Introducing GitHub Copilot in Visual Studio Code

GitHub Copilot is an AI-powered coding assistant designed to help developers write code faster and more efficiently. Here is a simple step-by-step guide to introduce some of its features using Visual Studio Code (VS Code).

Step 1: What is GitHub Copilot?

Explanation: GitHub Copilot is an AI-powered tool that generates code suggestions directly in your editor. It helps with:

- Code completion
- Writing functions
- Learning new languages or frameworks

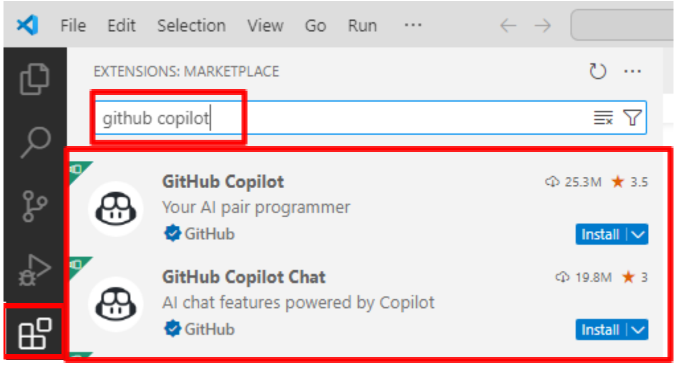
How It Works:

- It analyzes your current code and suggests lines or blocks of code based on context.

Step 2: Installing GitHub Copilot Extension

Steps:

1. Open VS Code.
2. Click on the Extensions icon in the Activity Bar or press `Ctrl+Shift+X` (Windows/Linux) or `Cmd+Shift+X` (Mac).
3. Search for **GitHub Copilot**.
4. Click **Install**.
   *(**Note:** When you Install Github Copilot, Github Copilot Chat will also be installed)*

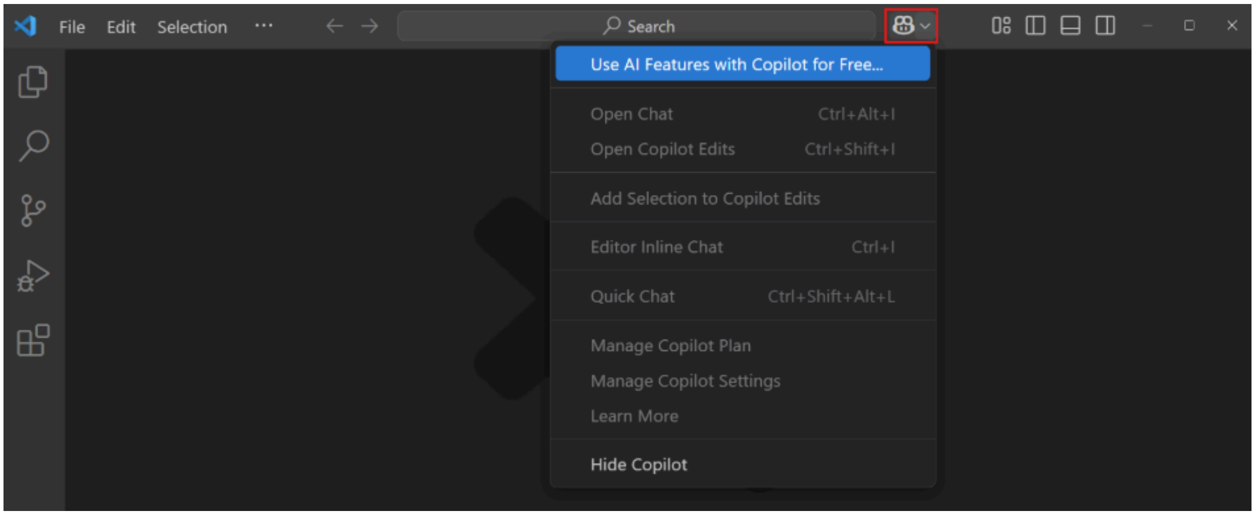


Explanation: This installs the extension needed for Copilot to function in VS Code.

