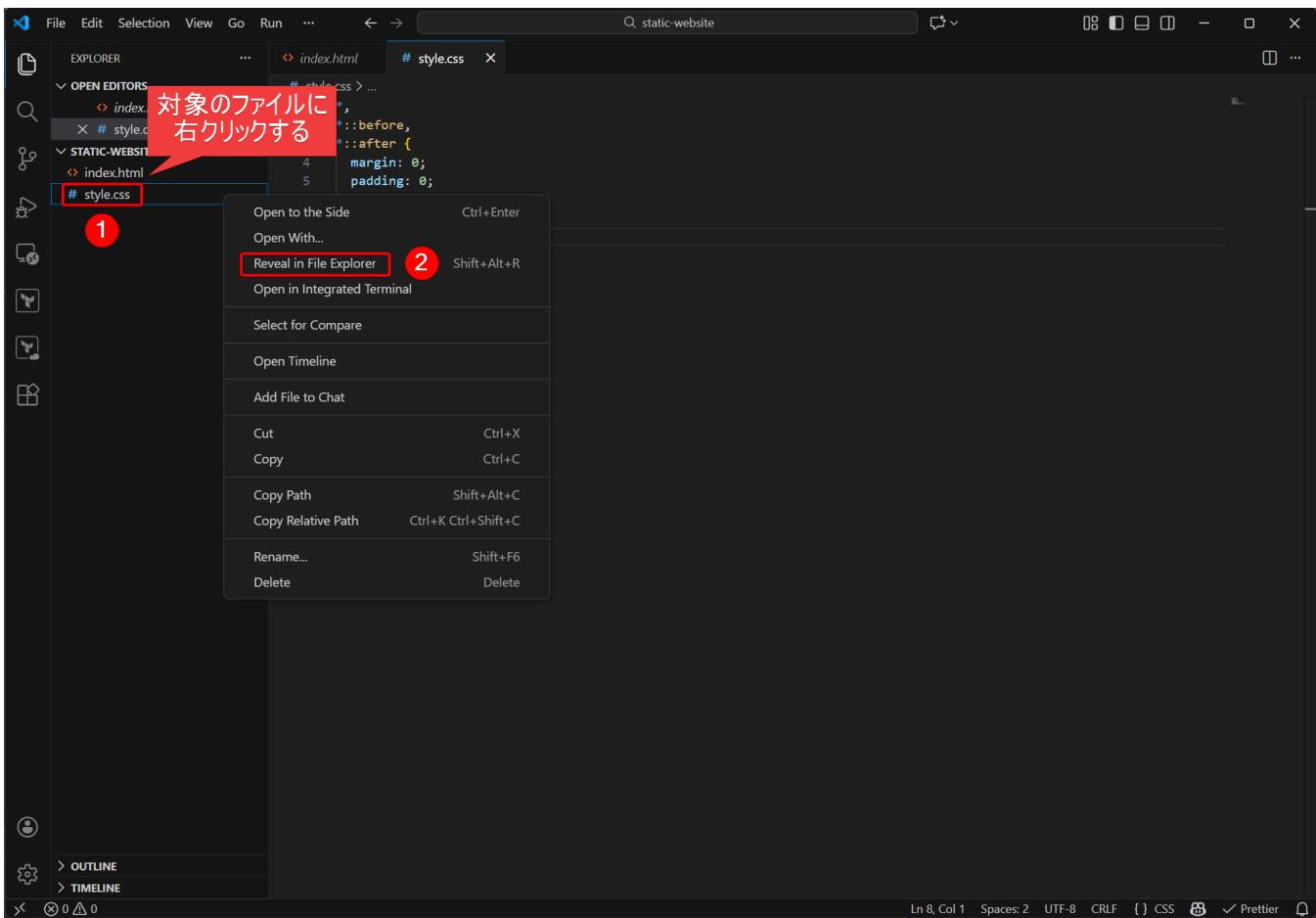


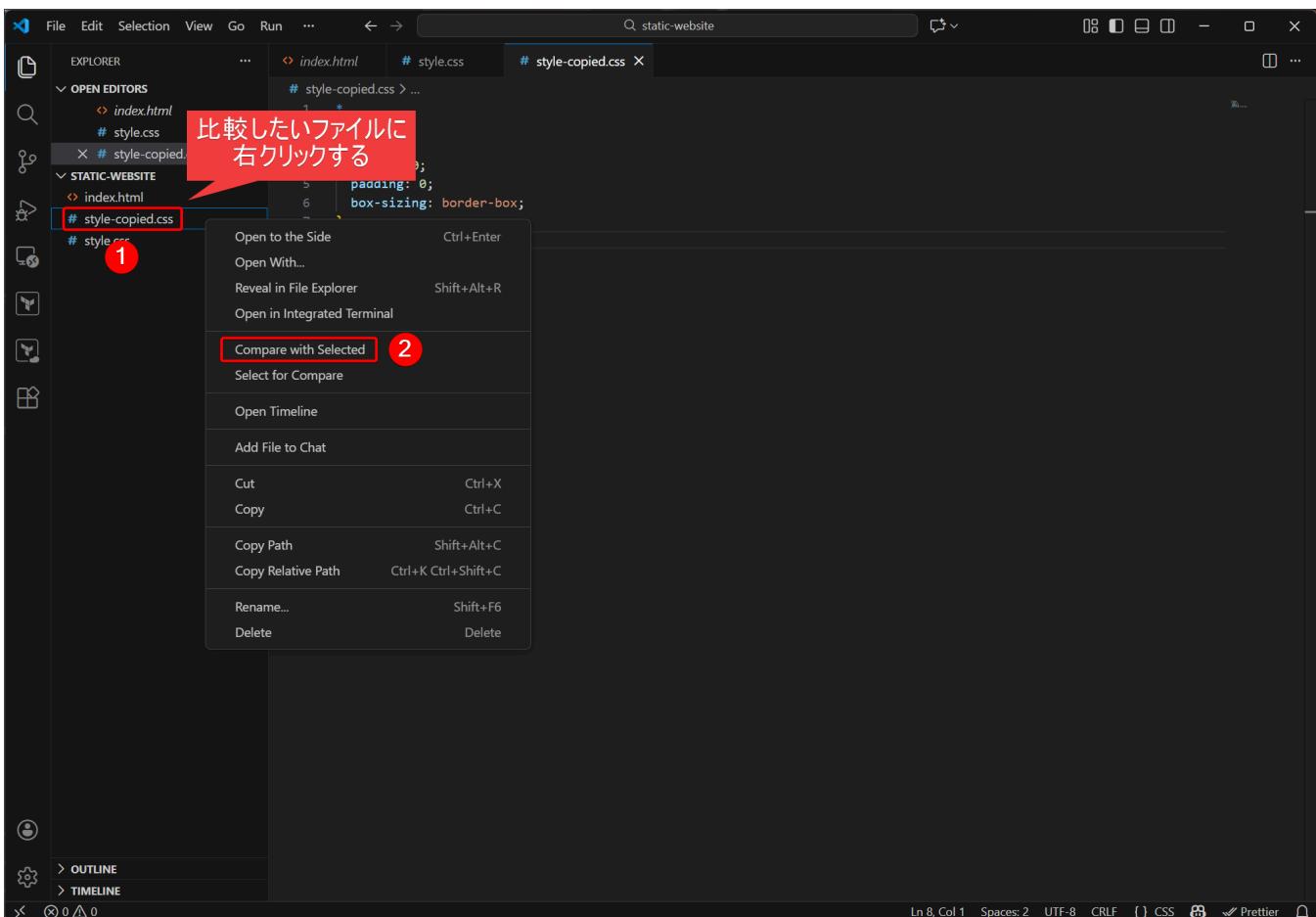
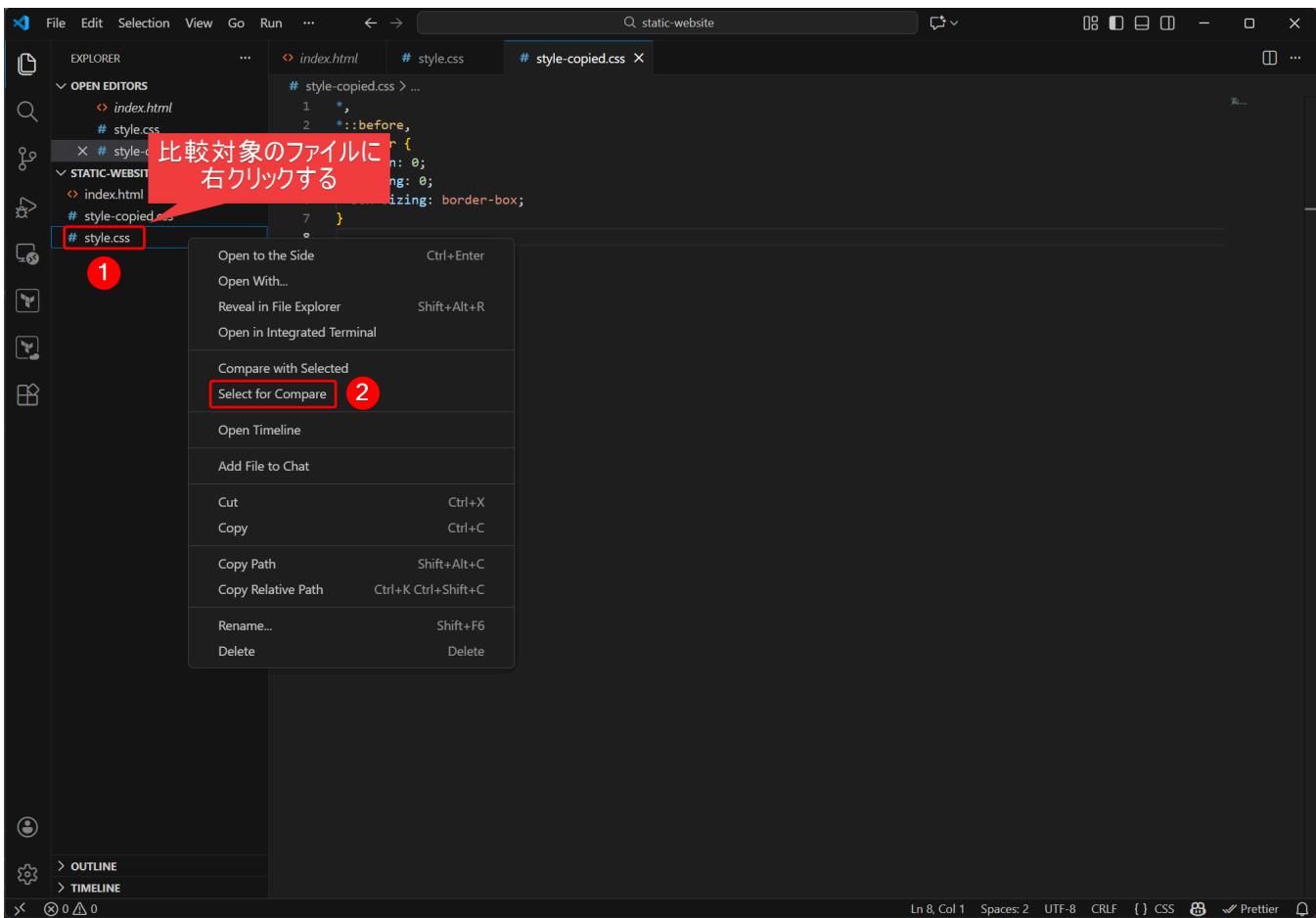


ファイルの拡張に従って
言語・形式が表示される



言語・形式が正しい場合
コードハイライトが機能する





A screenshot of the Visual Studio Code interface. The top editor shows the file `# style.css ↔ style-copied.css`. The bottom editor shows the file `# style-copied.css`. A red box highlights the bottom editor, and a red arrow points from it to a red callout box containing the Japanese text: "中身を比較する専用の画面が表示される" (A specialized comparison view is displayed). The status bar at the bottom right shows "Ln 8, Col 1".

A screenshot of the Visual Studio Code interface. The top editor shows the file `# style.css`. The bottom editor shows the file `# style-copied.css`. A red box highlights the bottom editor, and a red arrow points from it to a red callout box containing the Japanese text: "自動補完機能で補完可能となるコードが表示される" (Code that can be completed automatically by the auto-completion feature is displayed). The status bar at the bottom right shows "Ln 11, Col 8".

A screenshot of the Visual Studio Code interface. The top tab bar shows 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', and '...'. The search bar says 'static-website'. The left sidebar has icons for file explorer, search, and other tools. The main editor area shows an HTML file named 'index.html' with the following code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1>Hello World</h1>
    <div></div>
  </body>
</html>
```

The cursor is at the end of the opening 'div' tag. A red box highlights the 'div' tag, and a callout bubble says: 「Tab」ボタンを押すと
自動的に対象のコードが完成する (Pressing the 'Tab' button automatically completes the target code). The status bar at the bottom right shows 'Ln 11, Col 10' and other settings.

A screenshot of the Visual Studio Code interface. The top tab bar shows 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', and '...'. The search bar says 'static-website'. The left sidebar has icons for file explorer, search, and other tools. The main editor area shows a JavaScript file named 'index.js' with the following code:

```
1 confirm
```

The cursor is at the start of the word 'confirm'. A red box highlights the 'confirm' keyword, and a callout bubble says: 補完可能な一覧が
表示されることが多い (A list of completable items is often displayed). The status bar at the bottom right shows 'Ln 1, Col 3' and other settings.

A screenshot of the Visual Studio Code interface. The left sidebar shows an 'EXPLORER' view with files like 'index.html', 'index.js', and 'style.css'. The main editor area has an open file 'index.js' with the following code:

```
1 console
2
```

A code completion dropdown menu is open over the word 'console', listing various options such as 'confirm', 'var console: Console', 'const', 'continue', etc. A red arrow points from the text 'キーボードの矢印キーで対象の補完を選択できる' (Select the completion item with the keyboard arrow keys) to the dropdown menu.

A screenshot of the Visual Studio Code interface, similar to the one above, but with the code completion dropdown closed. The 'index.js' file now contains:

```
1 console
2
```

A red arrow points from the text '補完完了の場合' (When completion is finished) to the word 'console' in the code. The status bar at the bottom indicates 'Ln 1, Col 3' and other settings.

A screenshot of the Visual Studio Code interface. The left sidebar shows an 'EXPLORER' view with 'OPEN EDITORS' containing 'index.html', 'JS index.js' (which is the active editor), and '# style.css'. Below it is a 'STATIC-WEBSITE' section with 'index.html' and 'JS index.js'. The main editor area shows the following code:

```
1 console.assert
2
```

The cursor is at the start of the second line. A red box highlights the code completion dropdown for 'console.assert', which lists methods: assert, clear, count, countReset, debug, dir, dirxml, error, group, groupCollapsed, groupEnd, and info. A red arrow points from a callout box below the dropdown to the text: "プログラム言語の場合はこのようにメソッド一覧も表示される".

A screenshot of the Visual Studio Code interface, identical to the one above but with a different code snippet. The editor shows:

```
1 console.log
2
```

The cursor is at the start of the second line. A red box highlights the code completion dropdown for 'console.log', which lists methods: log, timeLog, clear, table, dirxml, and groupCollapsed. A red arrow points from a callout box below the dropdown to the text: "メソッド名に含まれる文字を入力すると更に補完可能なメソッドを絞れる".

This screenshot shows a clean development environment in VS Code. The Explorer sidebar lists files: index.html, index.js, style.css, and style-copied.css. The index.js editor tab is active, displaying the following code:

```
JS indexjs > ...
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 }
6
```

A red box highlights the code area with the label "サンプルコード エラーがない状態" (Sample code, no errors). Another red box highlights the status bar at the bottom with the label "エラーがない状態" (No errors).

This screenshot shows the same setup as the previous one, but with an error present in the code. The index.js editor tab is active, displaying the following code:

```
JS indexjs > ...
1 console.log('Hello World');
2
3 const sayHello = (name => {
4   console.log(name);
5 });
6
```

A red box highlights the status bar at the bottom with the label "エラーがある状態" (Error state). A red box highlights the status bar again with the label "エラーの数が表示される" (Number of errors displayed). A red box highlights the code area with the label "赤い線が表示される" (Red lines displayed).

A screenshot of the Visual Studio Code interface. The left sidebar shows the Explorer, Outline, and Timeline. The main area has four tabs: index.html, JS index.js (with a red box around the '1' icon), # style.css, and # style-copied.css. The JS index.js tab is active, displaying the following code:

```
1 console.log('Hello World');
2
3 const sayHello = (name => {
4   console.log(name);
5 })
```

A red arrow points from a text box containing the Japanese text "エラーにマウスでホバーするとエラーの内容が表示される" (When you hover over an error, its details are displayed) to the error message "c ')' expected. ts(1005)" located in the status bar at the bottom of the editor.

A screenshot of the Visual Studio Code interface. The left sidebar shows the Explorer, Outline, and Timeline. The main area has four tabs: index.html, JS index.js (with a red box around the '1' icon), # style.css, and # style-copied.css. The JS index.js tab is active, displaying the following code:

```
1 console.log('Hello World');
2
3 const sayHello = (name => {
4   console.log(name);
5 })
```

A red arrow labeled '1' points from the Timeline section of the sidebar to the Problems panel. The Problems panel is open, showing one error: "JS index.js ① ⚡ ')' expected. ts(1005) [Ln 5, Col 2]". A red box highlights the 'PROBLEMS 1' tab. A red arrow labeled '2' points from the error message in the Problems panel to a text box containing the Japanese text "すべてのエラー・警告が表示される" (All errors and warnings are displayed).

The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows files in the workspace: index.html, index.js, style.css, style-copied.css, index.html, index.js, style-copied.css, and style.css.
- Editor:** The index.js file is open, containing the following code:

```
1 console.log('Hello World');
2
3 const sayHello = [name] => {
4   console.log(name);
5 };
6
```

A red box highlights the closing brace of the sayHello function. A red arrow points from this box to a red box labeled "エラーが修正される状態" (Error is being resolved).
- Problems View:** Located at the bottom left, it shows the message "No problems have been detected in the workspace." A red box highlights this message.
- Terminal:** Located at the bottom right, showing the status bar with "Ln 3, Col 24" and other settings.

The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows files in the workspace: index.html, index.js, style.css, style-copied.css, index.html, index.js, style-copied.css, and style.css.
- Editor:** The index.js file is open, containing the following code:

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8   console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12   console.log(name);
13 };
14
```

A red box highlights the opening parenthesis of the sayKonichiwa function. A red arrow points from this box to a red box labeled "サンプルコード 同じ文字が複数存在するパターン" (Sample code where the same character exists multiple times).
- Terminal:** Located at the bottom right, showing the status bar with "Ln 14, Col 1" and other settings.

This screenshot shows the VS Code interface with the search bar at the top containing the text 'hello'. Below the search bar, the results are displayed in a red box. The first result is '検索でマッチしたテキスト' (Text matched by search), which points to the line 'console.log('Hello World');'. The second result is 'マッチした数' (Number of matches), which points to the status bar indicating '1 of 2'.

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4     console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8     console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12     console.log(name);
13 };
14
```

This screenshot shows the VS Code interface with the search bar at the top containing the text 'Hello'. Below the search bar, the results are displayed in a red box. The first result is 'マッチなしの場合' (Case where no match occurs), which points to the status bar indicating 'No results'. The second result is '大文字・小文字を別の者として区別する' (Distinguish between uppercase and lowercase letters), which points to the line 'console.log('Hello World');'.

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4     console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8     console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12     console.log(name);
13 };
14
```

This screenshot shows the VS Code interface with the search bar set to 'hello'. The results pane displays two matches from the 'index.js' file:

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4     console.log(name);
5 };
6
7 const sayBonjour = (name) => {
8     console.log(name);
9 };
10
11 const sayKonichiwa = (name) => {
12     console.log(name);
13 };
```

Annotations explain the search results:

- A red box highlights the word 'Hello' in the first result, with the text 'マッチした単語' (Matched word).
- A red box highlights the word 'sayHello' in the second result, with the text '「sayHello」という単語で認識されるため「マッチしない」になる' (Because it's recognized as the word 'sayHello', it's not matched).
- A red box highlights the search input 'hello' in the status bar, with the text '単語として区別する' (Distinguish as words).

This screenshot shows the VS Code interface with the search bar set to 'hello'. The results pane displays two matches from the 'index.js' file, with selection arrows indicating the order of matches:

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4     console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8     console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12     console.log(name);
13 };
```

Annotations explain the search results and selection:

- A red box highlights the word 'Hello' in the first result, with the text '1番目' (First).
- A red box highlights the word 'sayHello' in the second result, with the text '2番目' (Second).
- A red box highlights the up and down arrows in the status bar, with the text '「昇順」になる' (Becomes ascending order).
- A red box highlights the search input 'hello' in the status bar, with the text 'マッチした内容を順番で選択できる' (Can select matched content in order).

順番が変わると
対象のマッチした内容がハイライトされる

```
1  console.log('Hello World');
2
3  const sayHello = (name) => {
4      console.log(name);
5  };
6
7
8
9
10 const sayBonjour = (name) => {
11     console.log(name);
12 };
13
14
```

The screenshot shows the Visual Studio Code interface with a dark theme. The left sidebar displays the file tree, showing three open files: index.html, index.js, and style.css. The main editor area is focused on the index.js file, which contains the following code:

```
1  console.log('Hello World');
2
3  const sayHello = (name) => {
4  |   console.log(name);
5  };
6
7  const sayKonichiwa = (name) => {
8  |   console.log(name);
9  };
10
11 const sayBonjour = (name) => {
12 |   console.log(name);
13 };
14
```

A red arrow points from the Japanese text "置き換える機能" (Replace feature) to the "Replace" button in the search and replace dialog box. The dialog box is titled "hello" and has the "Replace" button highlighted.

This screenshot shows the VS Code interface with the following steps:

- 1**: A red circle highlights the word "name" in the code editor, which is also selected in the status bar.
- 2**: A red box highlights the search bar at the top of the interface, with the text "「Control + F」を押す" (Press Control + F) overlaid.
- A red box highlights the status bar at the bottom, which displays "Ln 3, Col 23 (4 selected)" and "Spaces: 2".