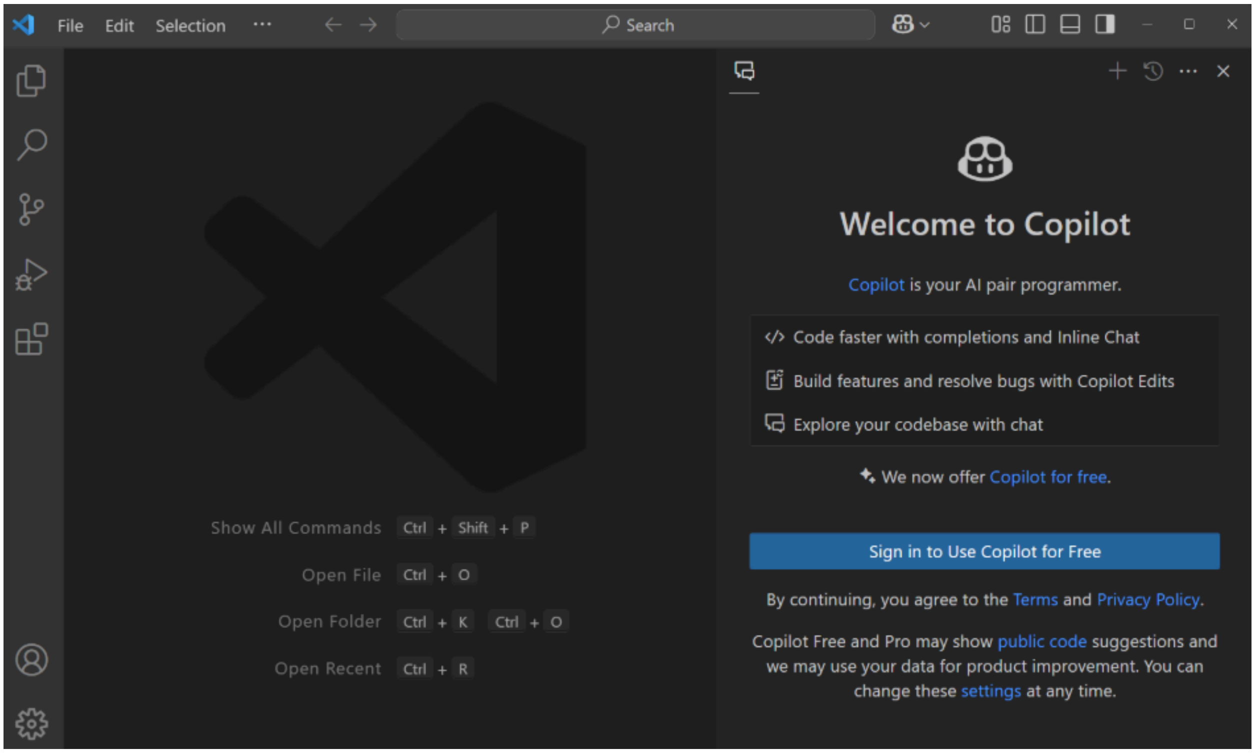
Step 3: Signing In and Activating Copilot

Steps:

1. After installation, you will be prompted to sign in to GitHub as shown in 2nd step. Else do as shown in below image.


File
Edit
Selection
Search
Use AI Features with Copilot for Free...
Open Chat Ctrl+Alt+I
Open Copilot Edits Ctrl+Shift+I
Add Selection to Copilot Edits
Editor Inline Chat Ctrl+I
Quick Chat Ctrl+Shift+Alt+L
Manage Copilot Plan
Manage Copilot Settings
Learn More
Hide Copilot

2. Click **Sign In** and authenticate with your GitHub credentials.


File
Edit
Selection
Search
Show All Commands Ctrl + Shift + P
Open File Ctrl + O
Open Folder Ctrl + K Ctrl + O
Open Recent Ctrl + R
Sign in to Use Copilot for Free

Welcome to Copilot

Copilot is your AI pair programmer.