The code in the editor is:

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8   console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12   console.log(name);
13 };
14
```

This screenshot shows the VS Code interface after performing a search for "name".

A red box highlights the search input field in the top bar, which contains the text "name".

A red box highlights the status bar at the bottom, which displays "Ln 3, Col 23 (4 selected)" and "Spaces: 2".

A red box highlights the message "自動的に選択したキーワードが検索キーワードの入力欄に反映される" (The selected keyword is reflected in the search input field) overlaid on the interface.

The code in the editor is:

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8   console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12   console.log(name);
13 };
14
```

「Control + R」を押すと置き換えるの機能が表示される

This screenshot shows the Visual Studio Code interface with a dark theme. The top bar includes 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', and other standard options. The title bar says 'static-website'. The left sidebar has sections for 'EXPLORER', 'OPEN EDITORS' (with 'index.html', 'index.js', '# style.css', and '# style-copied.css' listed), 'STATIC-WEBSITE' (with 'index.html', 'index.js', '# style-copied.css', and '# style.css'), and various icons for file operations. The main editor area contains a JavaScript file with code like:

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 }
6
7 const sayKonichiwa = (name) => {
8   console.log(name);
9 }
10
11 const sayBonjour = (name) => {
12   console.log(name);
13 };
14
```

The bottom status bar shows 'Ln 6, Col 1 Spaces: 2 UTF-8 CRLF {} JavaScript ✓ Prettier'. The bottom right corner of the interface has a small icon.

置き換える対象

This screenshot shows the same Visual Studio Code interface as the previous one, but with the search function open. A red box highlights the search input field which contains the word 'name'. The status bar at the bottom indicates 'Ln 3, Col 23 (4 selected) Spaces: 2 UTF-8 CRLF {} JavaScript ✓ Prettier'. The rest of the interface is identical to the first screenshot.

置き換える「値」

頭文字が大文字に変換
例:「person」→「Person」

This screenshot shows the VS Code interface with the 'index.js' file open. A red box highlights the 'name' placeholder in the 'Replace' field of the Refactor tool's replace dialog. A callout points to the placeholder with the text '置き換える「値」' (Value to replace) and '頭文字が大文字に変換' (First letter converted to uppercase), with an example '例:「person」→「Person」'.

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8   console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12   console.log(name);
13 };
14
```

置き換える対象の番
(ポジション)

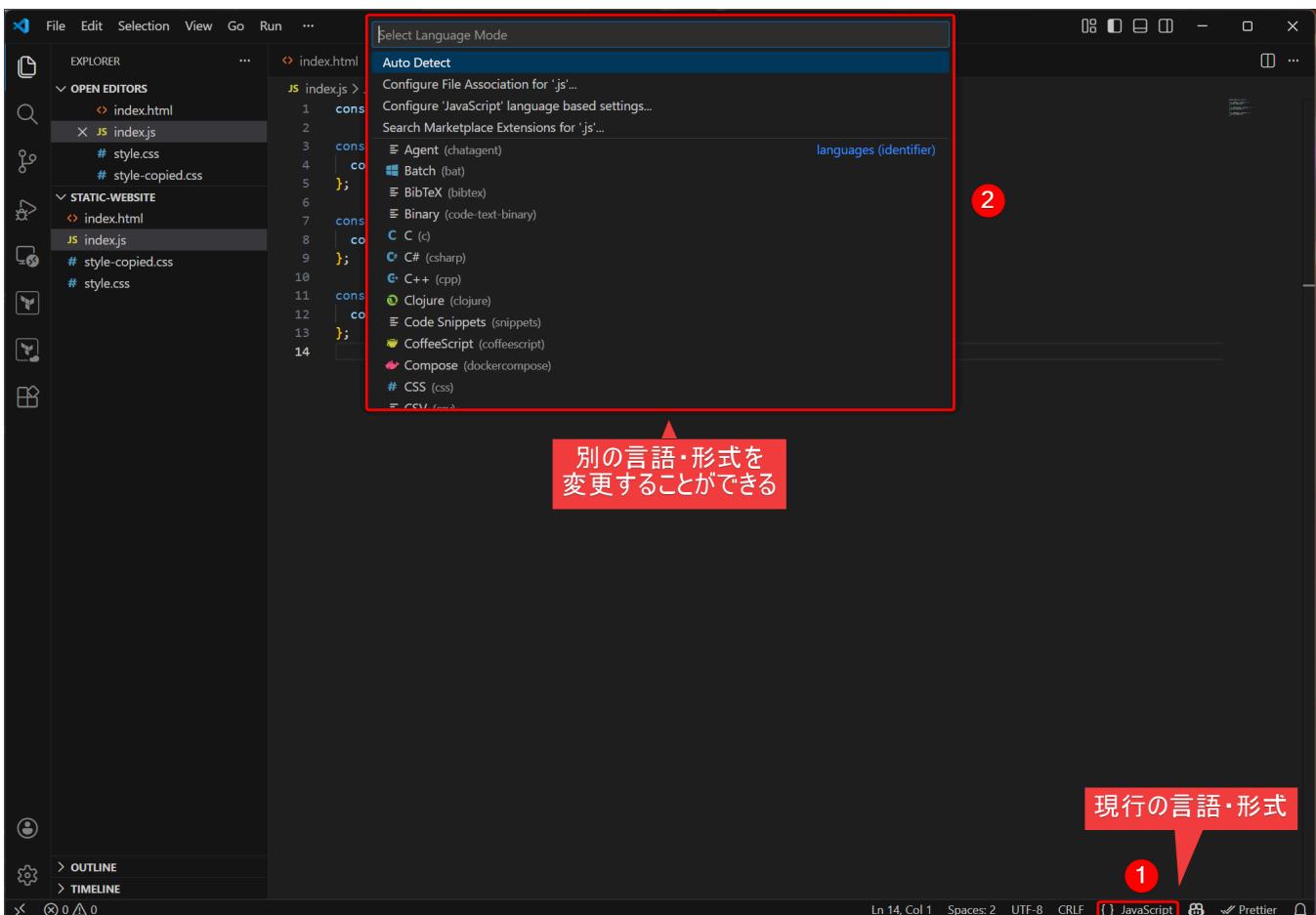
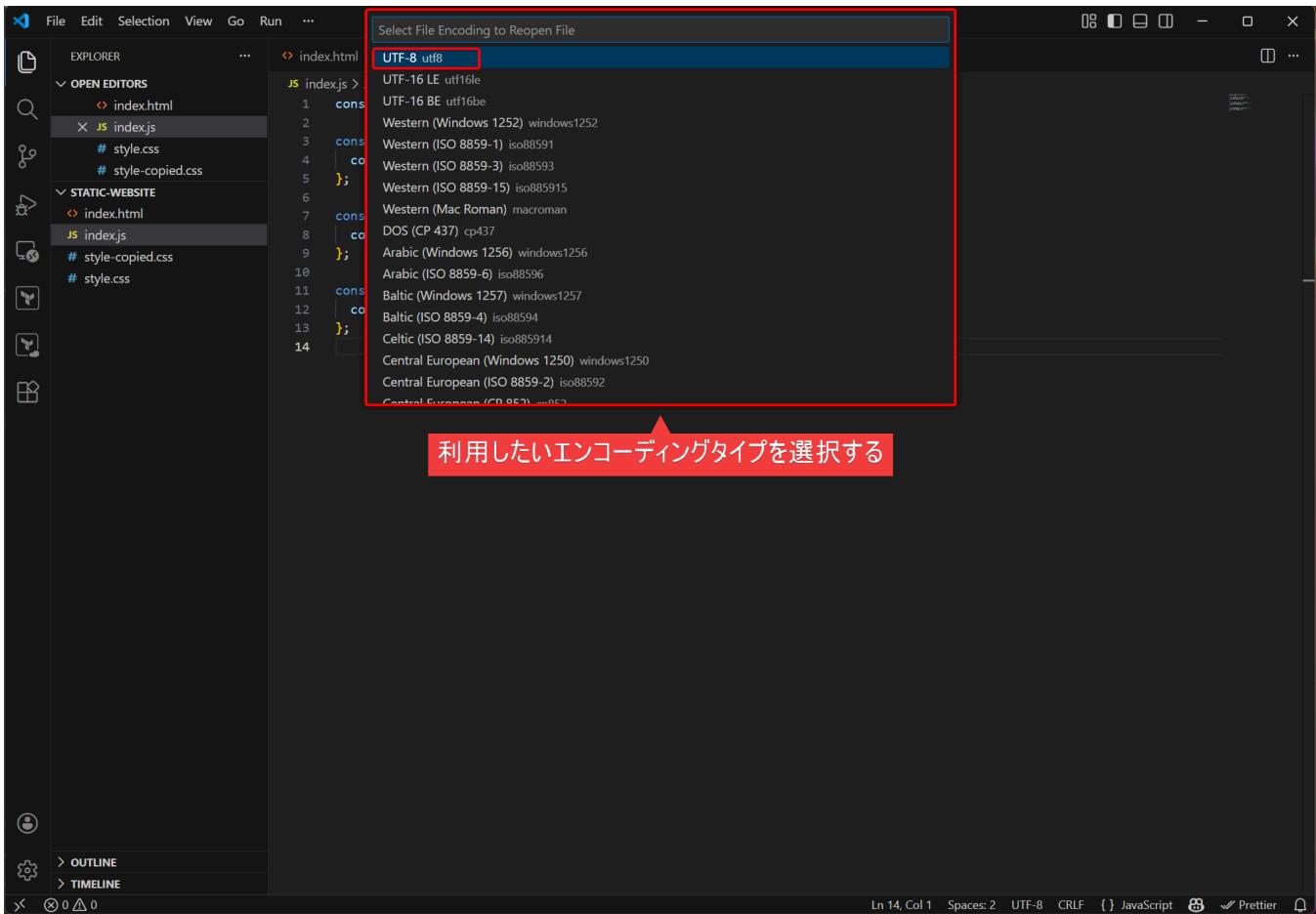
個別を置き換える

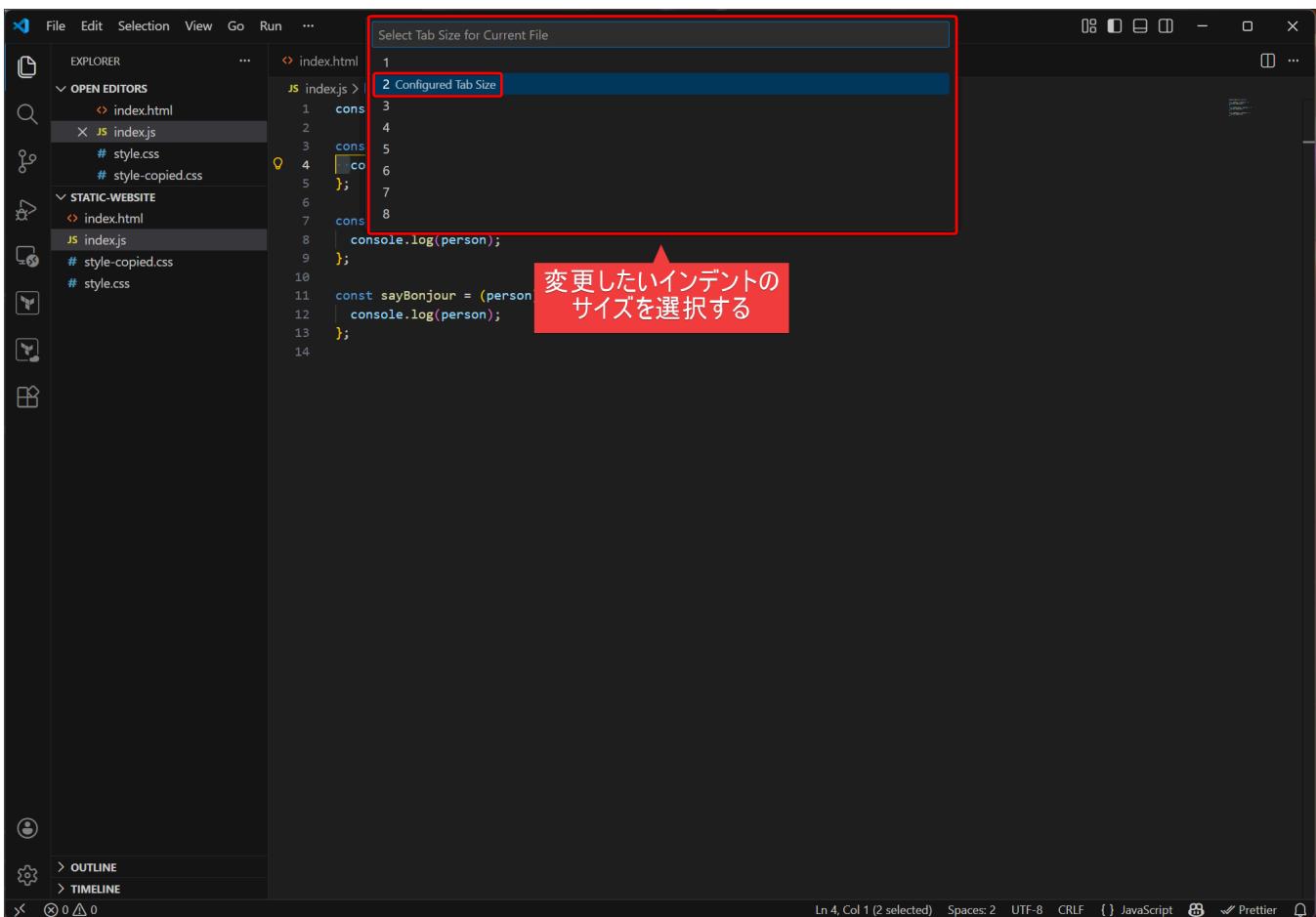
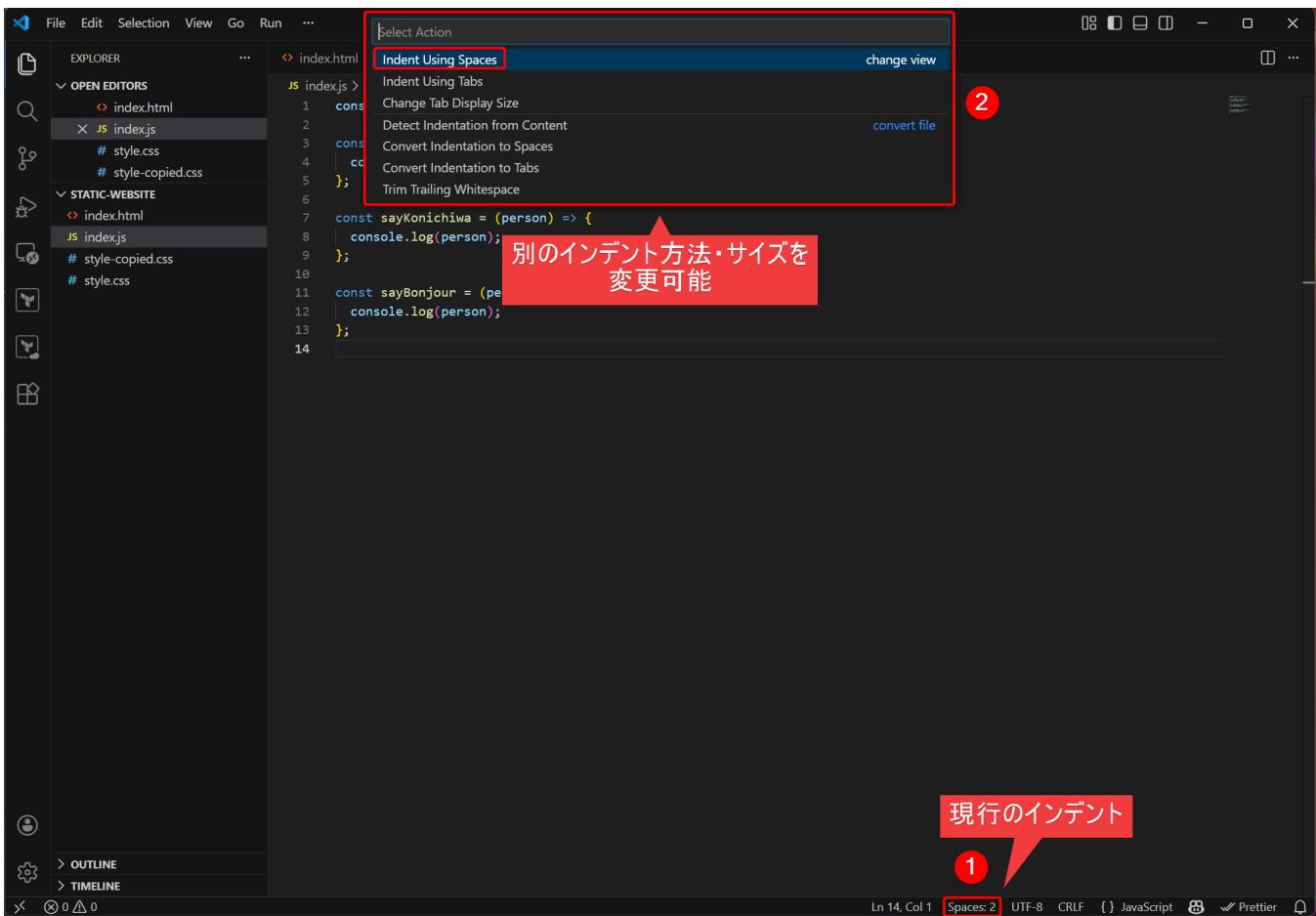
This screenshot shows the VS Code interface with the 'index.js' file open. A red box highlights the 'name' placeholder in the 'Replace' field of the Refactor tool's replace dialog. A red arrow points to the '1 of 5' count in the bottom right corner of the dialog. A callout points to the placeholder with the text '置き換える対象の番' (Position of target) and '(ポジション)' (Position), and another callout points to the count with the text '個別を置き換える' (Replace individually).

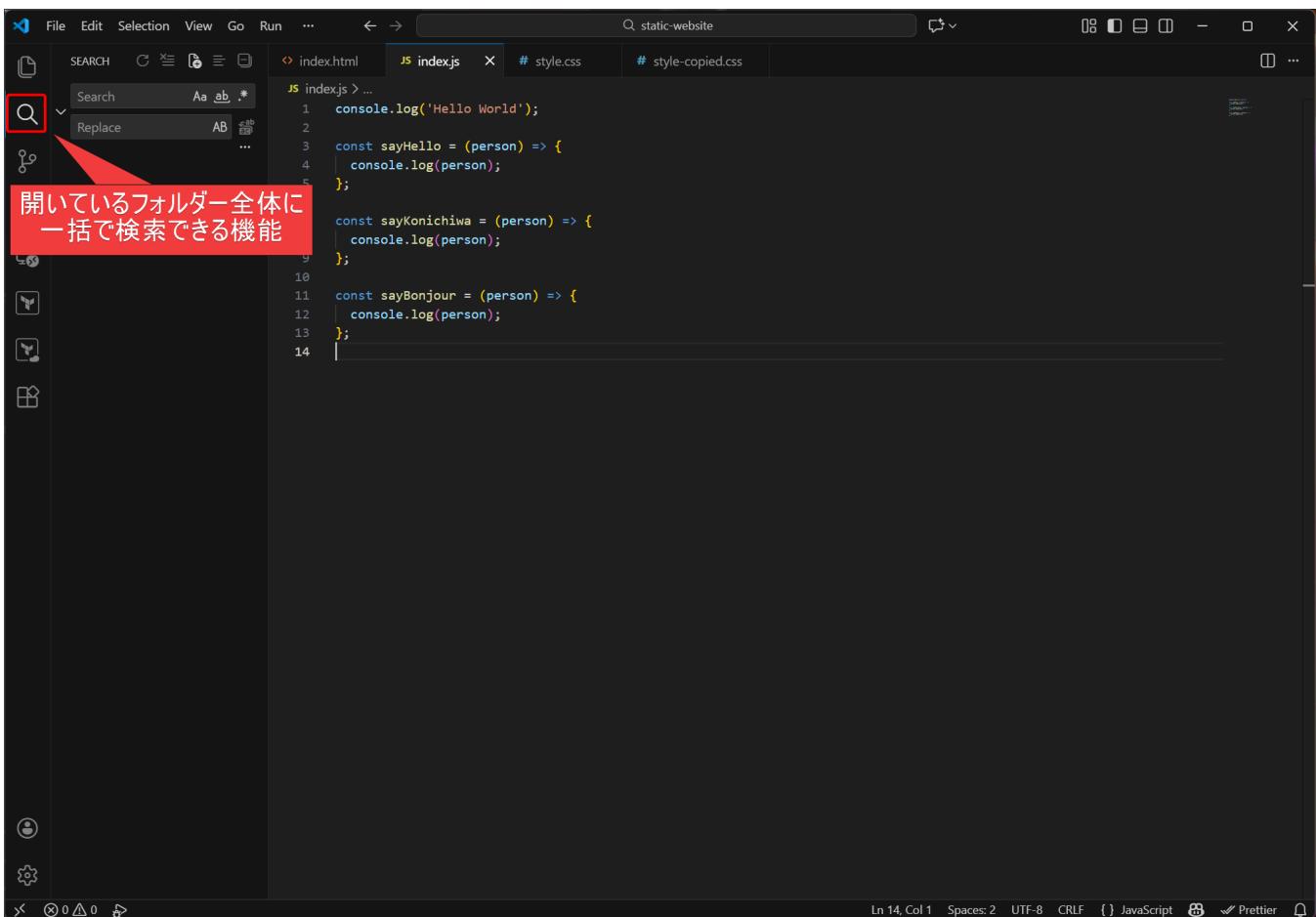
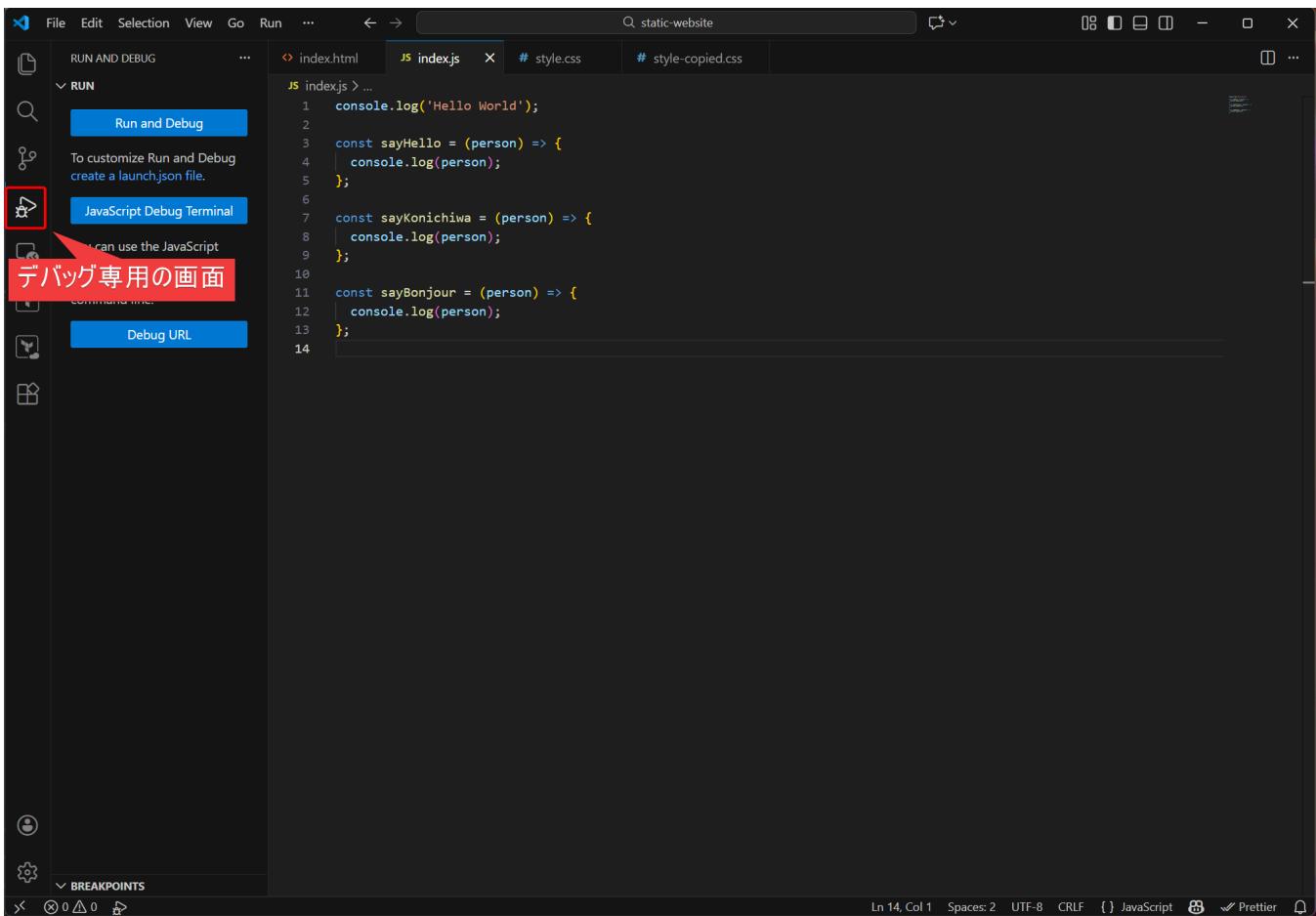
```
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (name) => {
8   console.log(name);
9 };
10
11 const sayBonjour = (name) => {
12   console.log(name);
13 };
14
```

A screenshot of the Visual Studio Code interface. The left sidebar shows the Explorer, Outline, and Timeline. The center is the editor with the file 'index.js' open. The status bar at the bottom indicates 'Ln 4, Col 19 (4 selected) Spaces: 2 UTF-8 CRLF {} JavaScript ✓ Prettier'. A search bar at the top right contains 'sayHello'. Below it is a search results panel with 'name' and 'person' listed. A red box highlights the 'Replace' button in the search results panel. A red callout bubble points to this button with the text 'マッチした内容のすべてを一括で置き換える' (Replace all occurrences of the selected content).