Code faster with completions and Inline Chat

Build features and resolve bugs with Copilot Edits

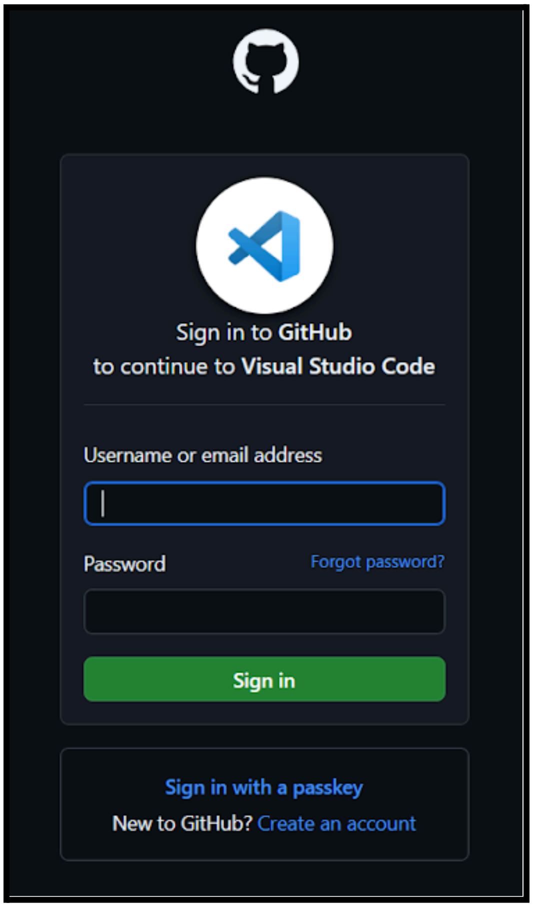
Explore your codebase with chat

We now offer Copilot for free.

By continuing, you agree to the Terms and Privacy Policy.

Copilot Free and Pro may show public code suggestions and we may use your data for product improvement. You can change these settings at any time.

3. Authorize GitHub Copilot for your account.

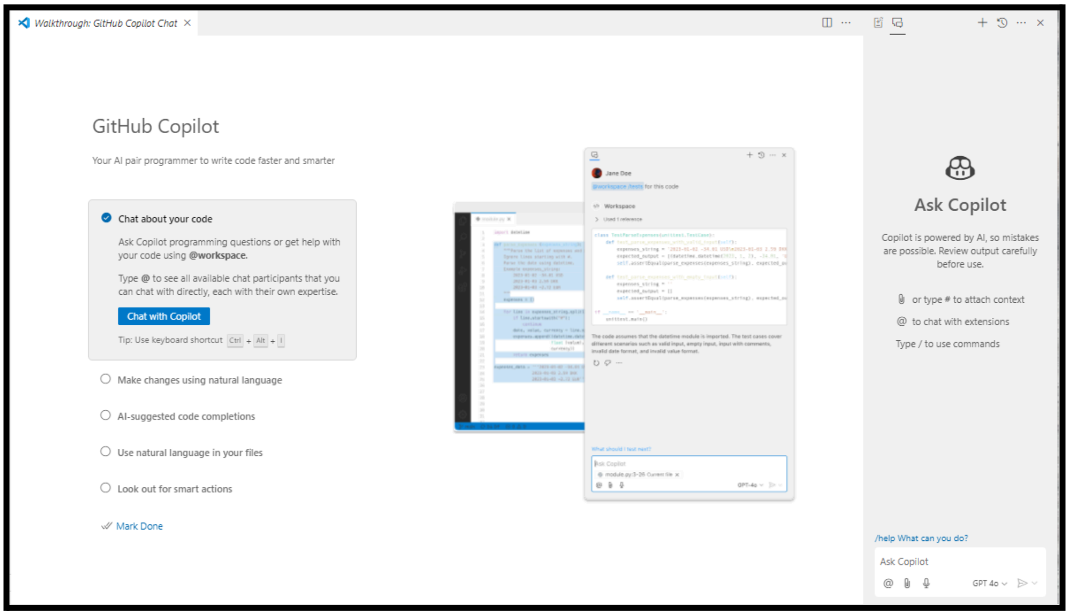


Sign in to **GitHub**
to continue to **Visual Studio Code**

Username or email address

Password

4. Return to VS Code to confirm activation as shown below.



GitHub Copilot

Your AI pair programmer to write code faster and smarter

● Chat about your code

Ask Copilot programming questions or get help with your code using **@workspace**.

Type @ to see all available chat participants that you can chat with directly, each with their own expertise.

Tip: Use keyboard shortcut Ctrl + Alt + I

○ Make changes using natural language

○ AI-suggested code completions

○ Use natural language in your files

○ Look out for smart actions

Ask Copilot

Copilot is powered by AI, so mistakes are possible. Review output carefully before use.

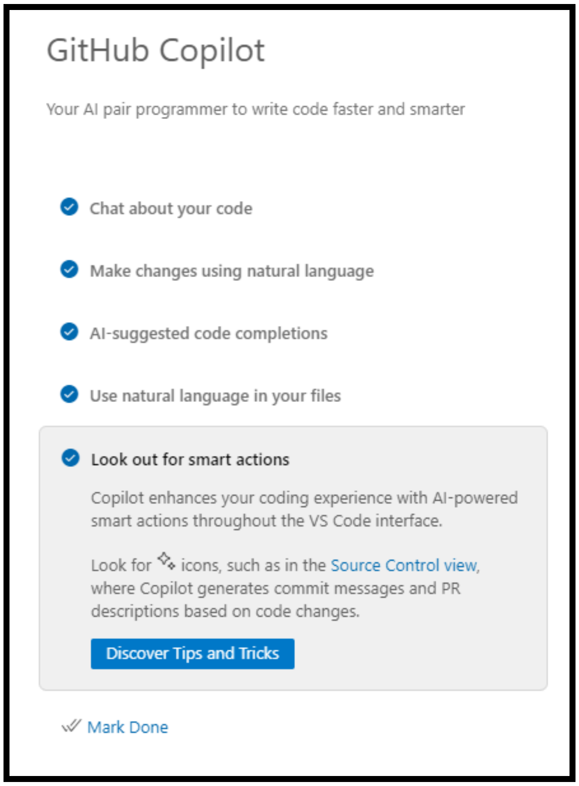
or type # to attach context

@ to chat with extensions

Type / to use commands

5. Select all the options and mark as done as shown below.

Explanation: This step connects GitHub Copilot to your account, allowing it to personalize suggestions based on your coding style.



GitHub Copilot

Your AI pair programmer to write code faster and smarter

✓ Chat about your code

✓ Make changes using natural language

✓ AI-suggested code completions

✓ Use natural language in your files

✓ Look out for smart actions

Copilot enhances your coding experience with AI-powered smart actions throughout the VS Code interface.

Look for ✨ icons, such as in the Source Control view, where Copilot generates commit messages and PR descriptions based on code changes.

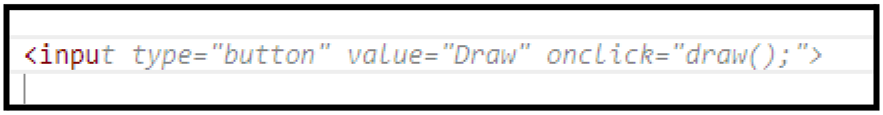
Common Features of Using Copilot

A. Code Suggestions

Steps:

1. Open any file in a programming language supported by Copilot (e.g., HTML, CSS, JavaScript,).
2. Start typing an html element, css selectors, comment, variable name or function.
3. Copilot will display a suggestion inline.
   - Press `Tab` to accept the suggestion.
   - Press `Esc` to dismiss it.

Explanation: Copilot provides context-aware suggestions to speed up your coding. For example, typing `<input` might generate an entire input element as shown below.

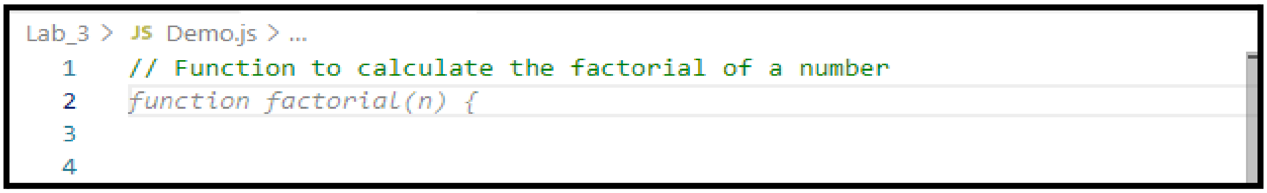


```
<input type="button" value="Draw" onclick="draw();">
```

B. Leveraging Natural Language Comments

Steps:

1. Add a comment describing what you want to achieve (e.g., `# Function to calculate factorial`).



```
// Function to calculate the factorial of a number
function factorial(n) {


```

(***Note:*** *If you do not see a suggestion as shown above, place your cursor at the end of the comment and press enter to generate a suggestion)*

2. The copilot will suggest code that fulfills the description.

Explanation: This feature allows you to generate code simply by describing your intention in natural language.