A screenshot of the Visual Studio Code interface. The left sidebar shows the Explorer, Outline, and Timeline. The center is the editor with the file 'index.js' open. The status bar at the bottom indicates 'Ln 14, Col 1 Spaces: 2 UTF-8 CRLF {} JavaScript ✓ Prettier'. A context menu is open over the code, with 'Reopen with Encoding' highlighted. A red box highlights 'Reopen with Encoding'. A red callout bubble points to the status bar with the text '現状のエンコーディングタイプ' (Current encoding type). A red number '1' is placed over the 'UTF-8' text in the status bar, and a red number '2' is placed over the 'Reopen with Encoding' option in the context menu.







検索キーワード

SEARCH person Aa ab *

Replace

7 results in 2 files

editor

index.html JS index.js # style.css # style-copied.css

index.html JS index.js

マッチしたファイル

マッチした数

そのファイル内にマッチした数

マッチした数

JS index.js

1 JS index.js

6 JS index.js

```
'Hello World');
lo = (person) => {
  ...
}
const sayKonichiwa = (person) => {
  console.log(person);
}
const sayBonjour = (person) => {
  console.log(person);
}
const sayHello = (person) => {
  console.log(person);
}
const sayBonjour = (person) ...
const sayHello = (person) ...
console.log(person);
```