C.Navigating Multiple Suggestions

Steps:

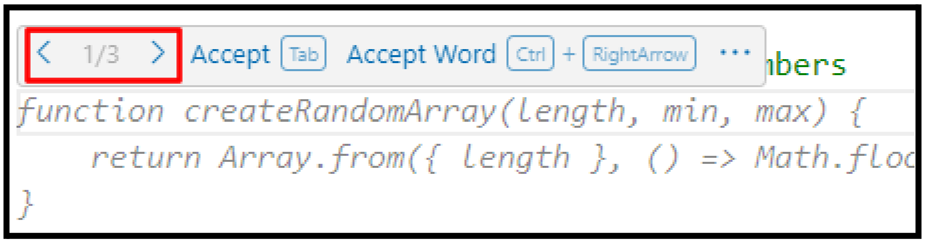
1. If Copilot generates multiple suggestions, use:

   Approach 1

   - `Alt+[` and `Alt+]` (Windows/Linux)
   - `Option+[` and `Option+]` (Mac) to cycle through them.

   Approach 2

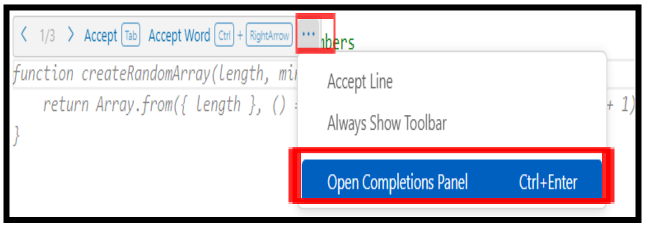
   - Hover your mouse over the suggestion and navigate with side bars.



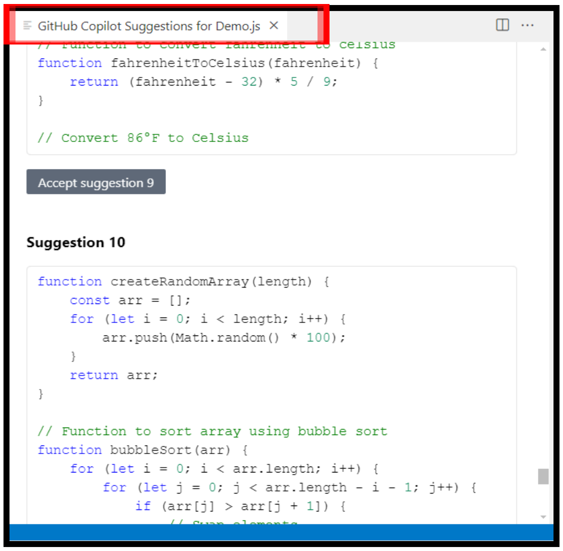
```
ıbers
function createRandomArray(length, min, max) {
    return Array.from({ length }, () => Math.floc
}
```

   Approach 3

   - See all suggestions as shown below and accept the most relevant suggestion to you.


1/3
Accept Tab
Accept Word Ctrl + RightArrow
···
Accept Line
Always Show Toolbar
Open Completions Panel Ctrl+Enter

```
ıbers
function createRandomArray(length, mi
    return Array.from({ length }, () 
}
```



```
// Function to convert fahrenheit to celsius
function fahrenheitToCelsius(fahrenheit) {
    return (fahrenheit - 32) * 5 / 9;
}

// Convert 86°F to Celsius
```

Suggestion 10

```
function createRandomArray(length) {
    const arr = [];
    for (let i = 0; i < length; i++) {
        arr.push(Math.random() * 100);
    }
    return arr;
}

// Function to sort array using bubble sort
function bubbleSort(arr) {
    for (let i = 0; i < arr.length; i++) {
        for (let j = 0; j < arr.length - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                // Swap elements
```