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript Prettier

マッチした箇所を選択すると対象の箇所がハイライトされる

SEARCH person Aa ab *

index.html JS index.js # style.css # style-copied.css

index.html JS index.js

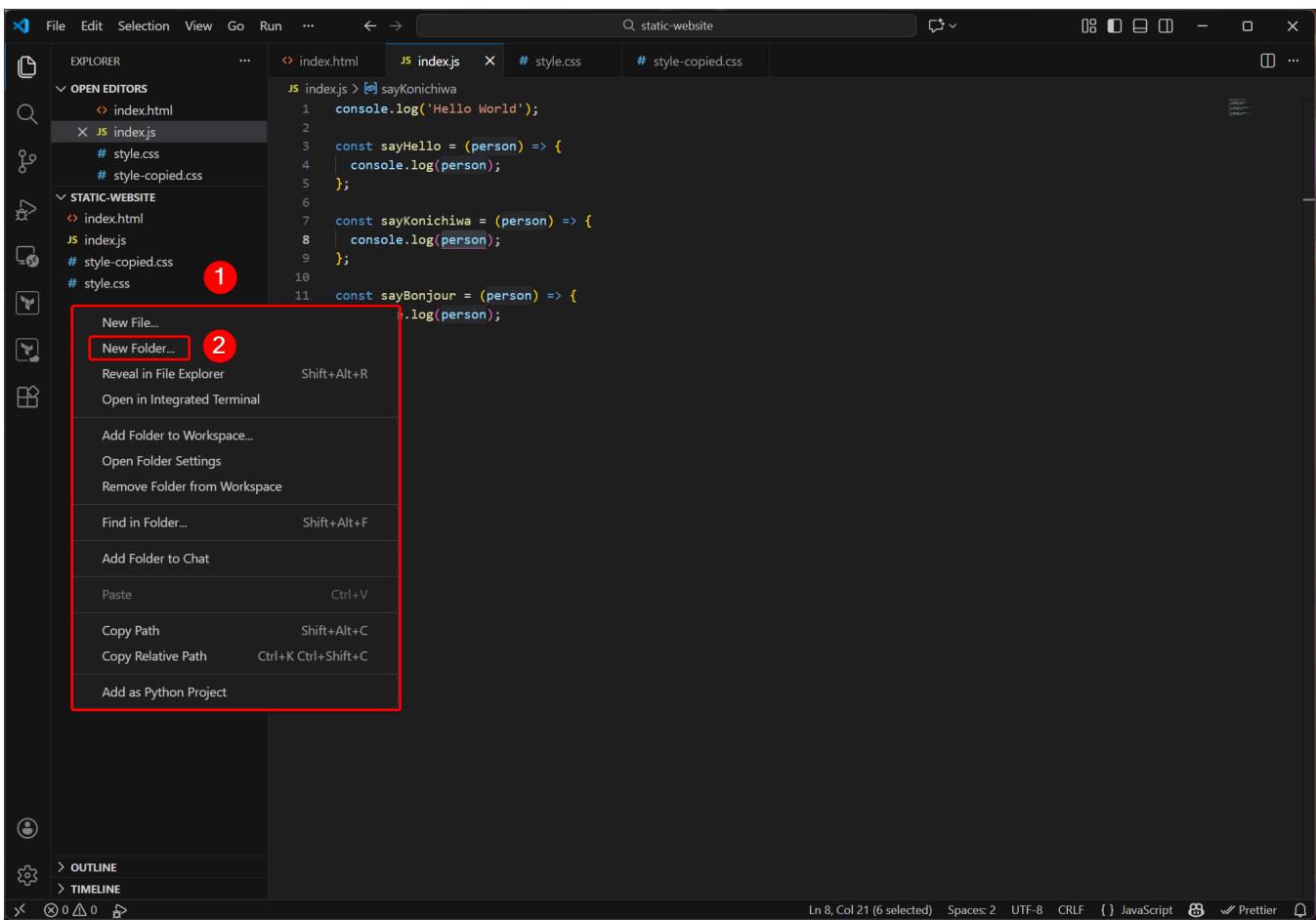
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1>Hello World</h1>
    <div>Person</div>
  </body>
</html>
```

Ln 11, Col 16 (6 selected) Spaces: 2 UTF-8 CRLF { } HTML Prettier

A screenshot of the Visual Studio Code interface. The left sidebar shows a search results panel with 7 results in 2 files, one of which is 'index.js'. The main editor area displays the contents of 'index.js':

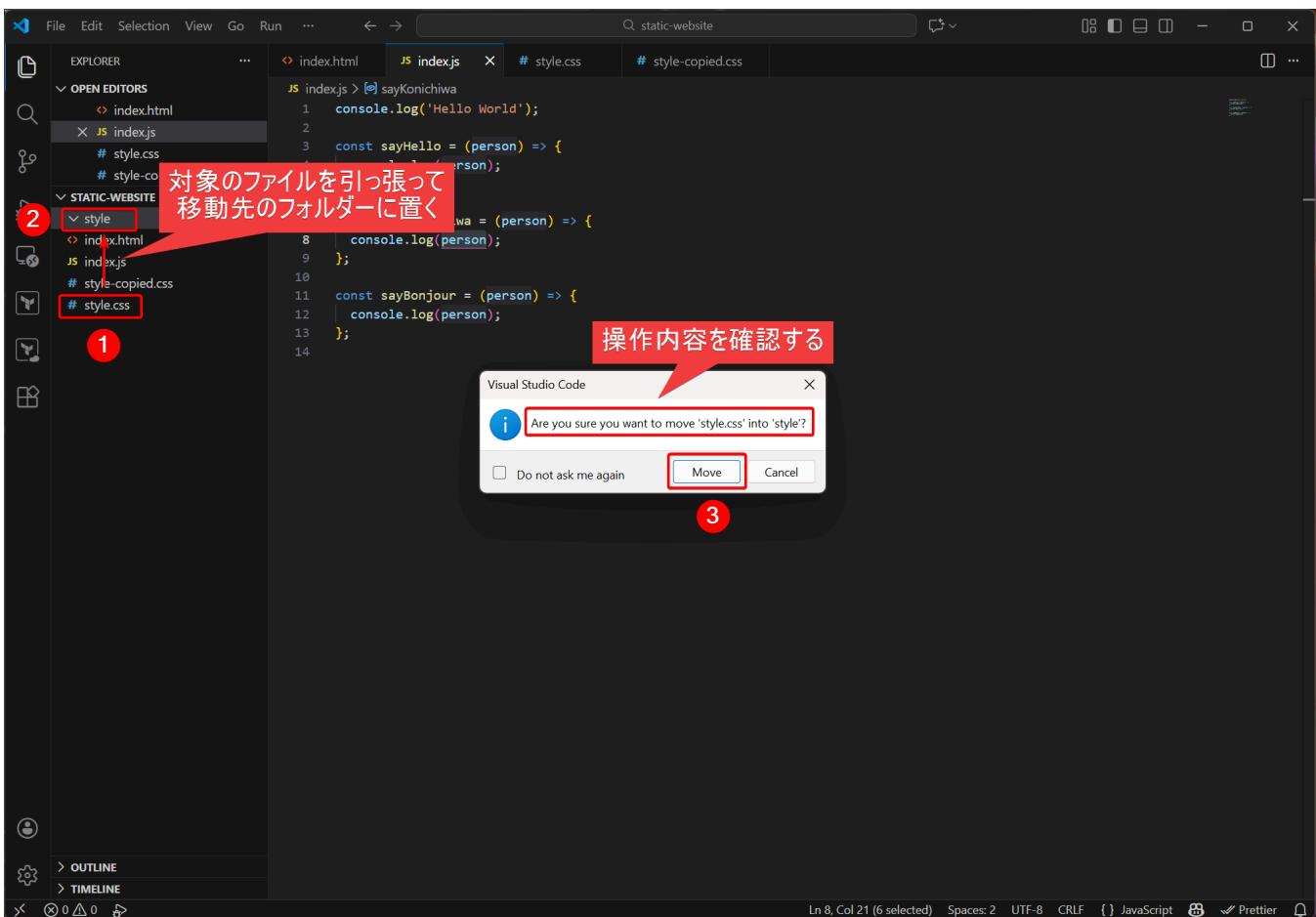
```
js index.js > [e] sayKonichiwa
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4 | console.log(person);
5 };
6
7 const sayKonichiwa = (person) => {
8 | console.log(person);
9 };
10 const sayBonjour = (person) => {
11 | console.log(person);
12 };
13
14
```

The word 'person' is highlighted in red across several lines of code. A red arrow points from the search results panel to the highlighted word in the code.



新しいフォルダ名をつける

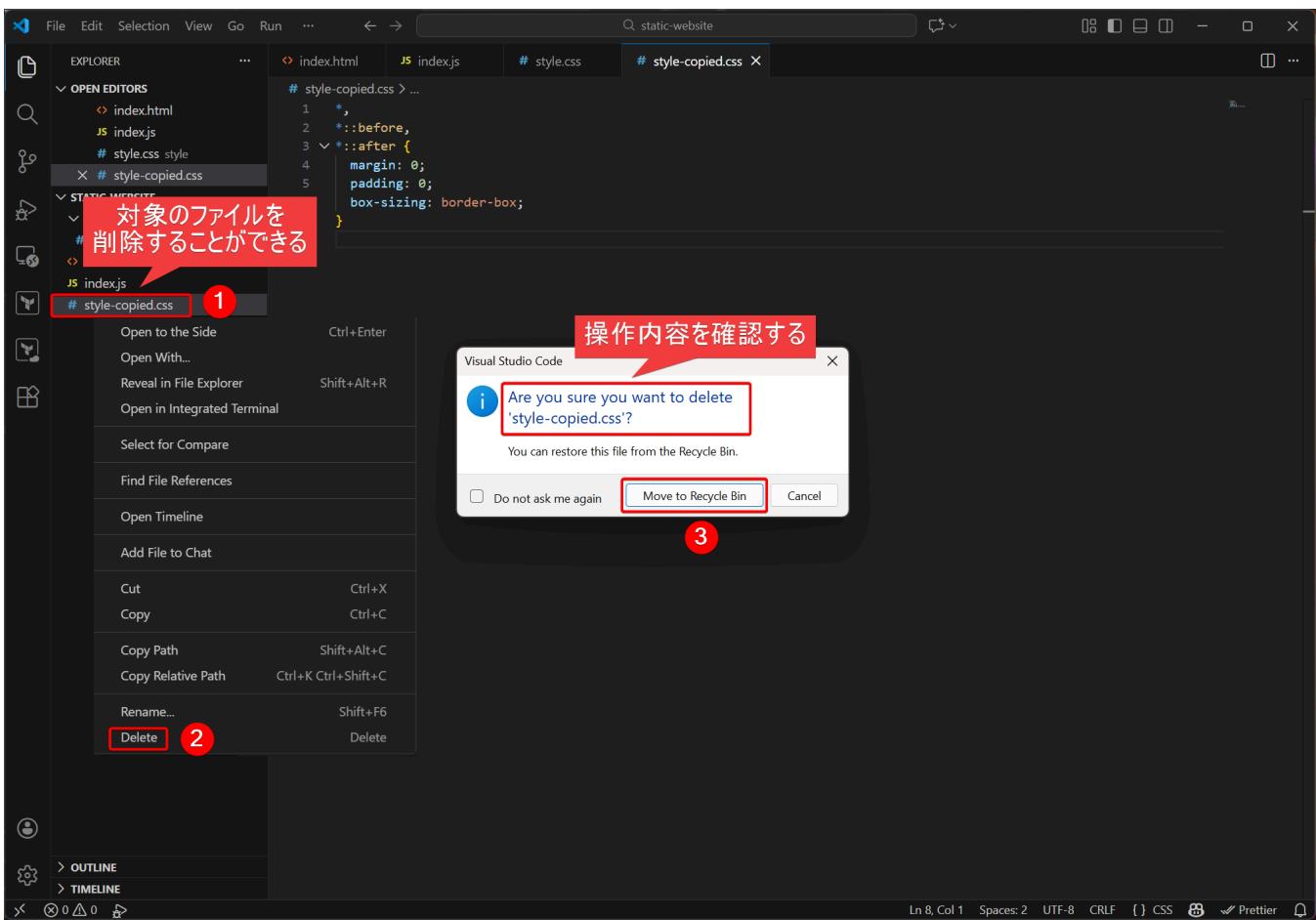
The screenshot shows the Visual Studio Code interface. In the Explorer sidebar, there is a folder named 'style' which is highlighted with a red box. The status bar at the bottom right indicates 'Ln 8, Col 21 (6 selected) Spaces: 2 UTF-8 CRLF {} JavaScript Prettier'.



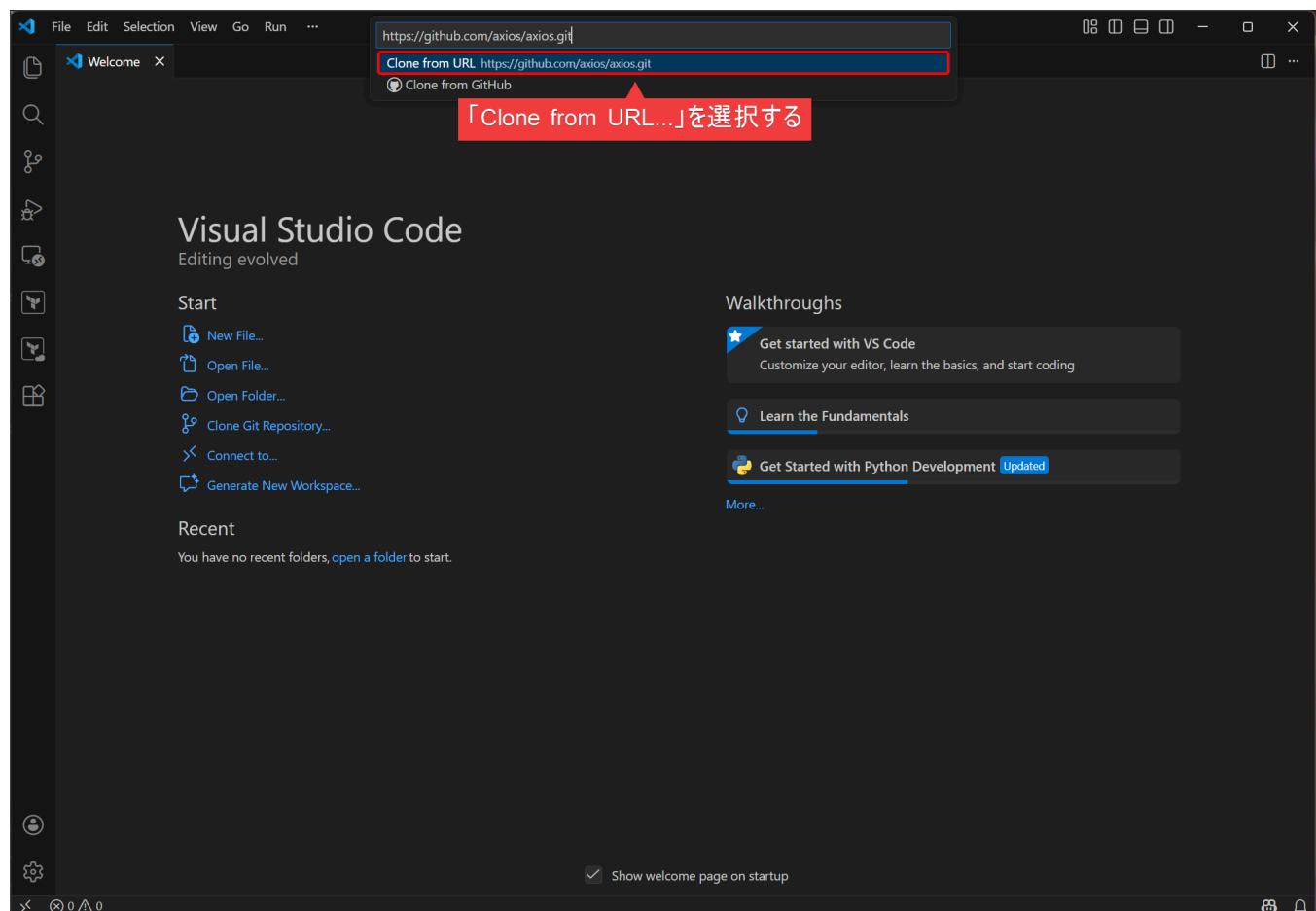
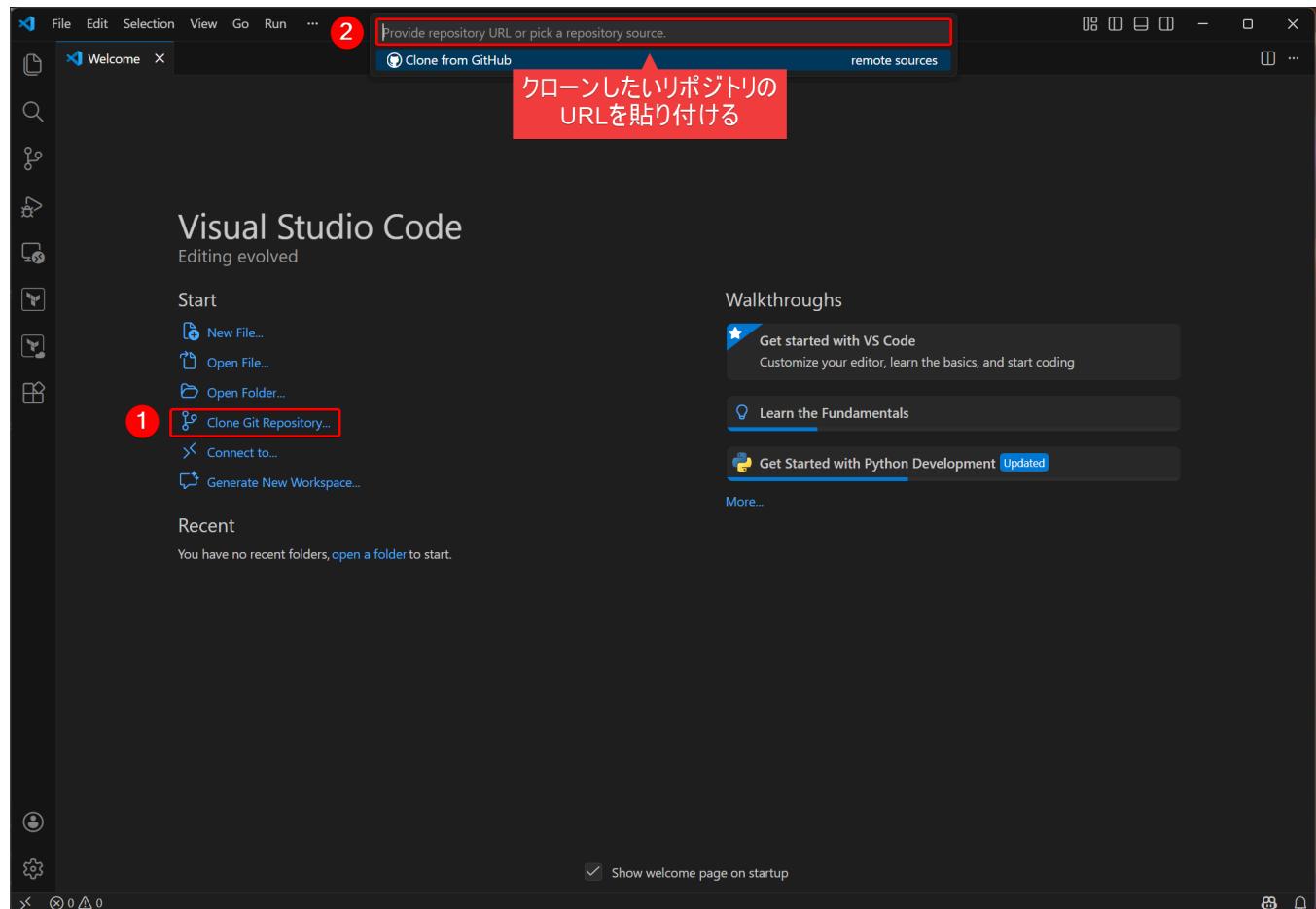
The screenshot shows the Visual Studio Code interface with the following details:

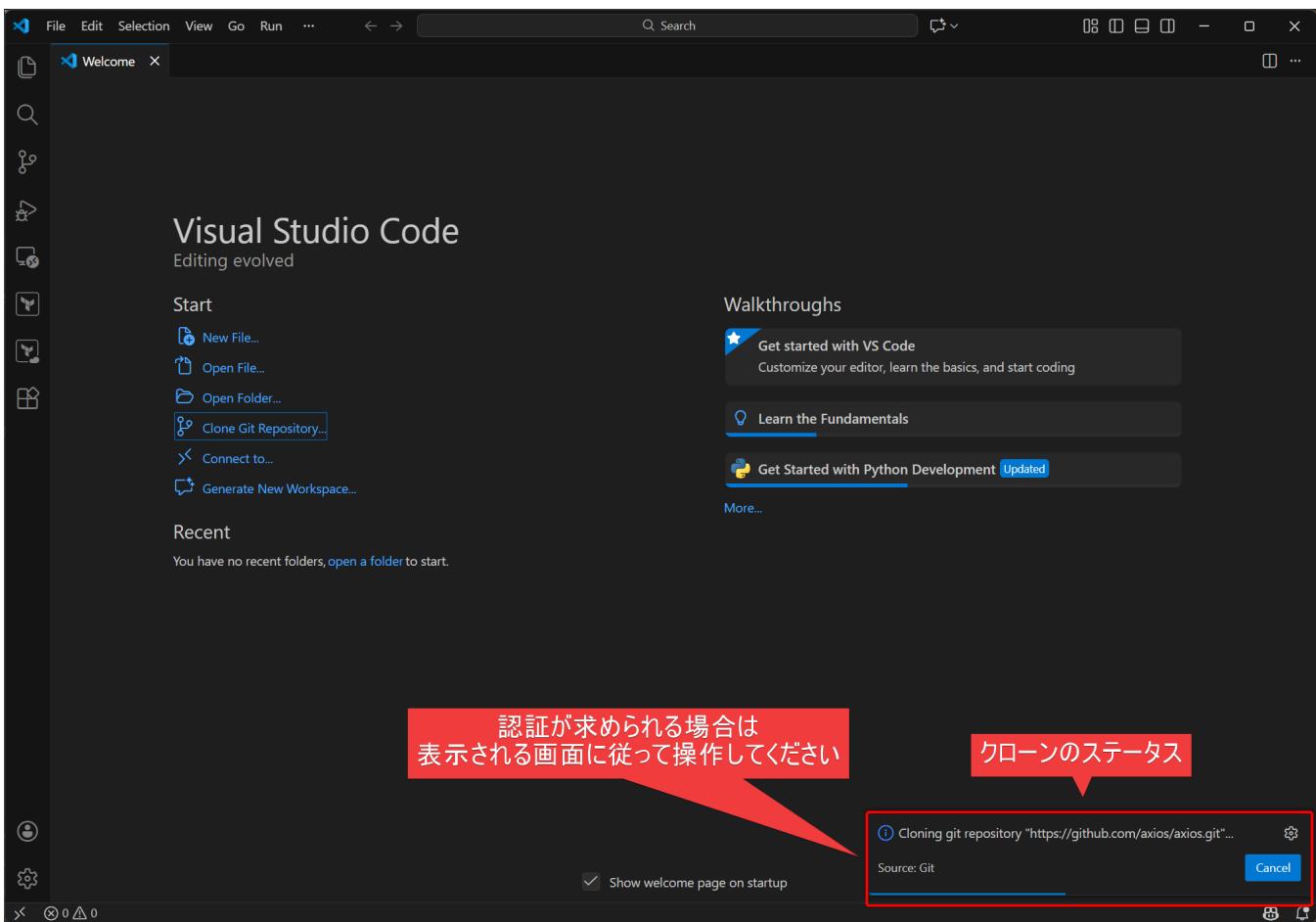
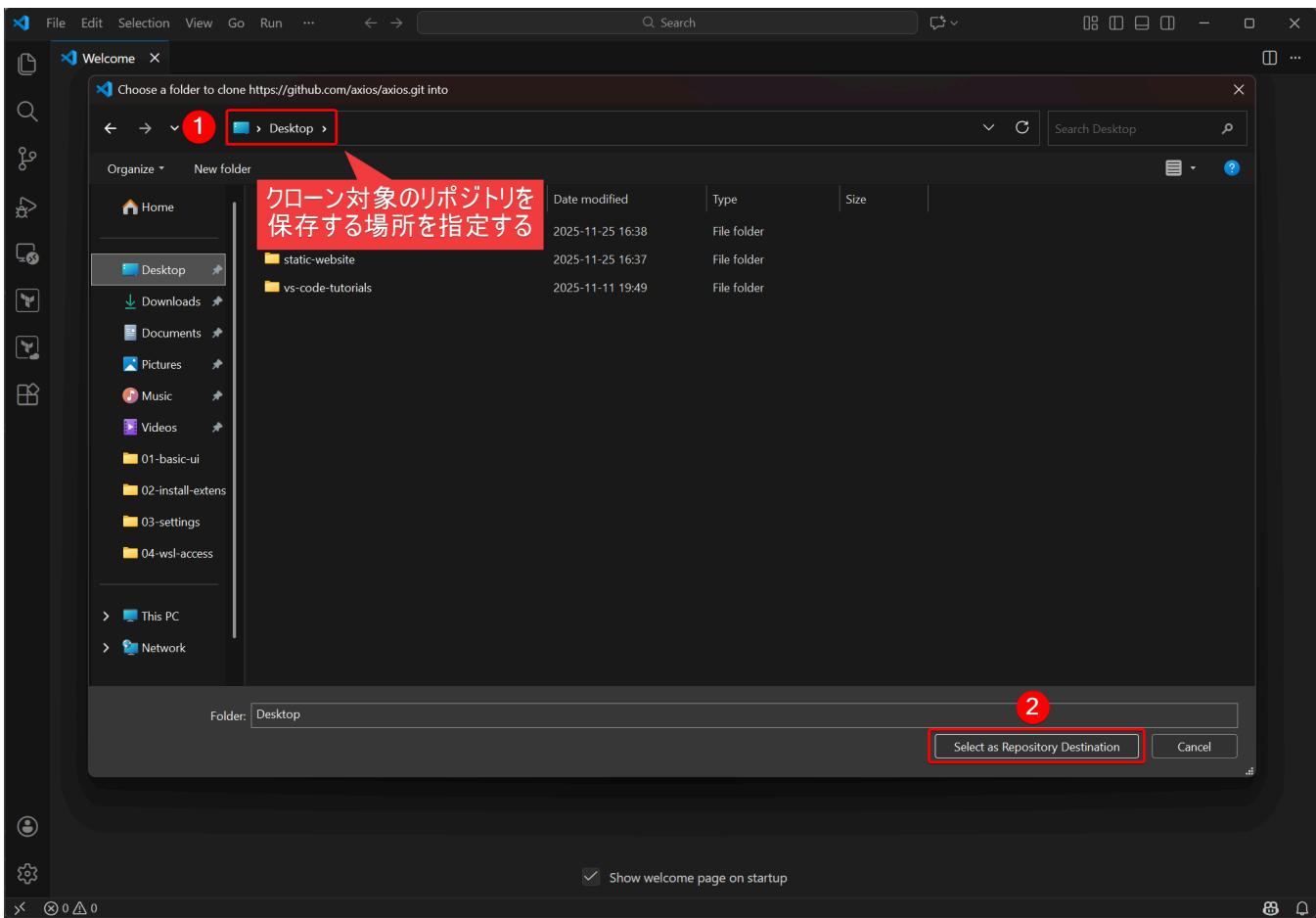
- File Explorer:** On the left, under "OPEN EDITORS", there are files: index.html, index.js, and #style.css. Under "STATIC-WEBSITE", there is a folder "style" containing #style.css.
- Editor:** The main editor window displays index.js with the following code:

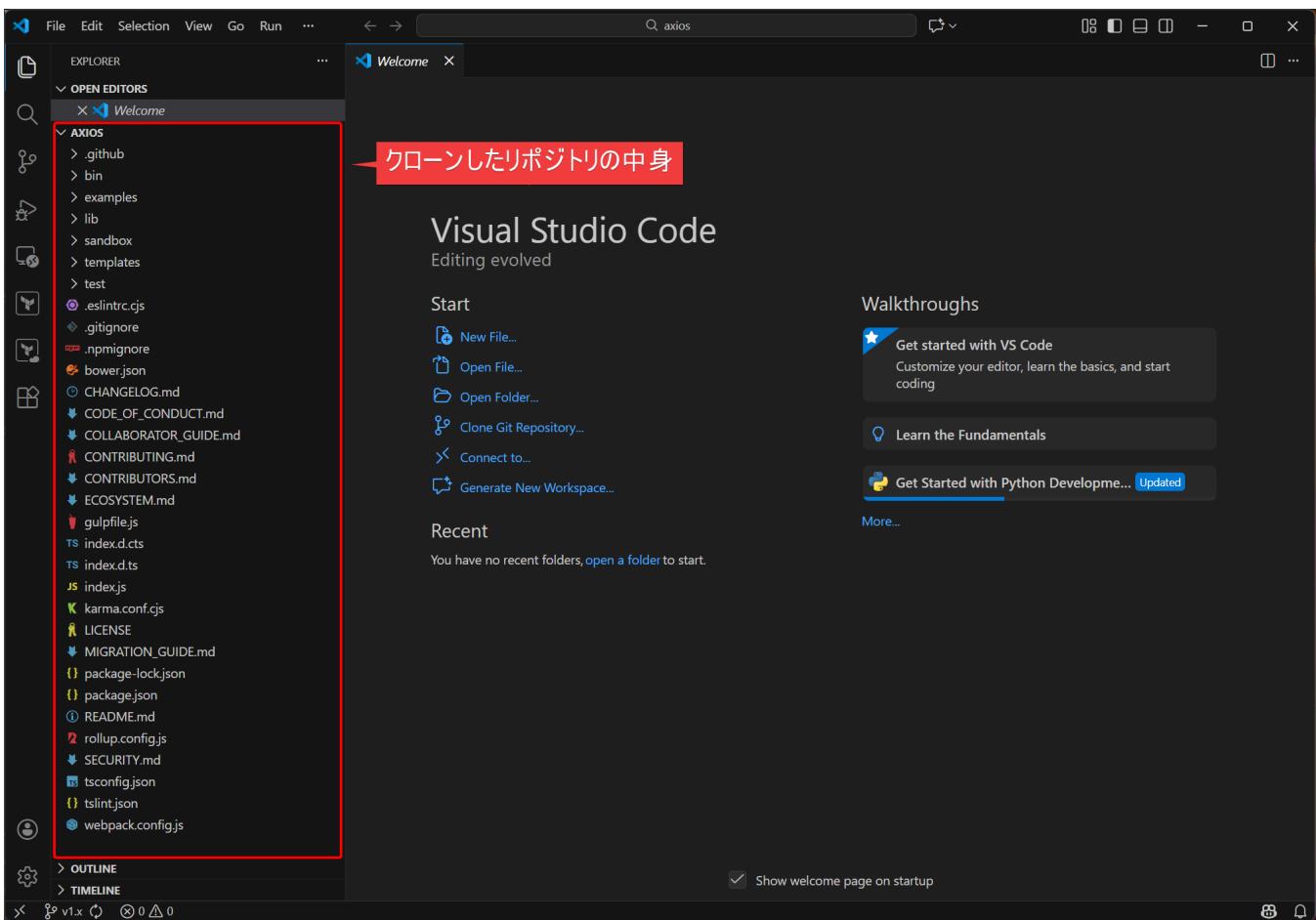
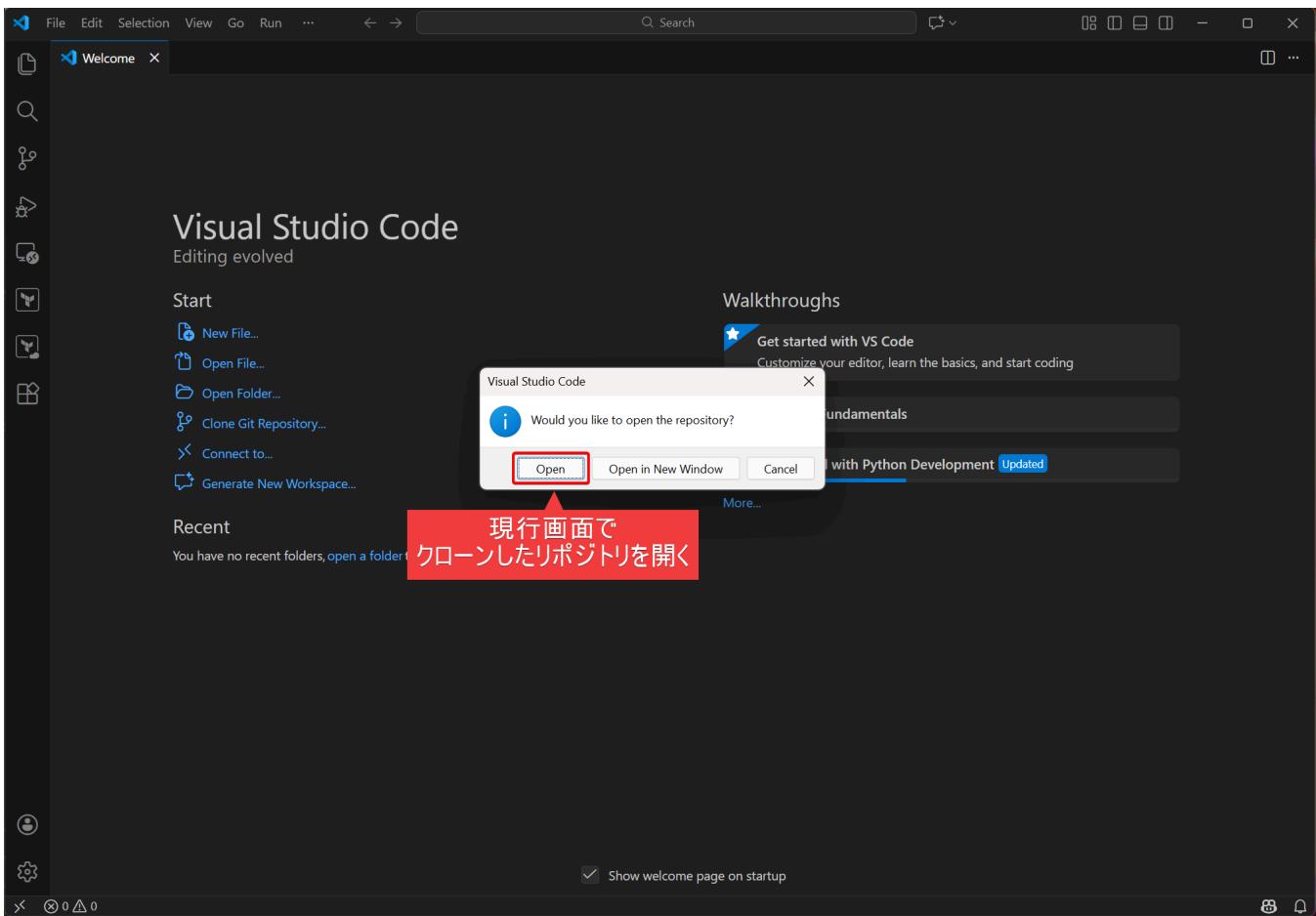
```
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4   console.log(person);
5 }
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 }
```
- Message Bar:** A red box highlights the message "ファイル移動完了後" (File moved successfully) at the bottom of the editor.
- Status Bar:** Shows "Ln 8, Col 21 (6 selected)" and other settings like "Spaces: 2", "UTF-8", "CRLF".

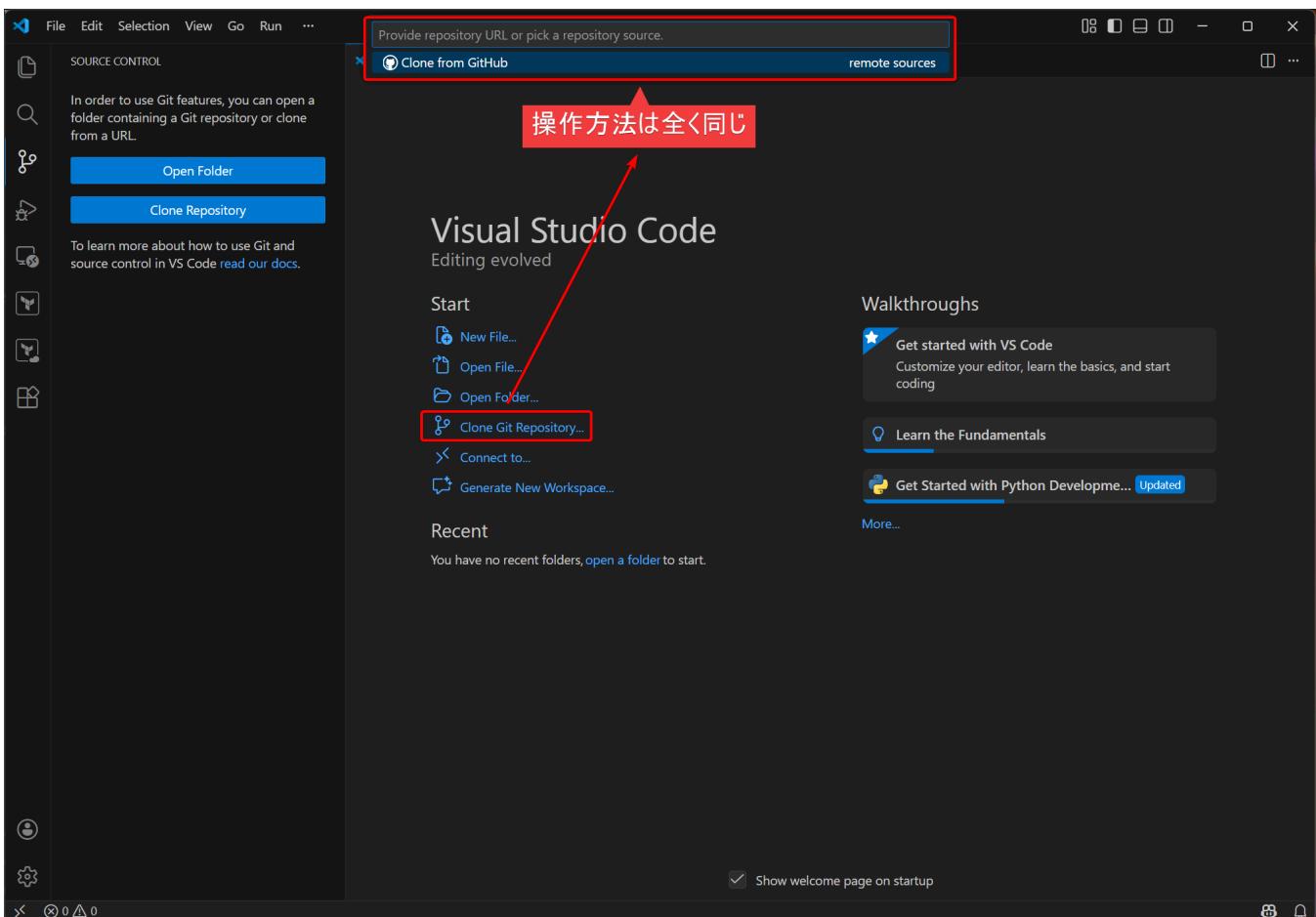
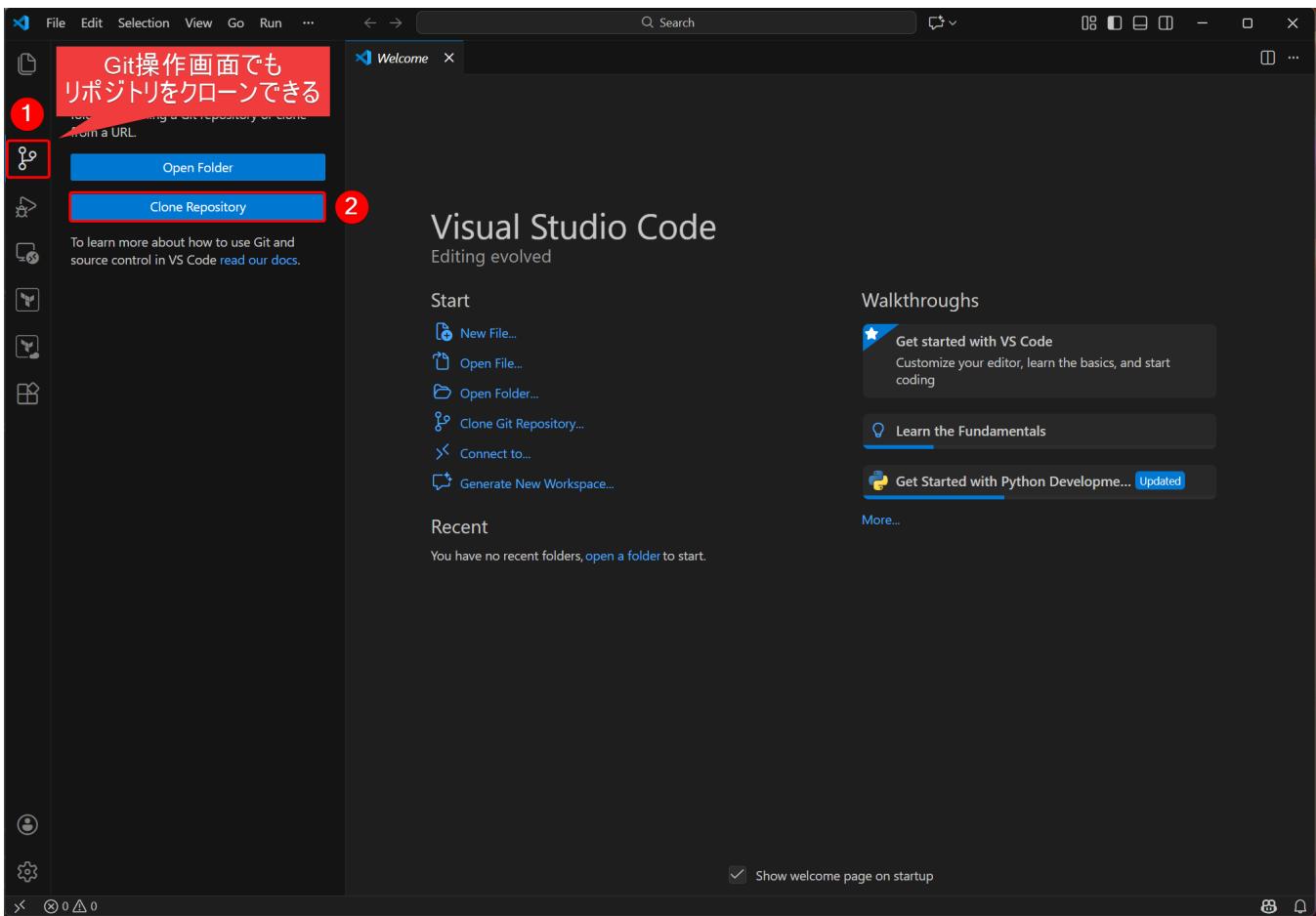


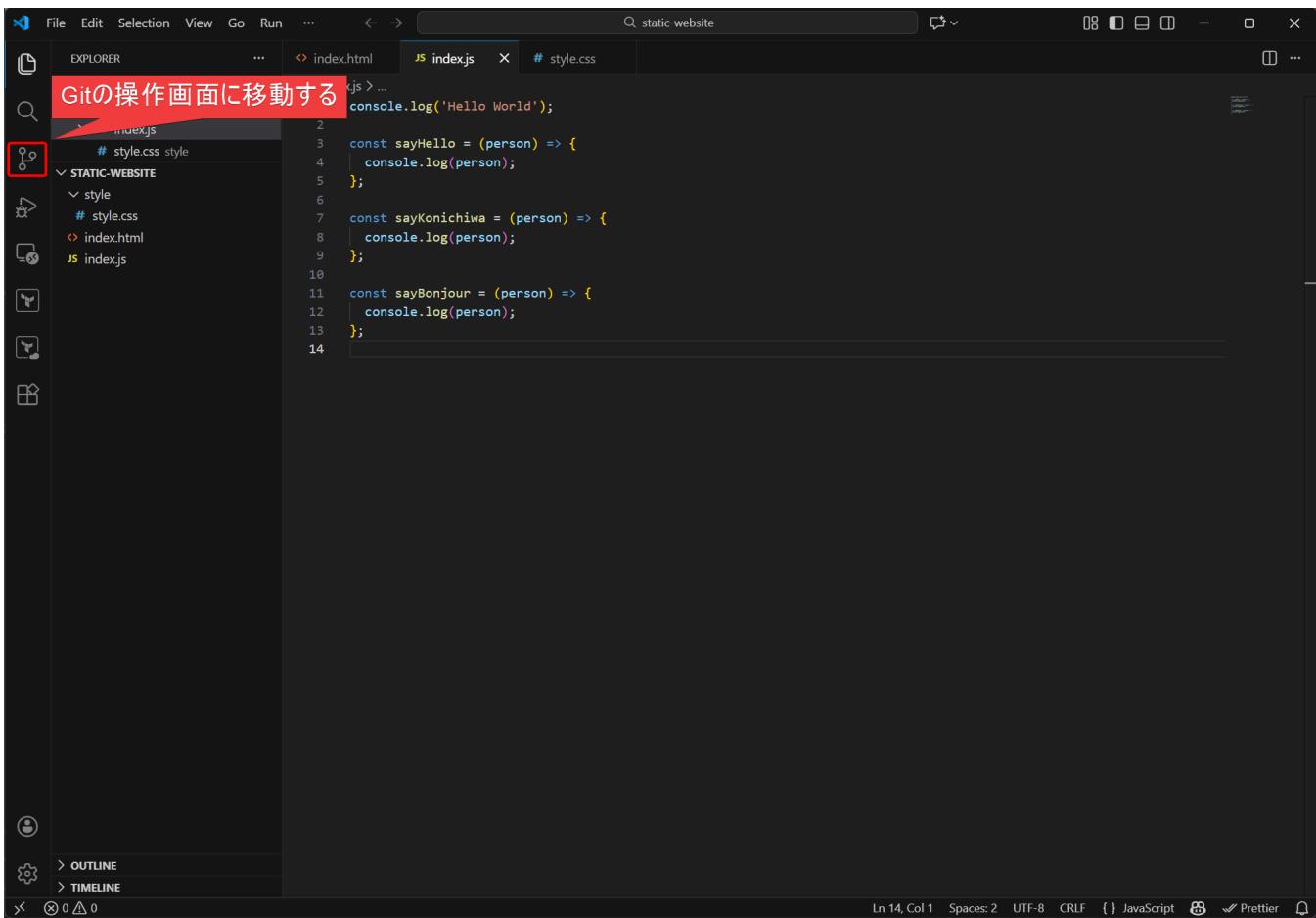
7. Source control











Gitの操作画面に移動する

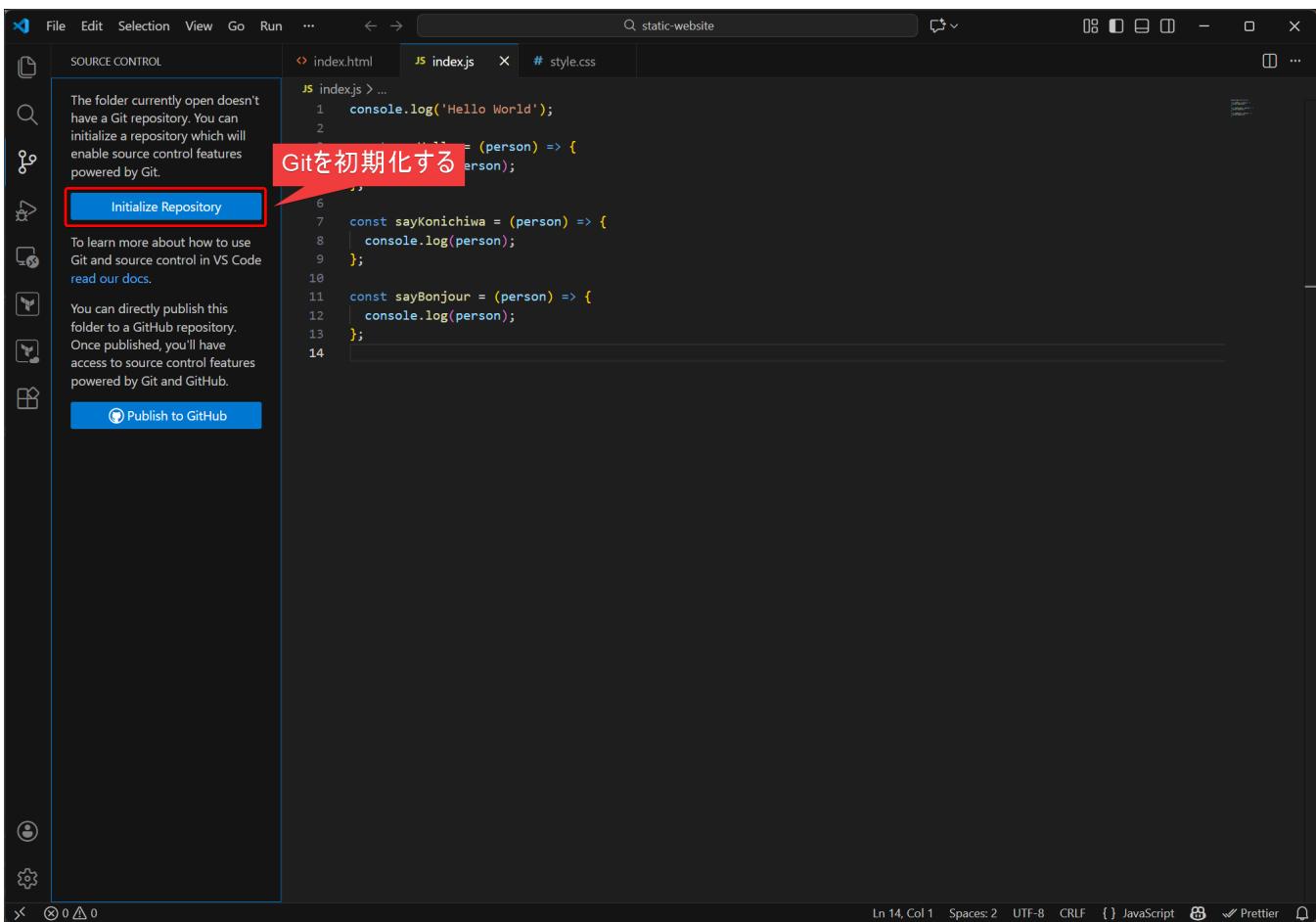
File Edit Selection View Go Run ... index.html JS index.js # style.css

EXPLORER

index.js
style.css style
STATIC-WEBSITE
style
style.css
index.html
JS index.js