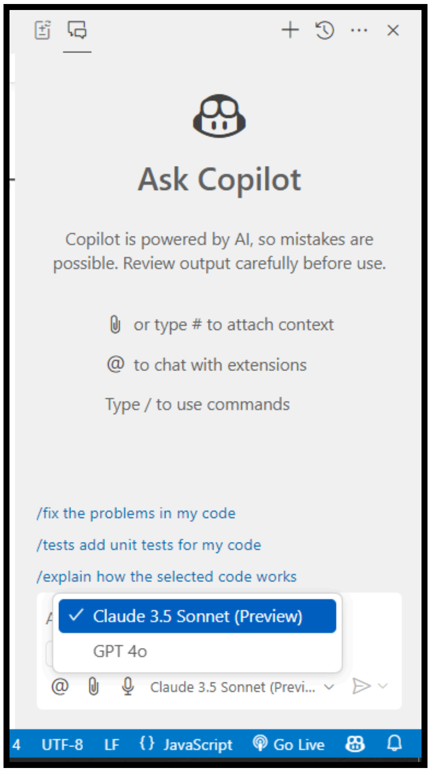
Explanation: This allows you to choose the most relevant suggestion from a list.

D. Chat with Copilot

Steps:

1. Choose the Model you want to chat.

   GPT-4 is typically faster for raw code generation, while Claude might shine in contexts where explanations or detailed discussions are needed.



Ask Copilot

Copilot is powered by AI, so mistakes are possible. Review output carefully before use.

or type # to attach context

@ to chat with extensions

Type / to use commands

/fix the problems in my code

/tests add unit tests for my code

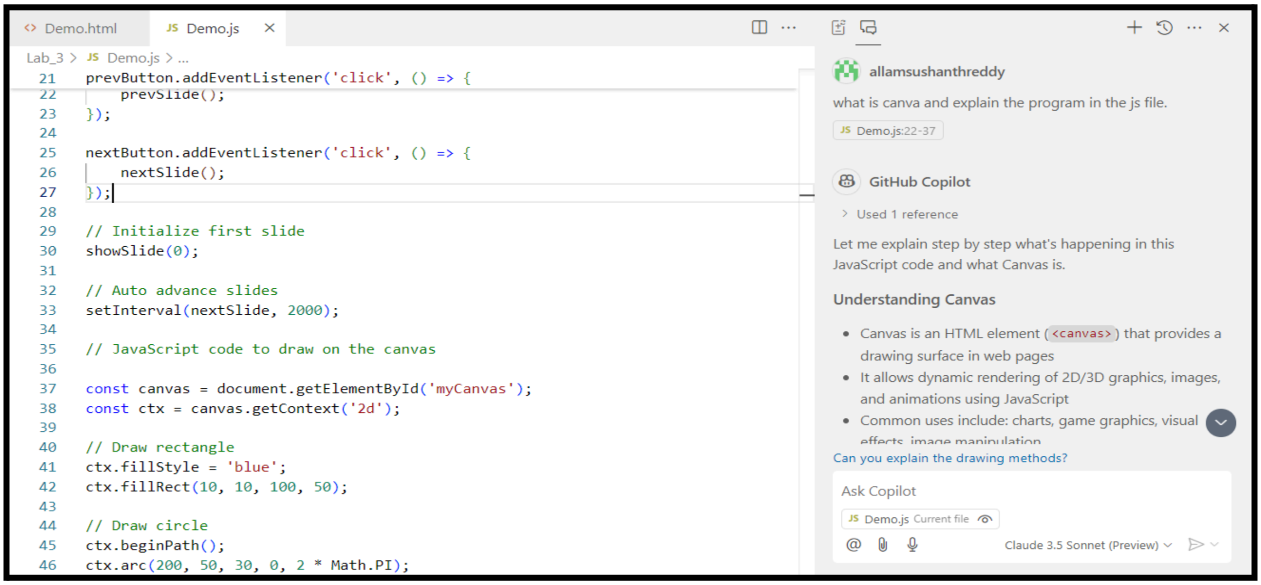
/explain how the selected code works

✓ Claude 3.5 Sonnet (Preview)

GPT 4o

Claude 3.5 Sonnet (Previ...

2. Ask your question
   (***Note:*** *If you want to ask a question related to a file, make sure the cursor is in that file and then ask your question.)*



```
prevButton.addEventListener('click', () => {
    prevSlide();
});

nextButton.addEventListener('click', () => {
    nextSlide();
});

// Initialize first slide
showSlide(0);

// Auto advance slides
setInterval(nextSlide, 2000);

// JavaScript code to draw on the canvas

const canvas = document.getElementById('myCanvas');
const ctx = canvas.getContext('2d');

// Draw rectangle
ctx.fillStyle = 'blue';
ctx.fillRect(10, 10, 100, 50);

// Draw circle
ctx.beginPath();
ctx.arc(200, 50, 30, 0, 2 * Math.PI);
```

allamsushanthreddy

what is canva and explain the program in the js file.

Demo.js:22-37

GitHub Copilot

> Used 1 reference

Let me explain step by step what's happening in this JavaScript code and what Canvas is.

Understanding Canvas

- Canvas is an HTML element (`<canvas>`) that provides a drawing surface in web pages
- It allows dynamic rendering of 2D/3D graphics, images, and animations using JavaScript
- Common uses include: charts, game graphics, visual effects, image manipulation

Can you explain the drawing methods?

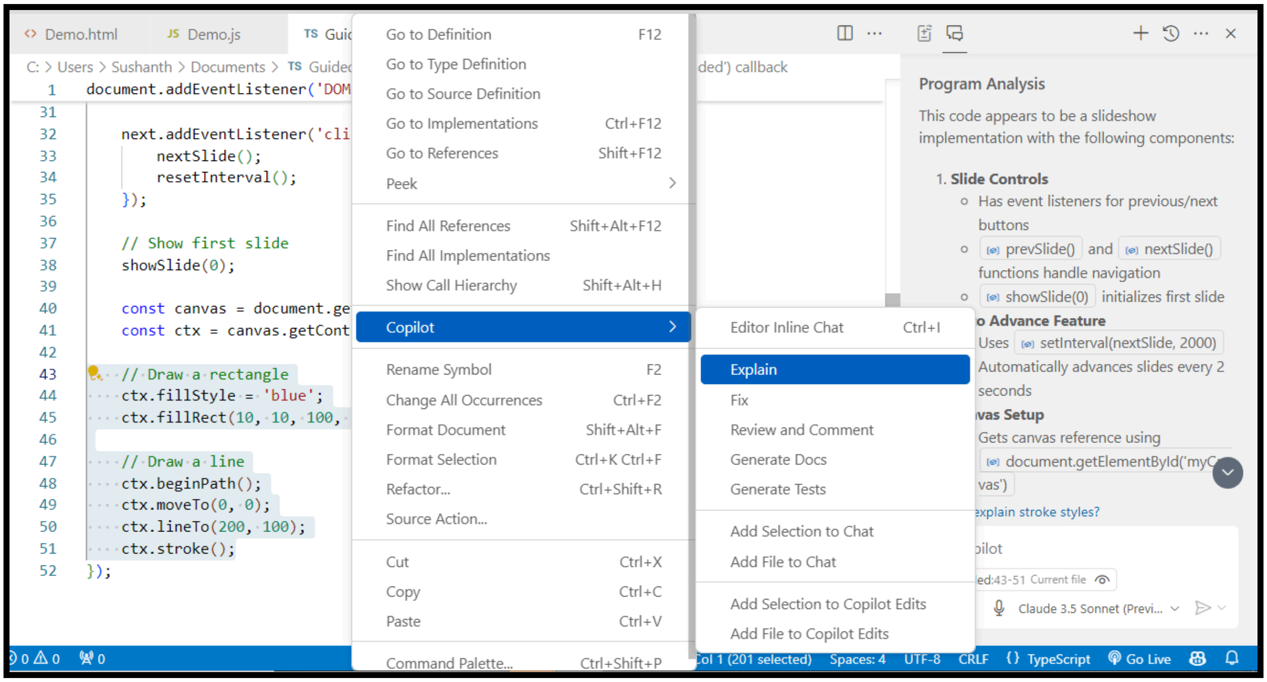
Ask Copilot

Demo.js Current file

Claude 3.5 Sonnet (Preview)

E. Code Explanation, Bug fixes and Reviews

1. Ask copilot to explain the code.
   Select the code > Right click > Copilot > Explain



Demo.html
Demo.js
Guic
C: > Users > Sushanth > Documents > Guidec
ded') callback
Go to Definition F12
Go to Type Definition
Go to Source Definition
Go to Implementations Ctrl+F12
Go to References Shift+F12
Peek
Find All References Shift+Alt+F12
Find All Implementations
Show Call Hierarchy Shift+Alt+H
Copilot
Rename Symbol F2
Change All Occurrences