```
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4   console.log(person);
5 };
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript Prettier



SOURCE CONTROL

The folder currently open doesn't have a Git repository. You can initialize a repository which will enable source control features powered by Git.

Initialize Repository

To learn more about how to use Git and source control in VS Code read our docs.

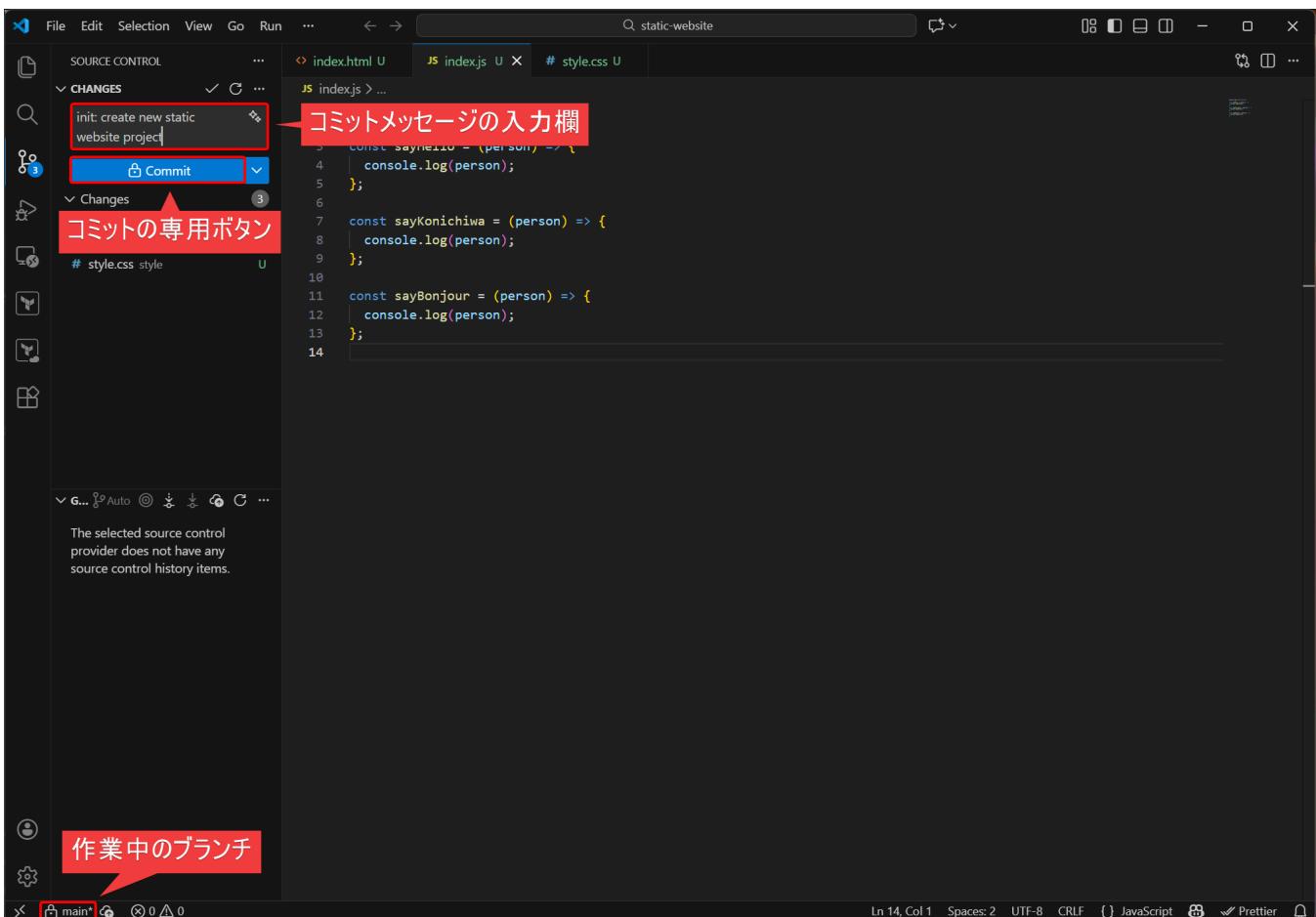
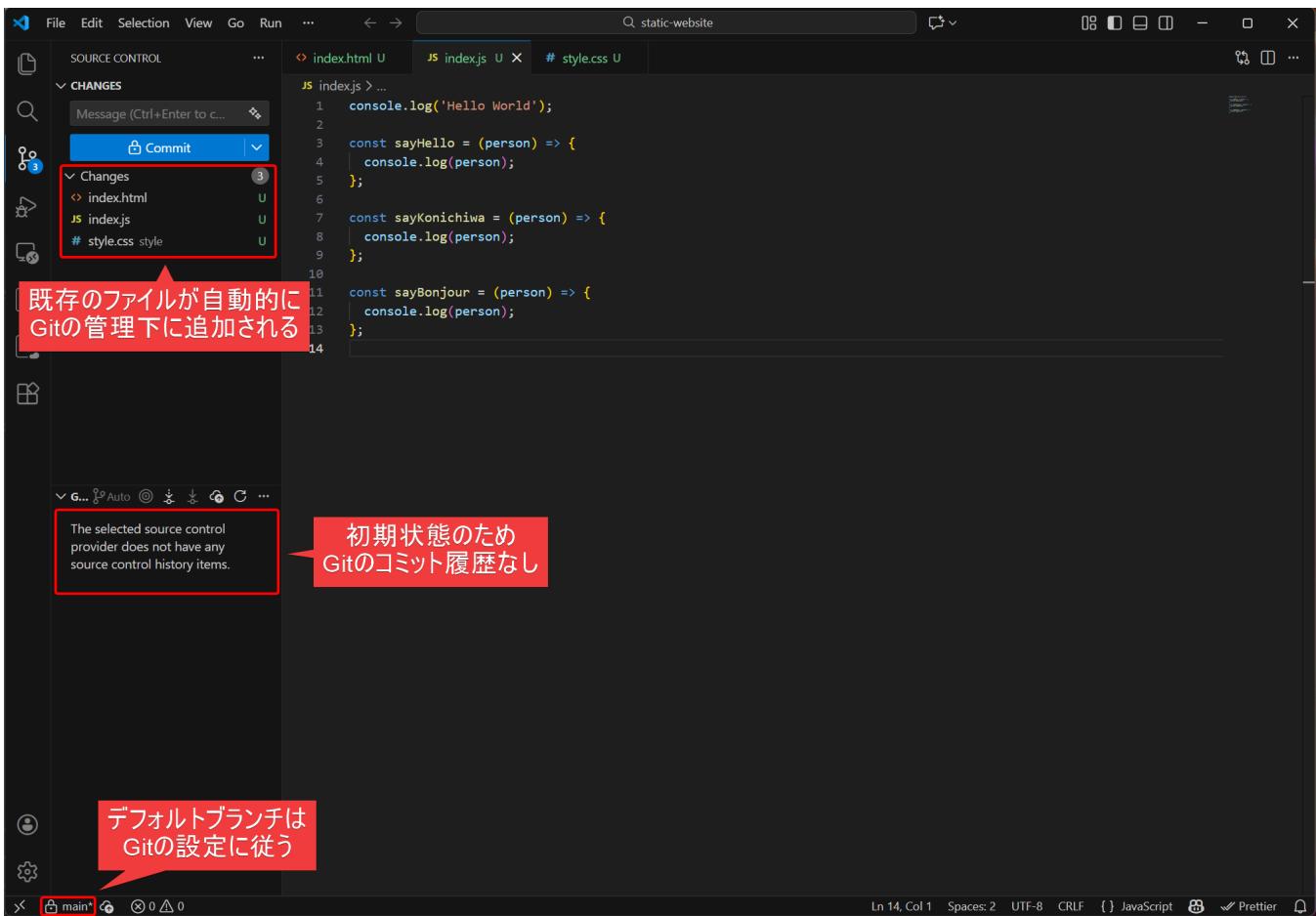
You can directly publish this folder to a GitHub repository. Once published, you'll have access to source control features powered by Git and GitHub.

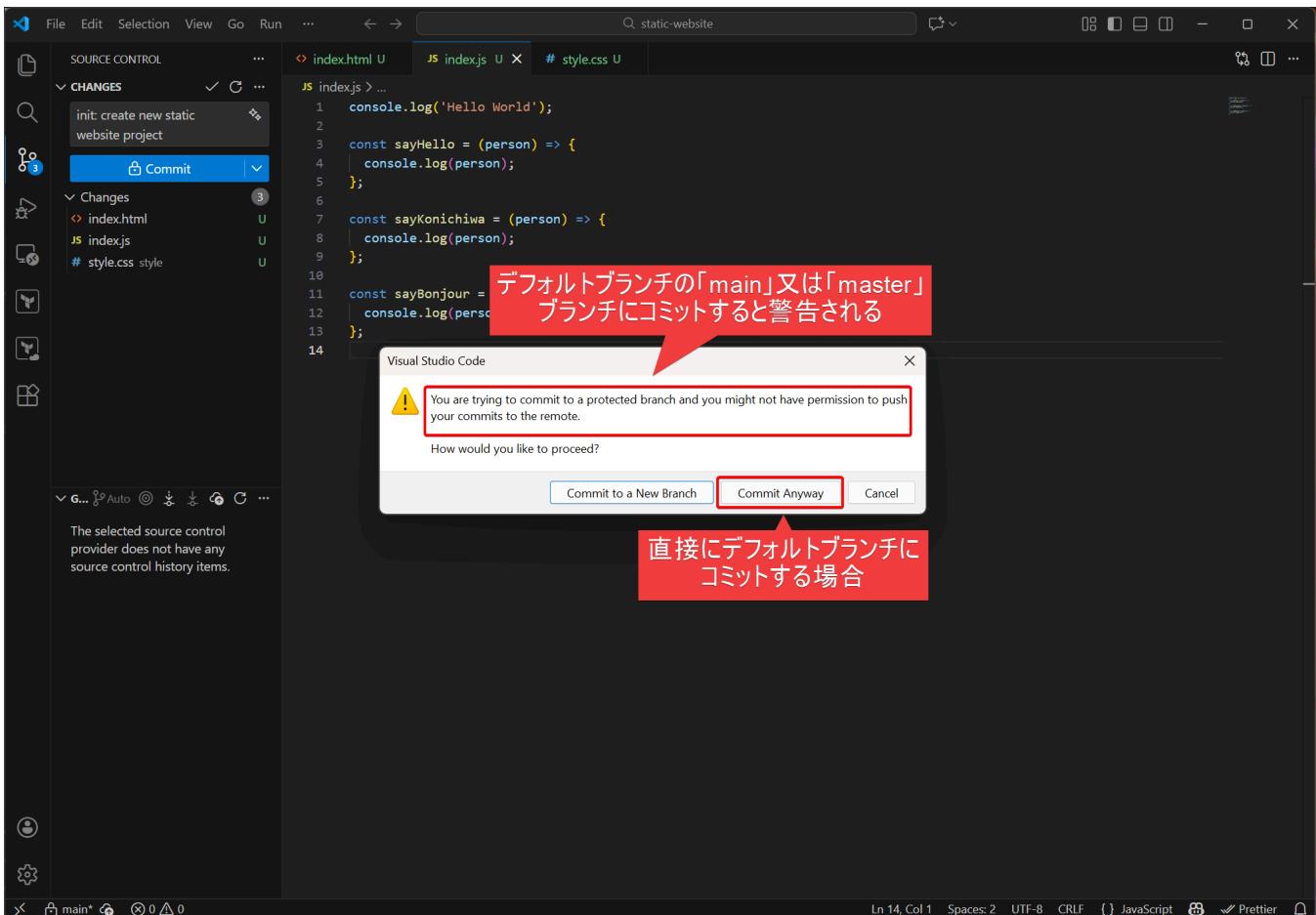
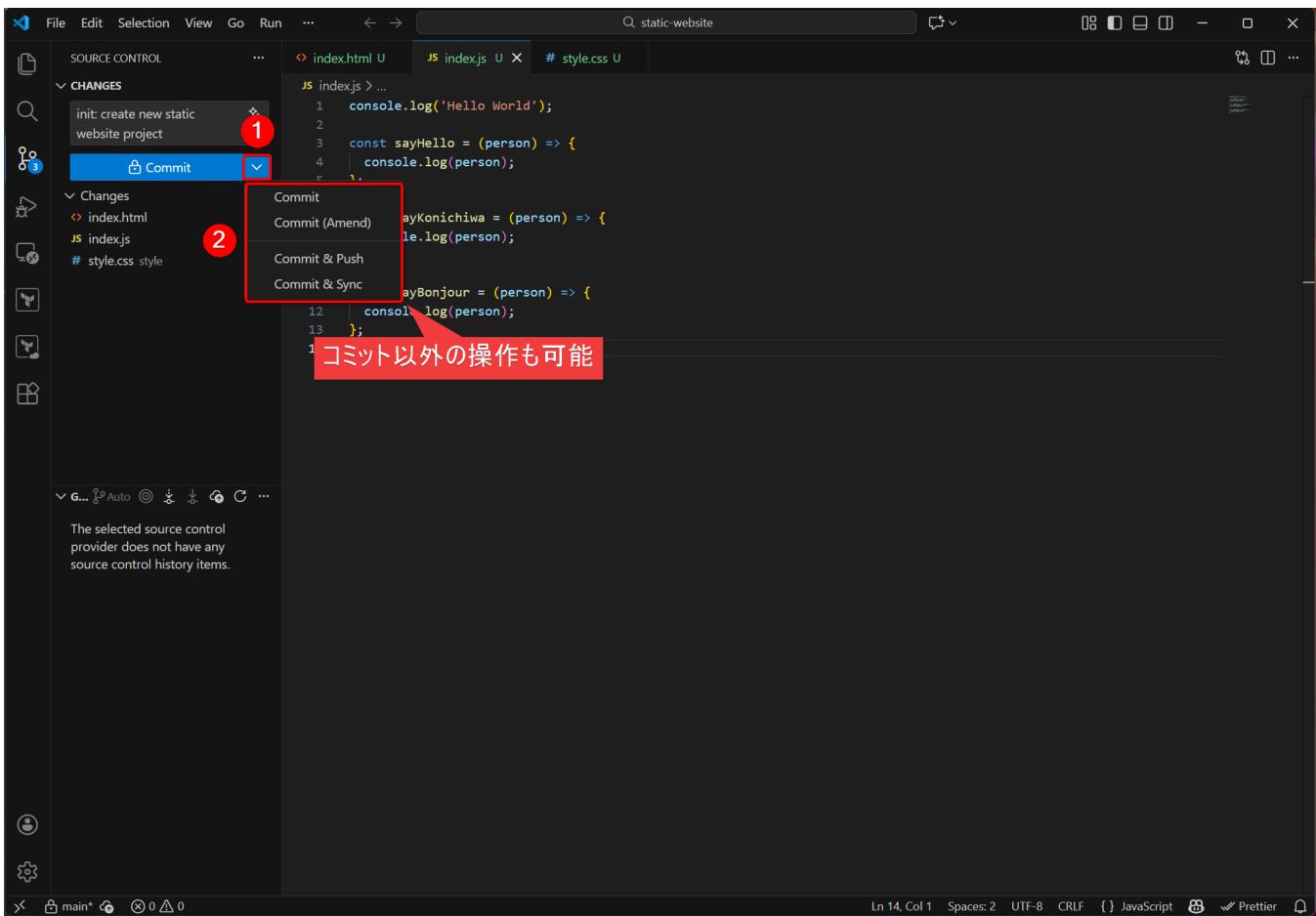
Publish to GitHub

index.html JS index.js # style.css

```
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4   console.log(person);
5 };
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript Prettier





The screenshot shows the VS Code interface with the following details:

- File Explorer (Left):** Shows icons for Source Control, Changes, and Publish Branch.
- Source Control Panel (Top Left):** Displays "Message (Ctrl+Enter to c...)" and a "Publish Branch" button.
- Editor (Top Right):** Shows files index.html, index.js, and style.css. The index.js file contains the following code:

```
index.js > ...
1  console.log('Hello World');
2
3  const sayHello = (person) => {
4    console.log(person);
5  };
6
7  const sayKonichiwa = (person) => {
8    console.log(person);
9  };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```
- Commit Message (Bottom Left):** A red box highlights the message "init: create new static w... @main". An arrow points from this box to the text "コミットしたブランチ" (Committed branch).
- Bottom Status Bar:** Shows "Ln 14, Col 1 Spaces: 2 UTF-8 CRLF {} JavaScript" and a Prettier icon.

The screenshot shows the VS Code interface with the following annotations:

- A red box highlights the commit message "Message (Ctrl+Enter to c...)" in the Changes sidebar.
- A red box highlights the file "index.js" in the Changes sidebar, which has an "M" icon indicating it was modified.
- A red box highlights the word "name" in the code, which is being renamed from "person".
- A red box contains the text: "「person」→「name」に変更した場合".
- A red box contains the text: "「M」は「Modify(修正)」を意味する".

The code in index.js is as follows:

```
1 console.log('Hello World');
2
3 const sayHello = name => {
4   console.log(name);
5 }
6
7 const sayBonjour = person => {
8   console.log(person);
9 }
10
11
12
13
14
```

变更箇所をクリックすると
変更内容が表示される

Commit

Changes
index.js

変更された数
(フォルダー全体)

index.js Git Local Changes (Working Tree) - 1 of 1 change

変更前

変更後

```
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4   console.log(person);
5 };
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

変更前の状態を戻す

Revert Change

index.html JS index.js M X # style.css

index.js > sayHello

```
1 console.log('Hello World');
2
3 const sayHello = (name) => [
4   console.log(name);
5 ];
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

变更前の状態に戻される

```
index.html js indexjs # style.css
JS indexjs > [sayHello
1 console.log('Hello World');
2
3 const sayHello = (person) => [
4   console.log(person);
5 ];
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

init: create new static w... @main

ファイル全体を一括で前の状態に戻したい場合

Kei (1 minute ago) Ln 4, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript ✓ Prettier

File Edit Selection View Go Run ... static-website

SOURCE CONTROL ...

CHANGES

Message (Ctrl+Enter to c...)

Publish Branch

Changes

index.js

Commit

+ M

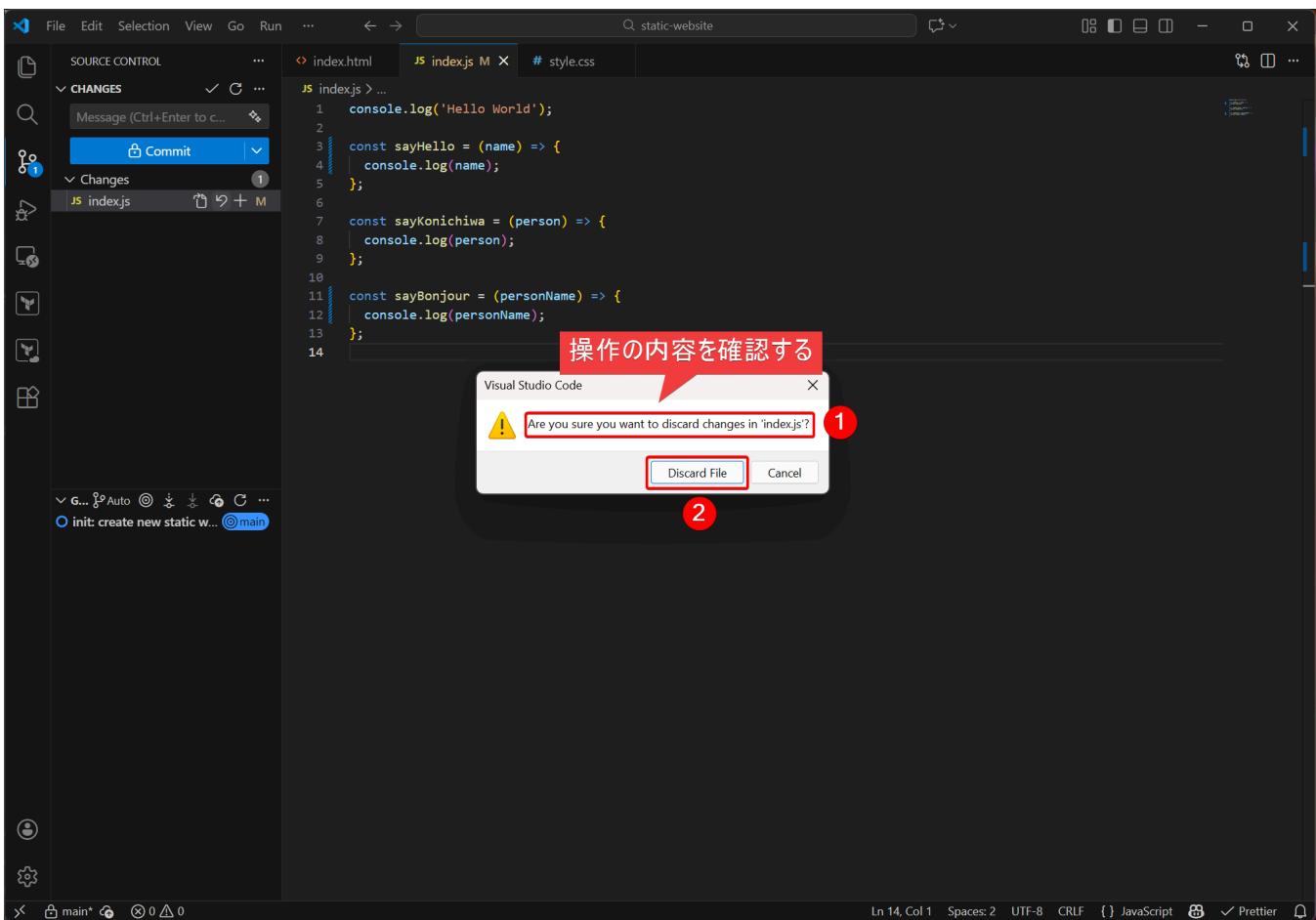
```
index.html js indexjs M # style.css
JS indexjs > ...
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

init: create new static w... @main

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript ✓ Prettier

例: 2か所を変更した場合

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4     console.log(name);
5 };
6
7 const sayKonichiwa = (person) => {
8     console.log(person);
9 };
10
11 const sayBonjour = (personName) => {
12     console.log(personName);
13 };
14
```



File Edit Selection View Go Run ... static-website

SOURCE CONTROL

CHANGES

Message (Ctrl+Enter to C...)

Publish Branch

```
index.html JS index.js # style.css
```

JS index.js > ...

```
1 console.log('Hello World');
2
3 const sayHello = (person) => {
4   console.log(person);
5 };
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

G... Auto C ...

init: create new static w... @main

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript ✓ Prettier

File Edit Selection View Go Run ... static-website

SOURCE CONTROL

CHANGES

Message (Ctrl+Enter to C...)

Commit

Changes

index.js M

```
index.html JS index.js M # style.css
```

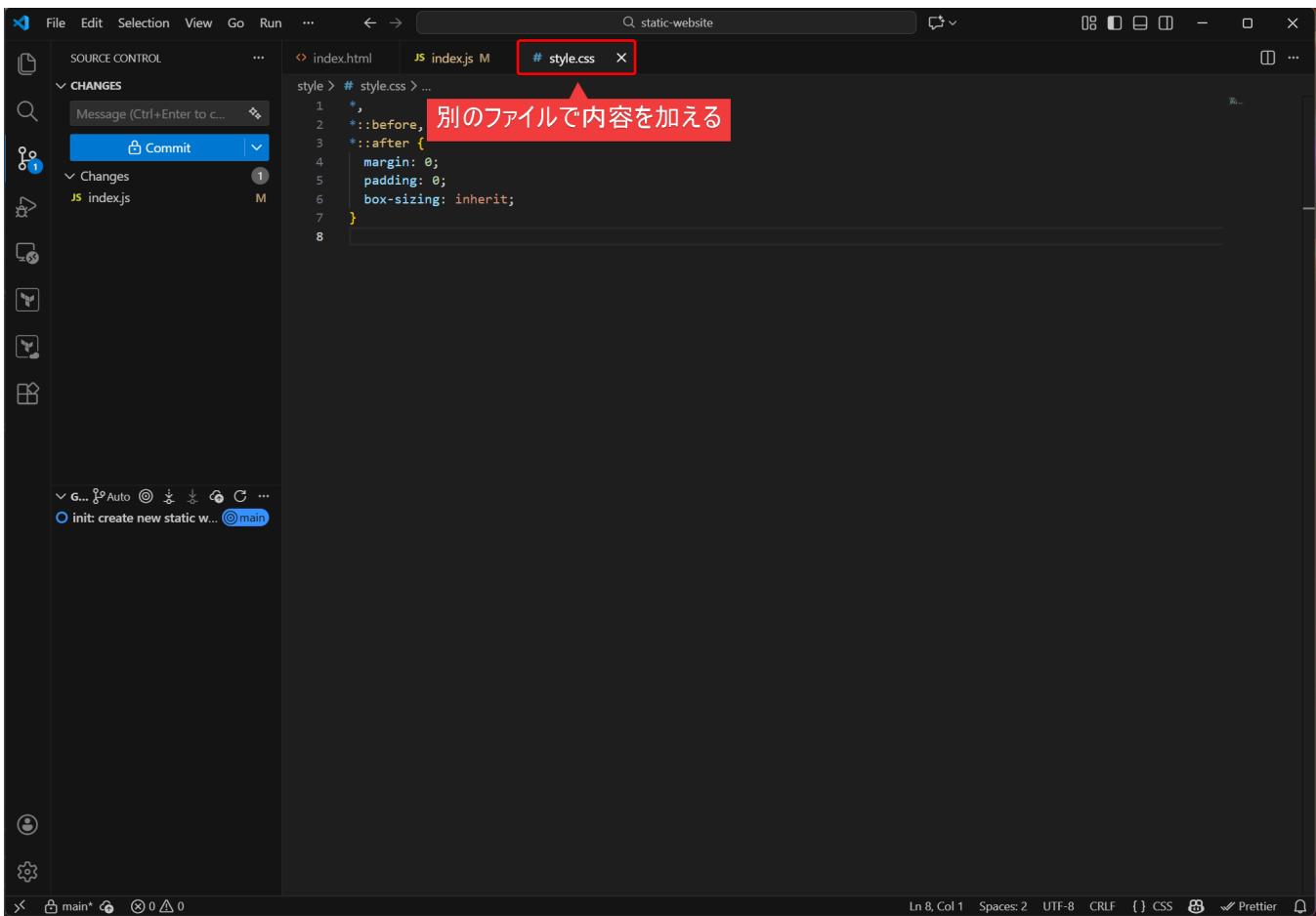
JS index.js > ...

```
1 console.log('Hello World');
2
3 const sayHello = (name) => {
4   console.log(name);
5 };
6
7 const sayKonichiwa = (person) => {
8   console.log(person);
9 };
10
11 const sayBonjour = (person) => {
12   console.log(person);
13 };
14
```

G... Auto C ...

init: create new static w... @main

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript ✓ Prettier



File Edit Selection View Go Run ... static-website

SOURCE CONTROL ...

CHANGES

Message (Ctrl+Enter to c...)

Commit

Changes

index.js M

style.css X

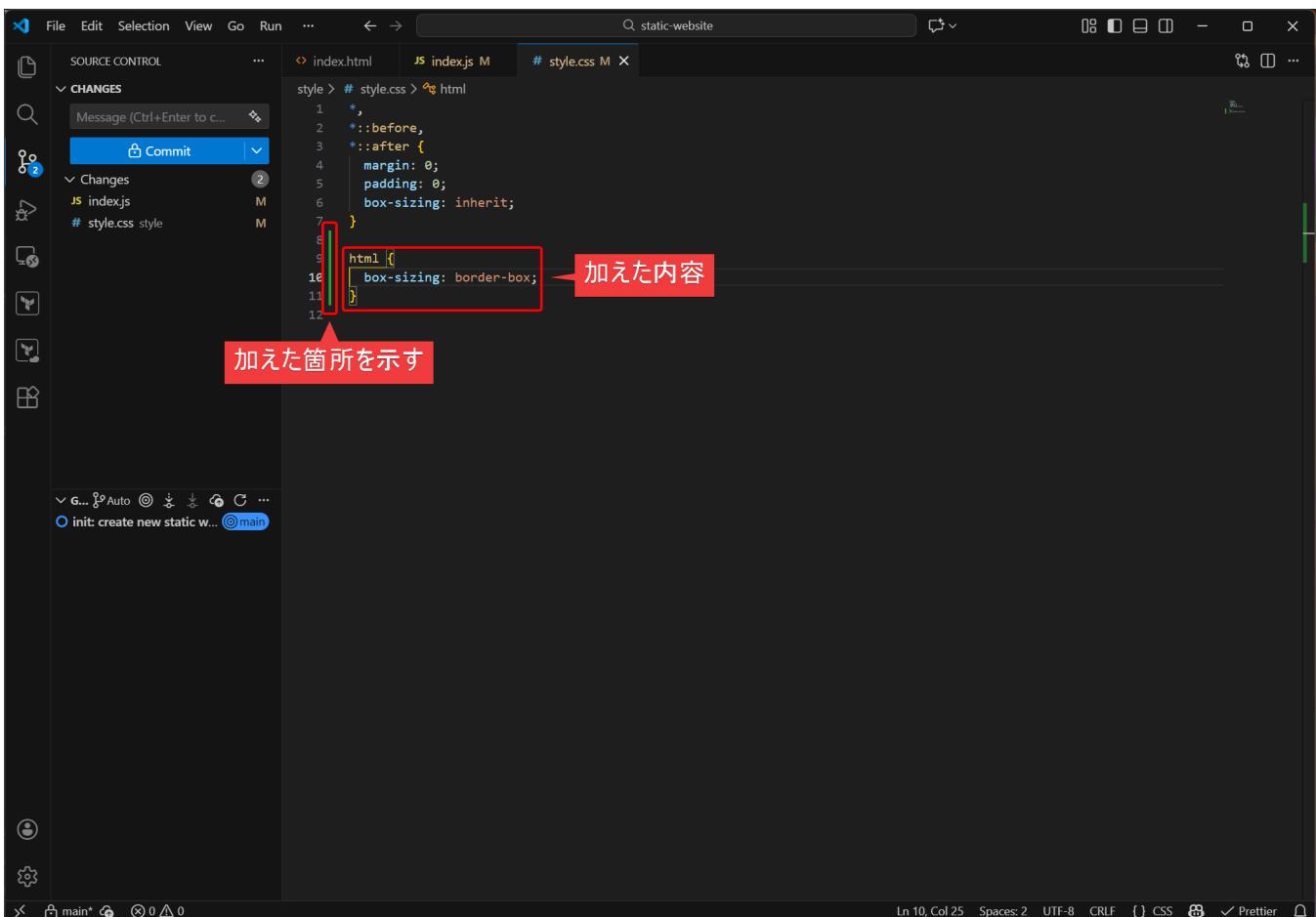
style > # style.css > ...

```
1 *,
2 *::before,
3 *::after {
4 margin: 0;
5 padding: 0;
6 box-sizing: inherit;
7 }
```

G... Auto C ...

init: create new static w... @main

Ln 8, Col 1 Spaces: 2 UTF-8 CRLF {} CSS ✓ Prettier



File Edit Selection View Go Run ... static-website

SOURCE CONTROL ...

CHANGES

Message (Ctrl+Enter to c...)

Commit

Changes

index.js M

style.css M

style > # style.css > html

```
1 *,
2 *::before,
3 *::after {
4 margin: 0;
5 padding: 0;
6 box-sizing: inherit;
7 }
8
9 html {
10   box-sizing: border-box;
11 }
```

加えた箇所を示す

G... Auto C ...

init: create new static w... @main

Ln 10, Col 25 Spaces: 2 UTF-8 CRLF {} CSS ✓ Prettier

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a tree view with "Changes" expanded, showing "index.js" and "# style.css" as modified files.
- Source Control:** Shows a "Commit" button.
- Editor:** The "style.css" file is open, showing the following code:

```
* {  
    margin: 0;  
    padding: 0;  
    box-sizing: inherit;  
}  
  
html {  
    box-sizing: border-box;  
}
```

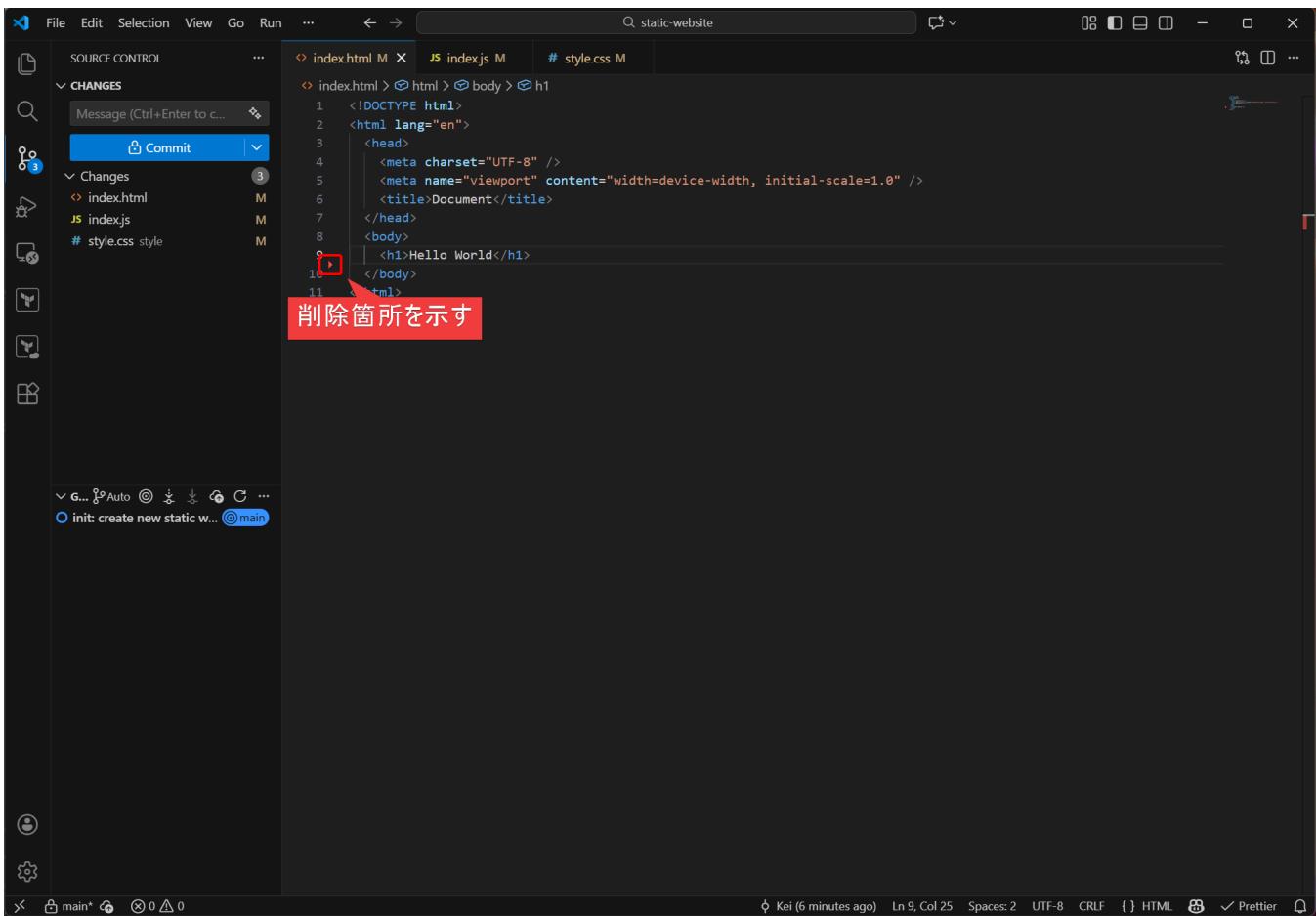
A red box highlights the last line "11 }". A red callout bubble above it says "クリックすると
加えた内容の詳細を確認できる" (Click here to view the details of the added content).
- Bottom Status Bar:** Shows "main* 0 0 0 0" and "init: create new static w... @main".

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a tree view with "Changes" expanded, showing "index.js" and "# style.css" as modified files.
- Source Control:** Shows a "Commit" button.
- Editor:** The "index.html" file is open, showing the following code:

```
<!DOCTYPE html>  
<html>  
    <head>  
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />  
        <title>Document</title>  
    </head>  
    <body>  
        <h1>Hello World</h1>  
        <div>Person</div>  
    </body>  
</html>
```

A red box highlights the line "<div>Person</div>". A red callout bubble to its left says "別のファイルで
削除を行う" (Delete in another file).
- Bottom Status Bar:** Shows "main* 0 0 0 0" and "init: create new static w... @main".



SOURCE CONTROL

CHANGES

Message (Ctrl+Enter to C...)

Commit

Changes

index.html M

index.js M

style.css M

index.html > body > h1

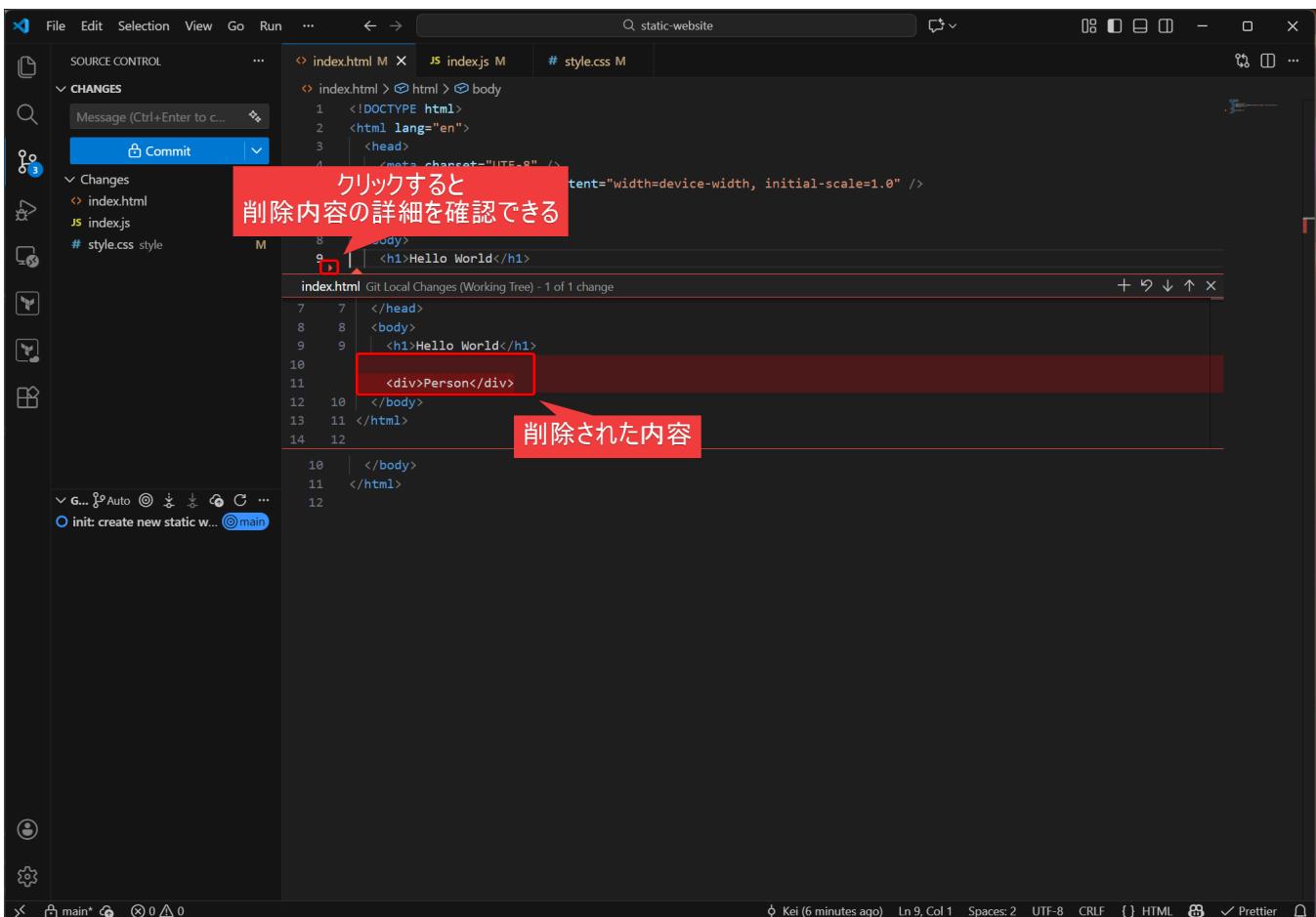
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1>Hello World</h1>
  </body>
</html>
```

削除箇所を示す

G... Auto ⌂ ...

init: create new static w... @main

Kei (6 minutes ago) Ln 9, Col 25 Spaces: 2 UTF-8 CRLF { } HTML ✓ Prettier



SOURCE CONTROL

CHANGES

Message (Ctrl+Enter to C...)

Commit

Changes

index.html M

index.js M

style.css M

index.html > body > h1

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1>Hello World</h1>
    <div>Person</div>
  </body>
</html>
```

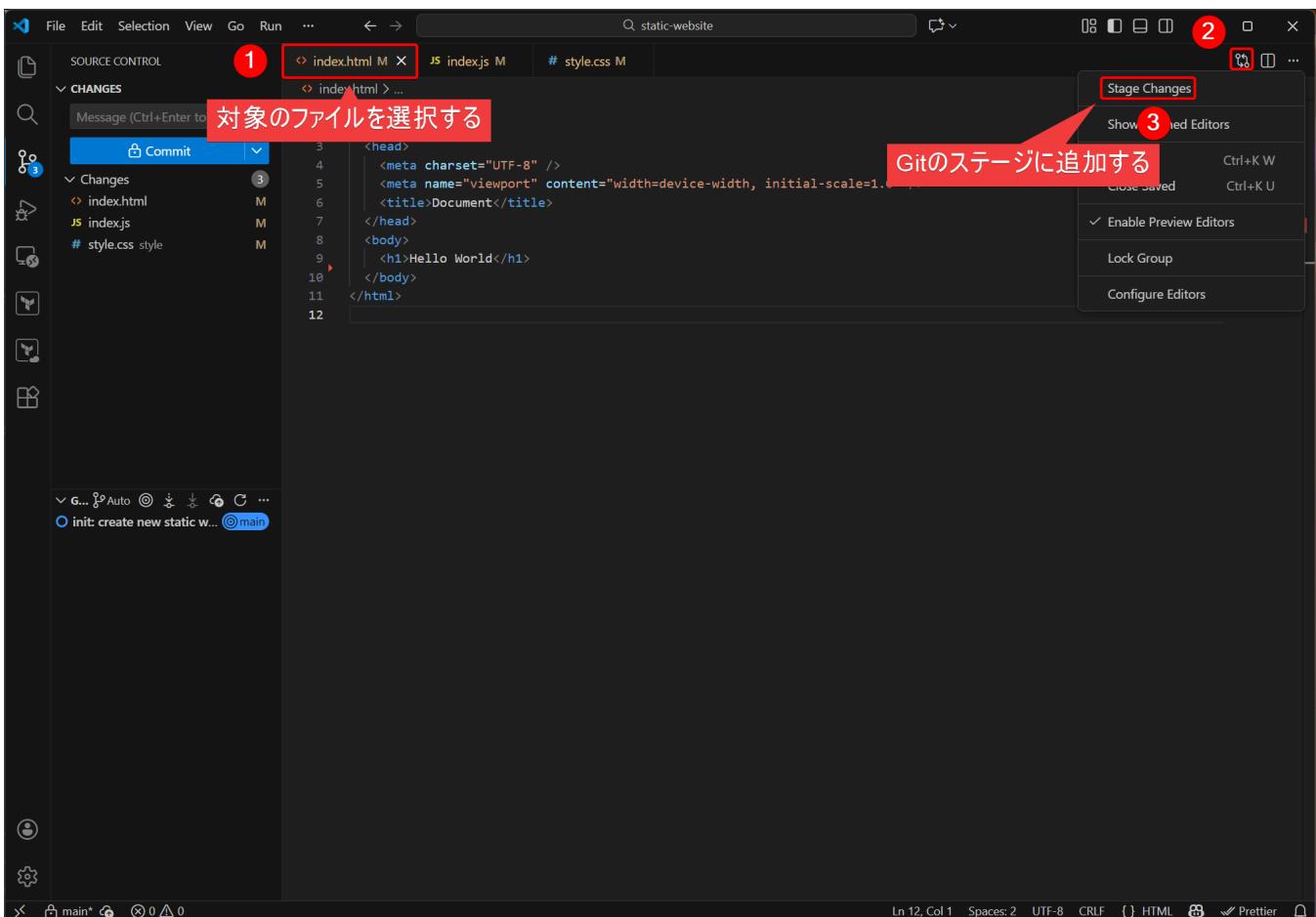
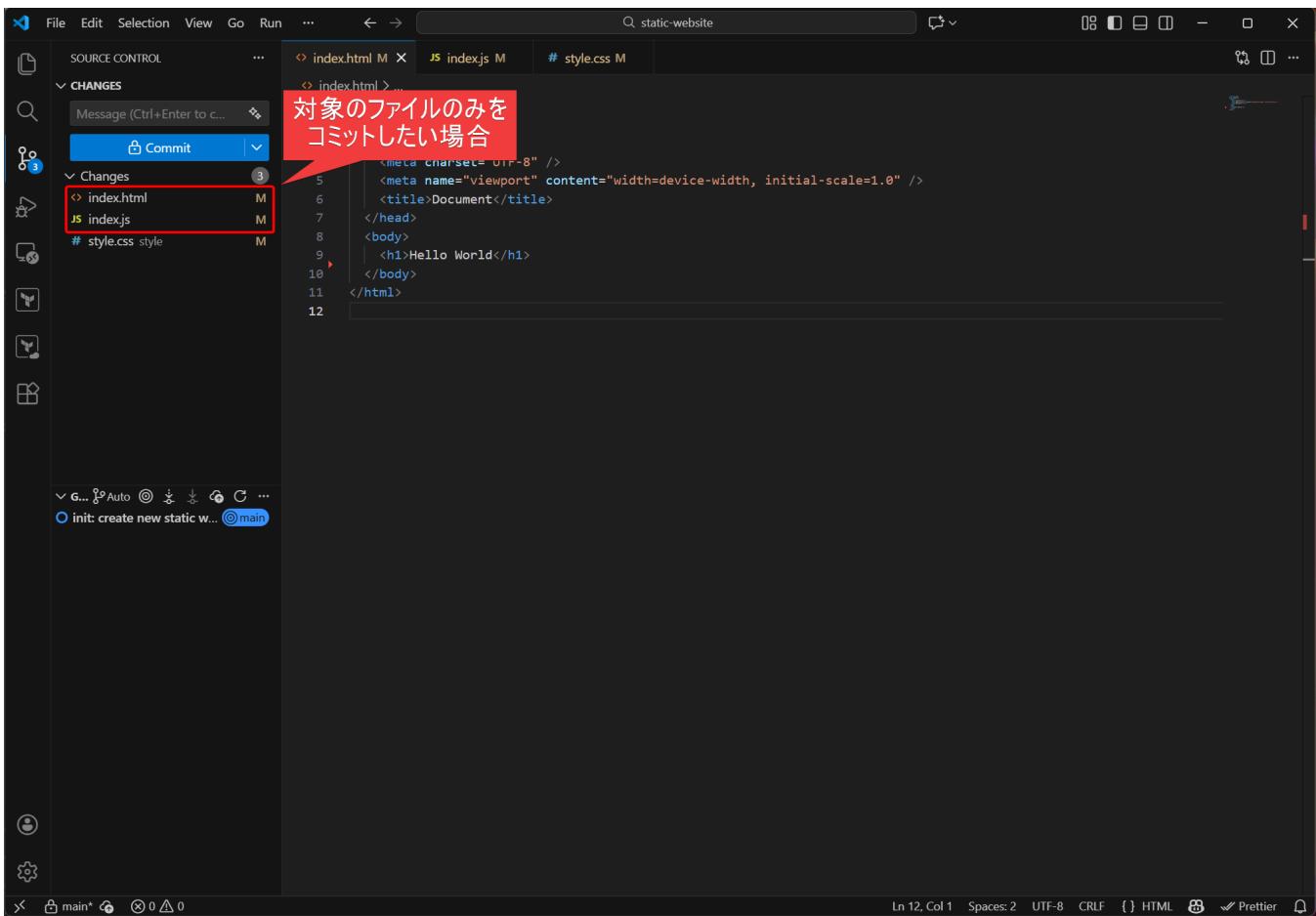
削除内容の詳細を確認できる

削除された内容

G... Auto ⌂ ...

init: create new static w... @main

Kei (6 minutes ago) Ln 9, Col 1 Spaces: 2 UTF-8 CRLF { } HTML ✓ Prettier



File Edit Selection View Go Run ... static-website

SOURCE CONTROL

CHANGES

Message (Ctrl+Enter to C...)

Commit

Staged Changes 1

index.html M

Changes 2

JS index.js M

style.css M

index.html > ...

ステージに追加された
ファイル一覧

```
<!DOCTYPE html><html lang="ja">
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Document</title>
</head>
<body>
  <h1>Hello World</h1>
</body>
</html>
```

G... Auto ...

init: create new static w... @main

Ln 12, Col 1 Spaces: 2 UTF-8 CRLF { } HTML Prettier

File Edit Selection View Go Run ... static-website

SOURCE CONTROL

CHANGES

Message (Ctrl+Enter to C...)

Commit

Staged Changes 1

index.html M

Changes 2

JS index.js M

style.css M

index.js > ...

```
console.log('Hello World');

const sayHello = (name) => {
  console.log(name);
};

const sayKonichiwa = (person) => {
  console.log(person);
};

const sayBonjour = (person) => {
  console.log(person);
};
```

Gitステージに他のファイルを
追加したい場合は同じ手順で行う

Stage Changes

Show 3 saved Editors

Close All Ctrl+K W

Close Saved Ctrl+K U

Enable Preview Editors

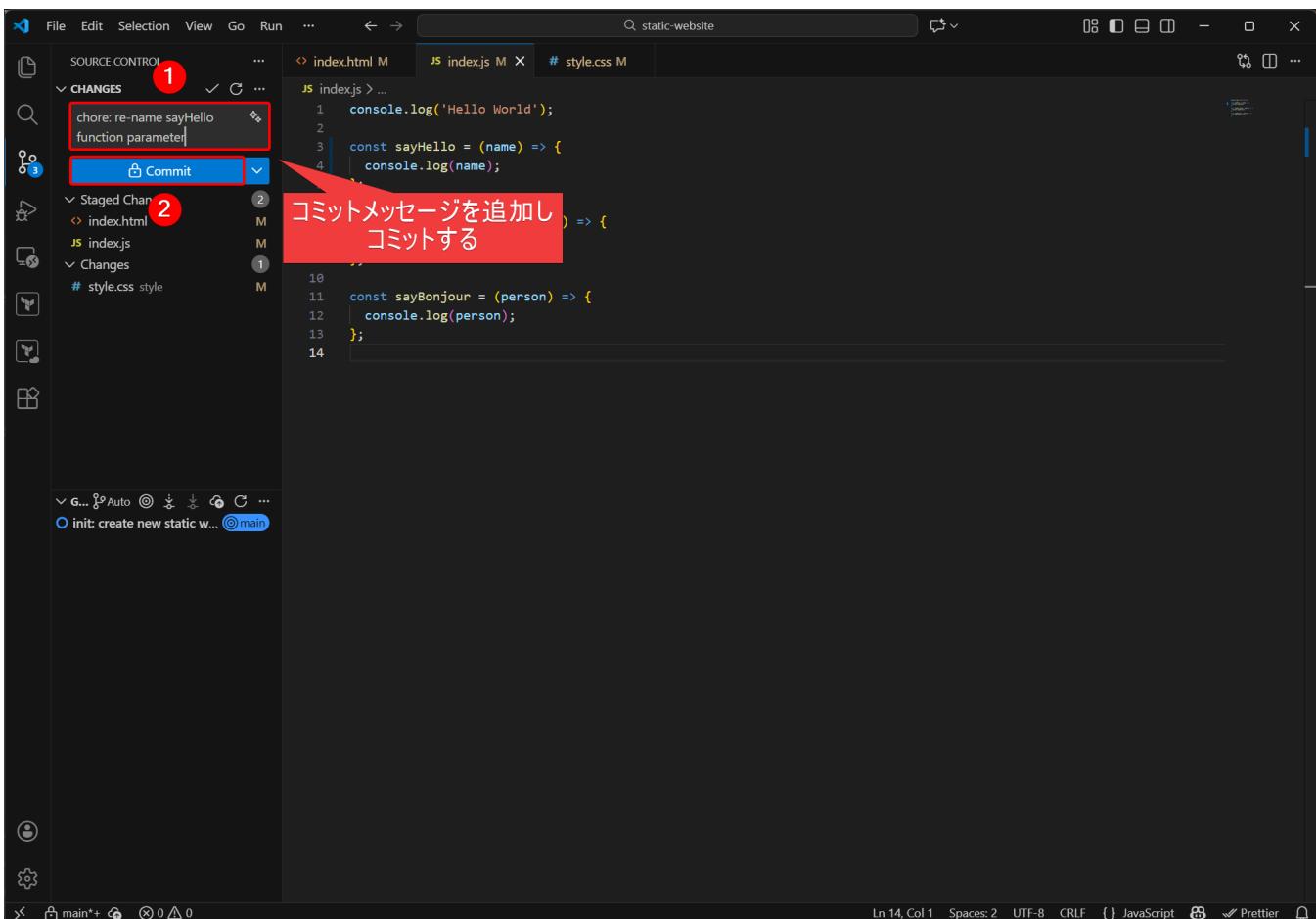
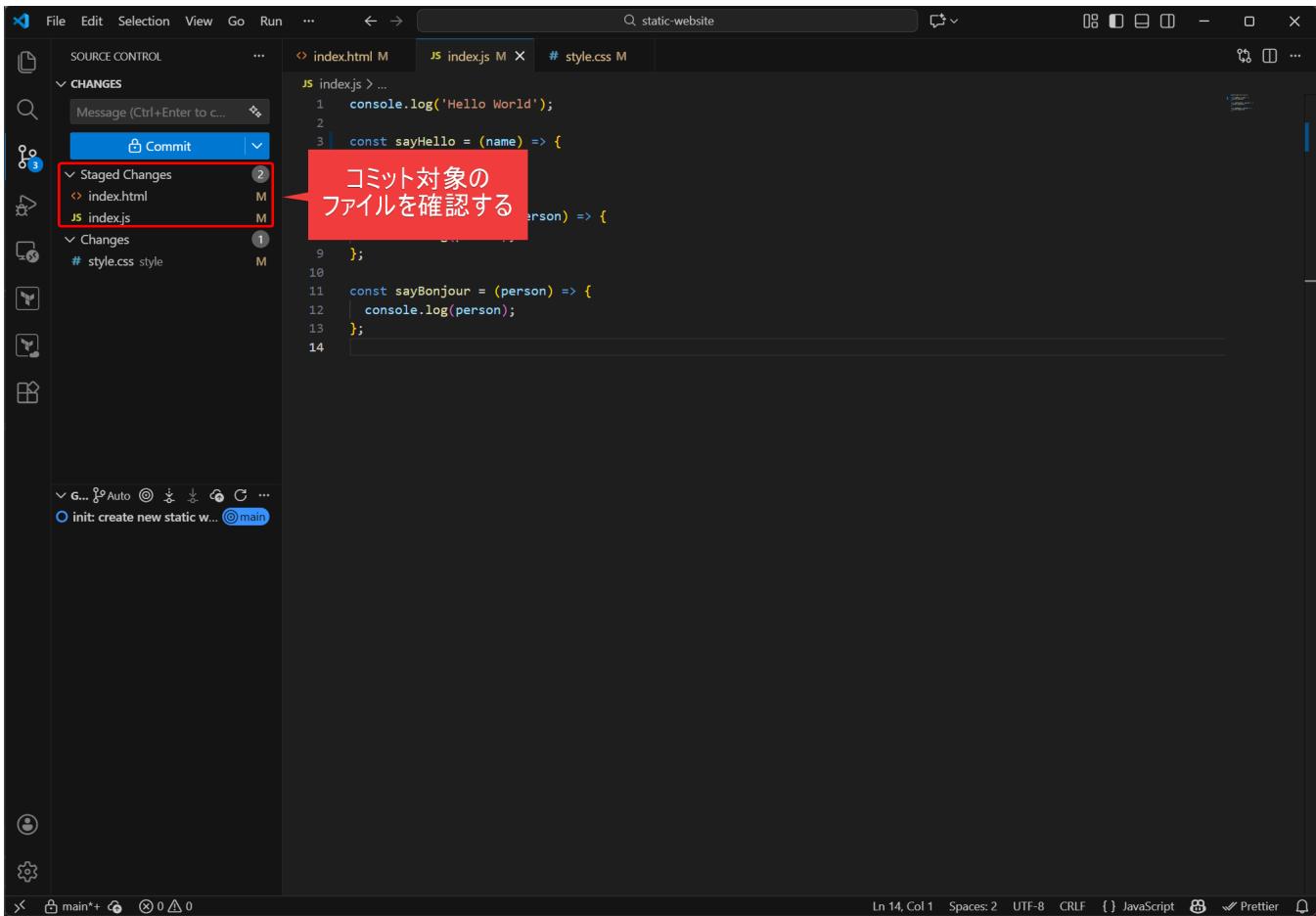
Lock Group

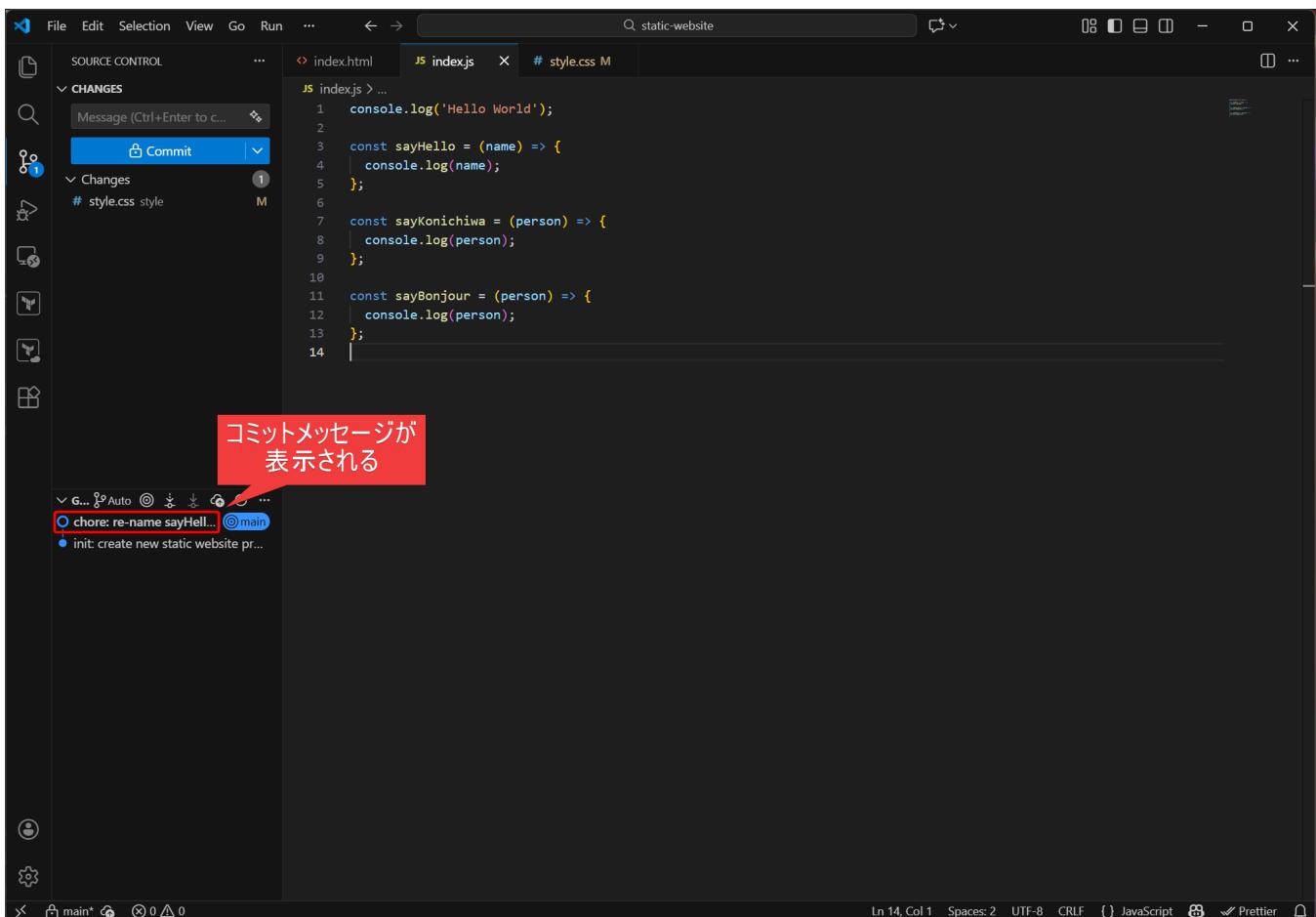
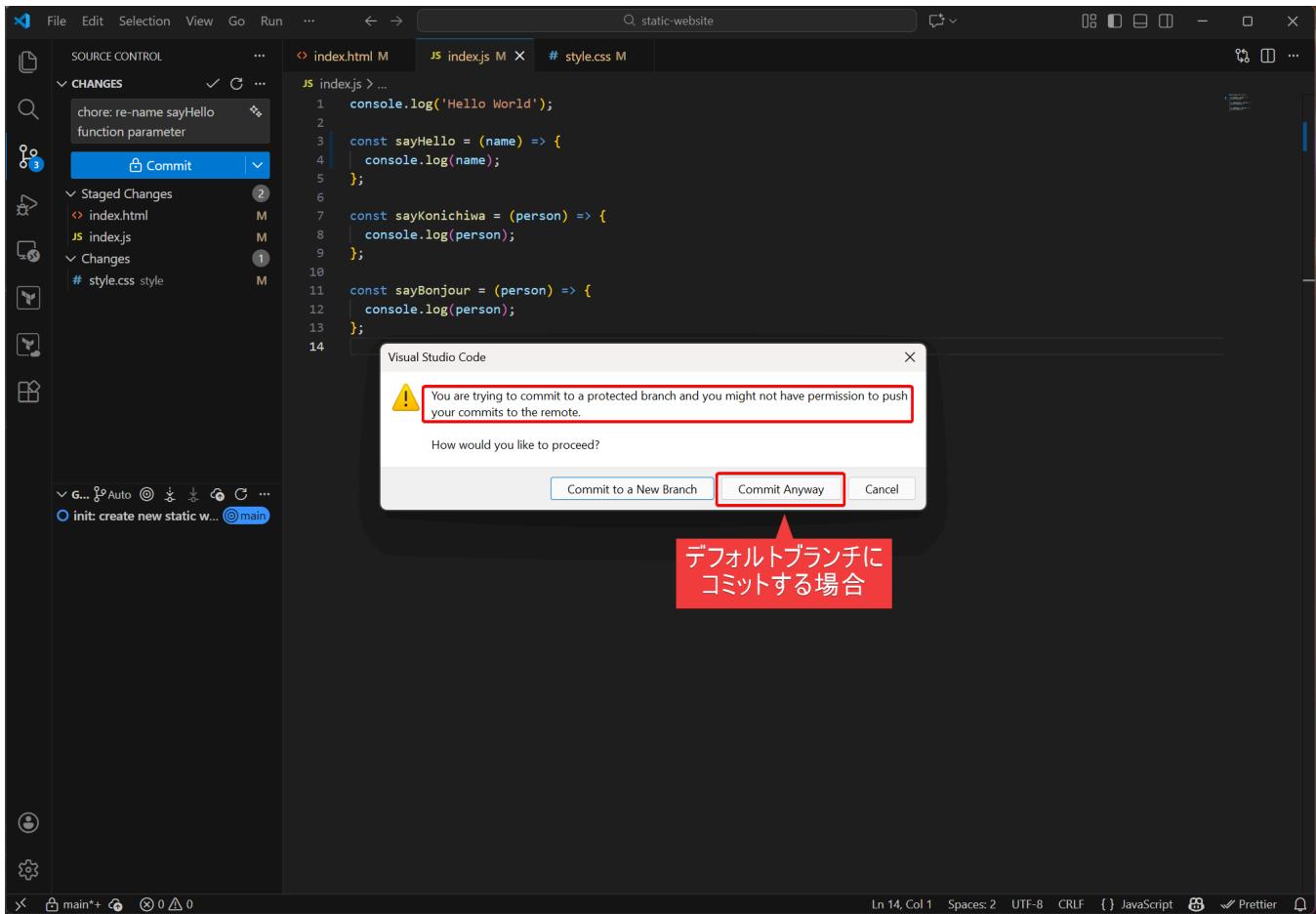
Configure Editors

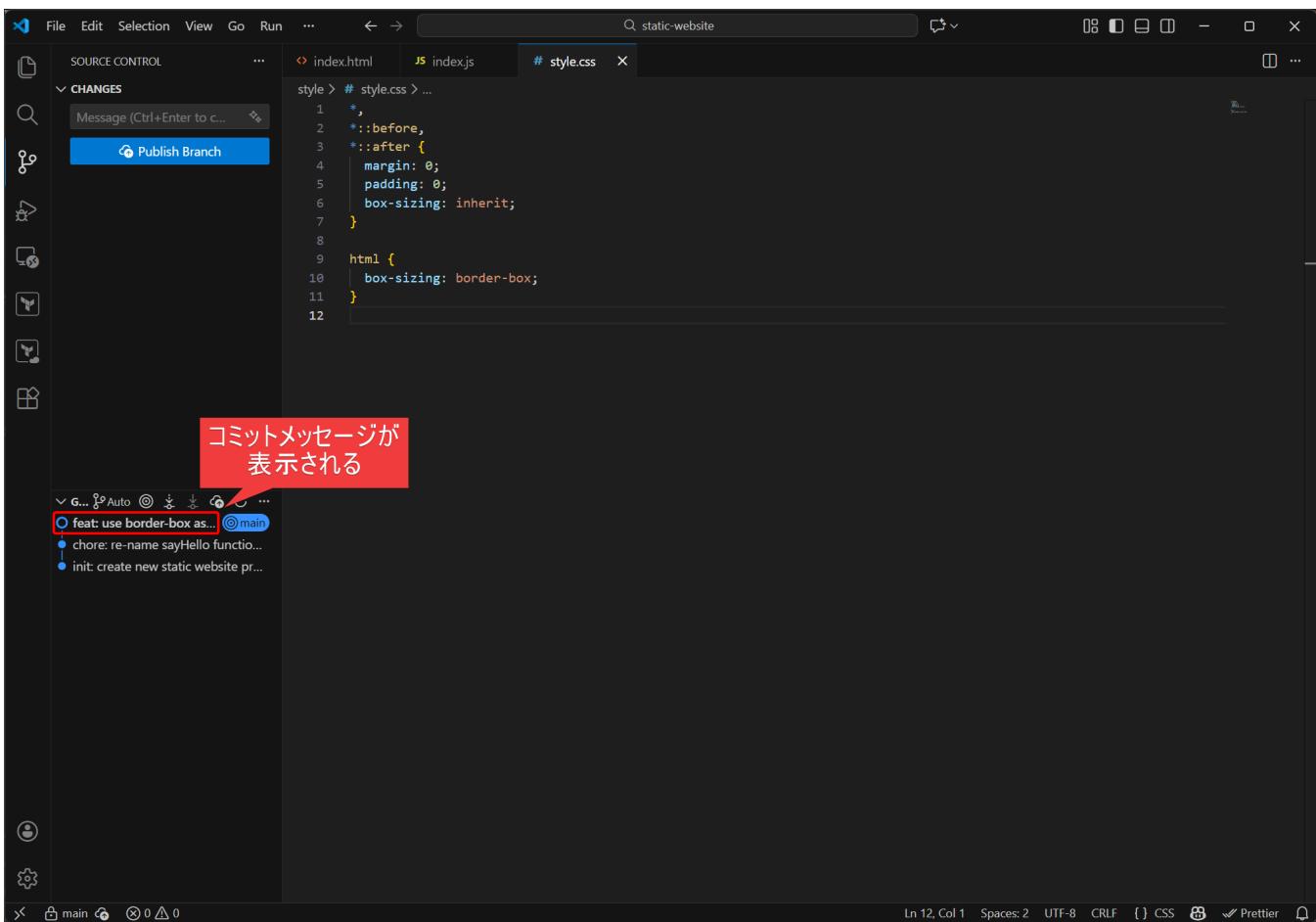
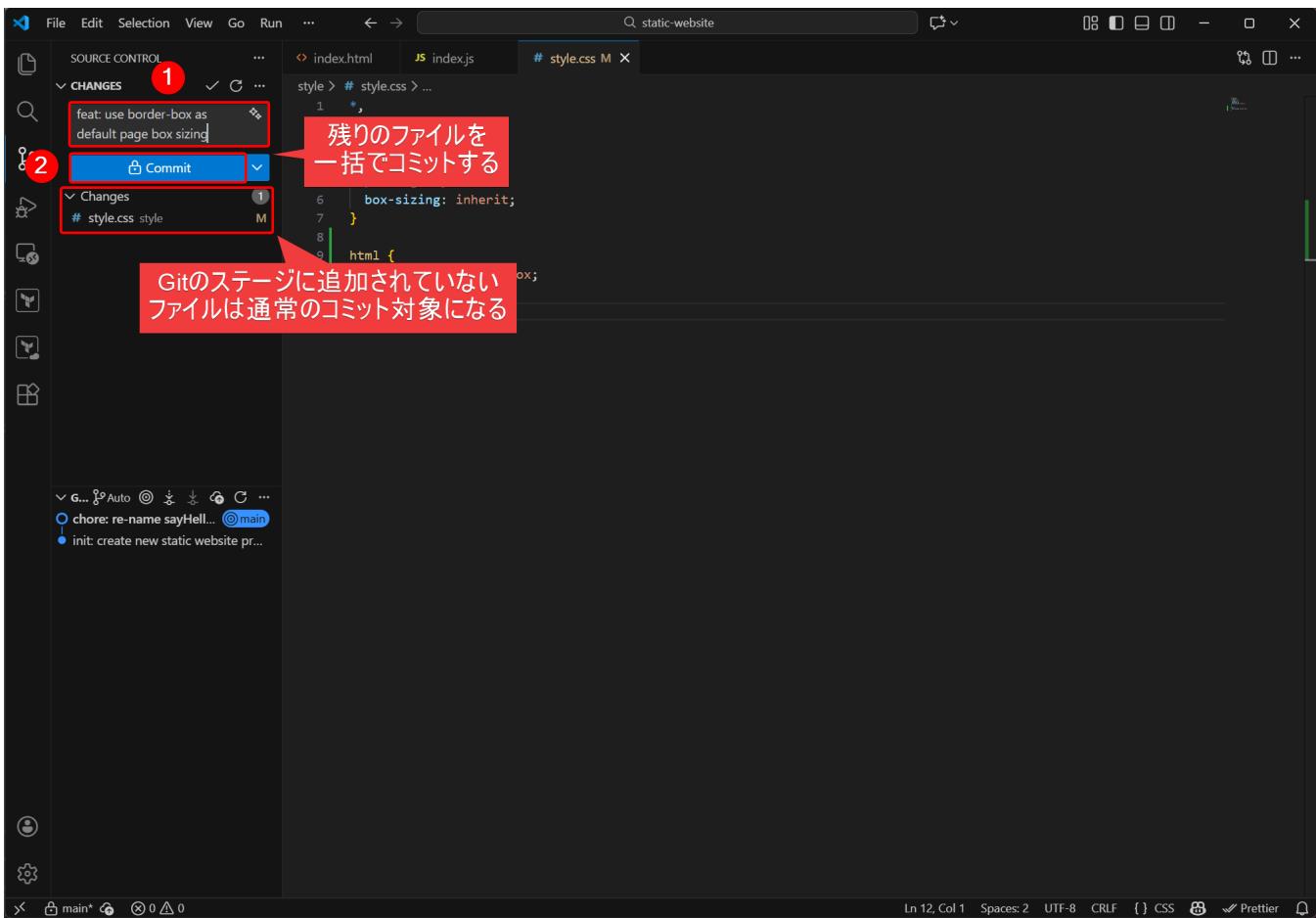
G... Auto ...

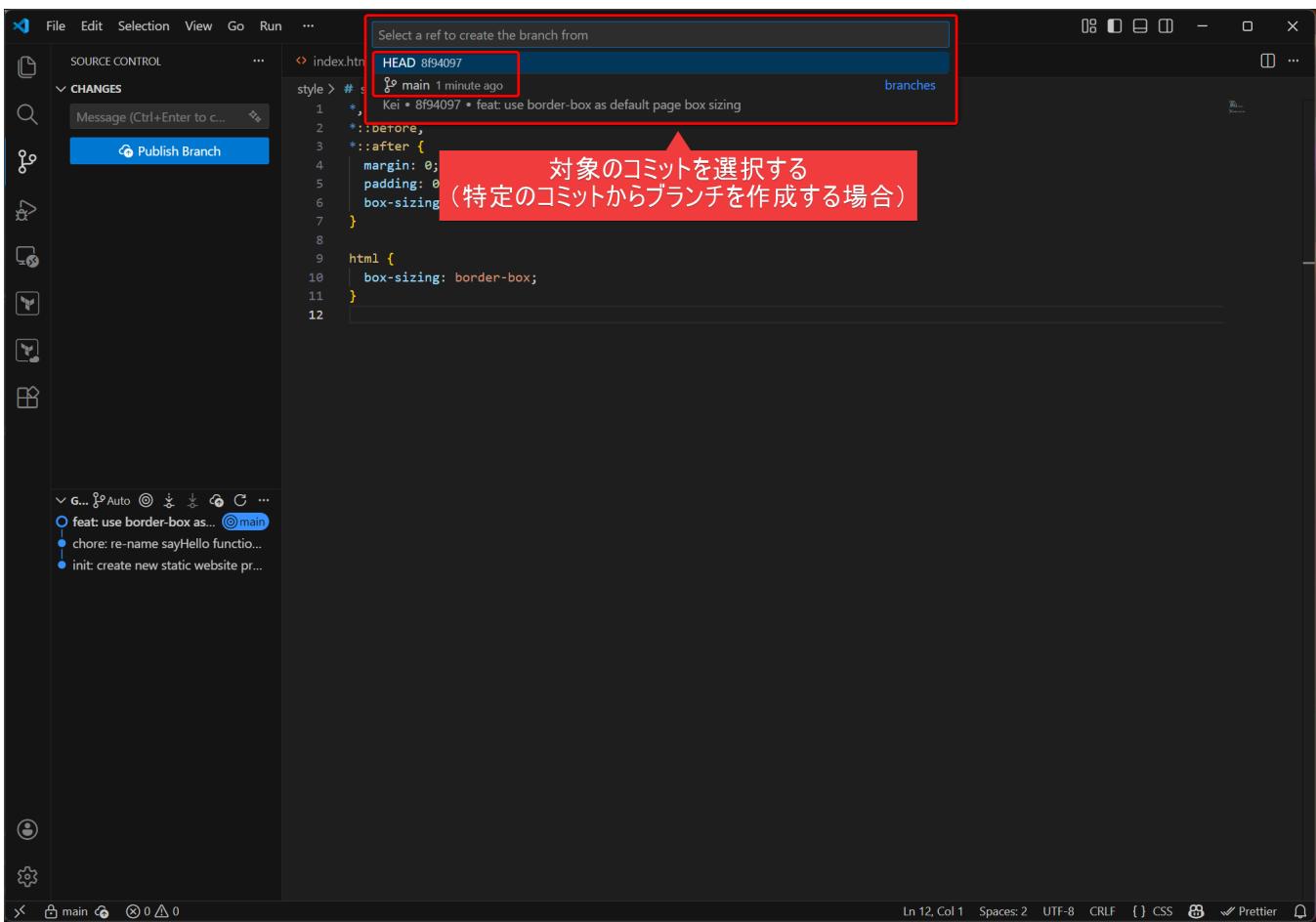
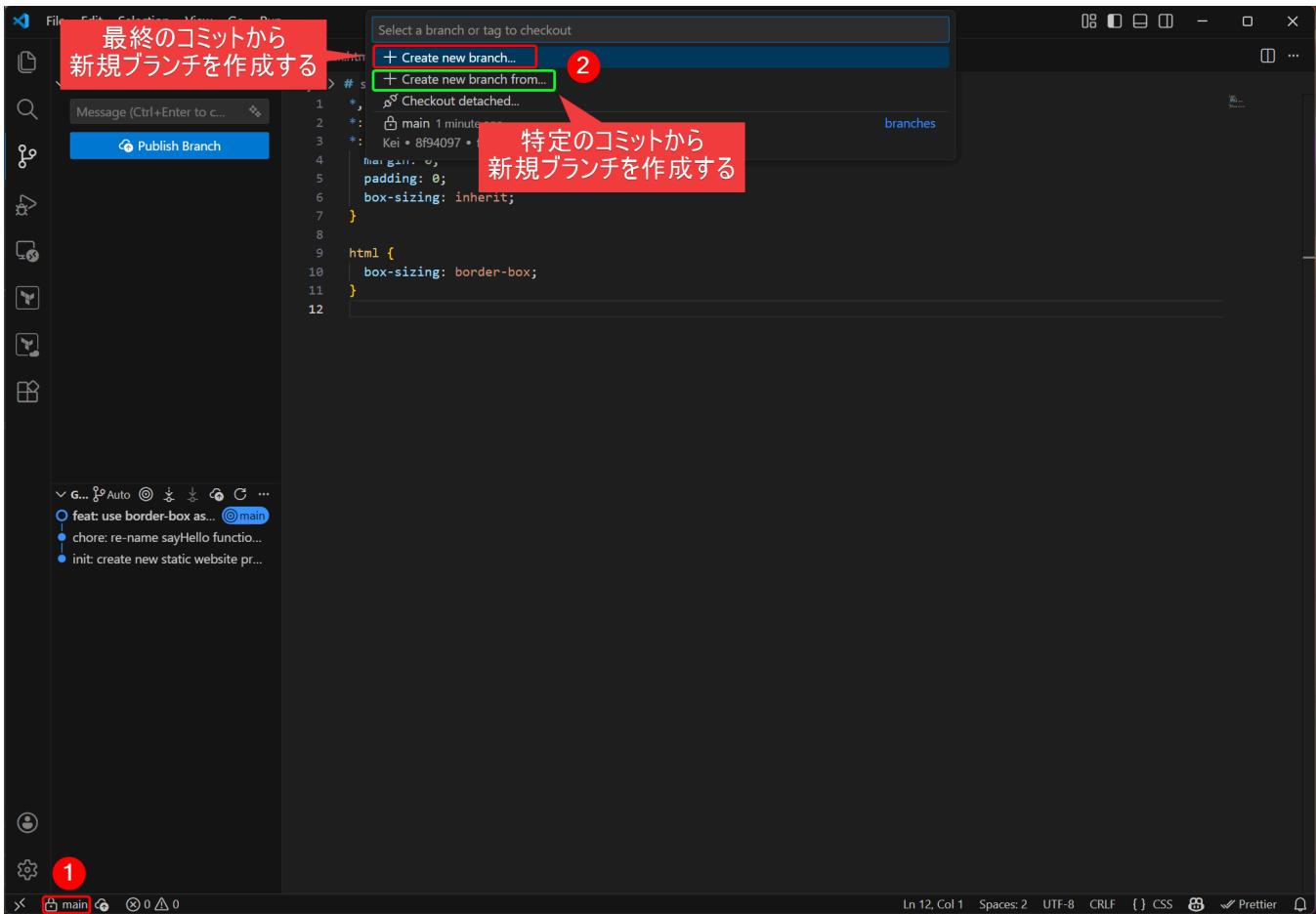
init: create new static w... @main

Ln 14, Col 1 Spaces: 2 UTF-8 CRLF { } JavaScript Prettier









The screenshot shows a code editor interface with a dark theme. At the top, there's a navigation bar with 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', and other icons. A search bar contains the text 'dev'. Below it, a message box says 'Please provide a new branch name (Press 'Enter' to confirm or 'Escape' to cancel)'. A red box highlights the search bar 'dev'. The main area shows a file named 'index.htm' with some CSS code. On the left, there's a sidebar titled 'CHANGES' with a list of commits, including 'feat: use border-box as ...' and 'chore: re-name sayHello functio...'. At the bottom, there's a status bar with file information and icons.

This screenshot shows the same code editor after a new branch has been created. The search bar now displays 'dev'. A red box highlights the search bar 'dev'. The sidebar 'CHANGES' now shows a commit for the new branch: 'feat: use border-box as ... @dev'. The status bar at the bottom also shows the branch name 'dev'.

The screenshot shows the VS Code interface with the 'SOURCE CONTROL' view open. A context menu is displayed over a list of branches, with the 'main' branch highlighted. A red callout box points to the 'main' branch with the Japanese text: '既存のブランチを選択するとそのブランチに切り替えられる' (Selecting an existing branch switches to that branch). A red circle labeled '2' is positioned near the top right of the dialog.

Select a branch or tag to checkout

- + Create new branch...
- + Create new branch from...
- Checkout detached...
- dev 2 minutes ago
- main 2 minutes ago
- Kei • 8f94097 • feat: use border-box as default page box sizing

branches

1

既存のブランチを選択すると
そのブランチに切り替えられる

2

Ln 12, Col 1 Spaces: 2 UTF-8 CRLF {} CSS ✓ Prettier

The screenshot shows the VS Code interface with the 'SOURCE CONTROL' view open. The status bar at the bottom indicates the current branch is 'main'. A red callout box points to the status bar with the Japanese text: 'ブランチの切り替え完了後' (After branch switch completion).

File Edit Selection View Go Run ...

SOURCE CONTROL ...

CHANGES

Message (Ctrl+Enter to c...)

Publish Branch

Ln 12, Col 1 Spaces: 2 UTF-8 CRLF {} CSS ✓ Prettier

1

Ln 12, Col 1 Spaces: 2 UTF-8 CRLF {} CSS ✓ Prettier

main

ブランチの切り替え完了後

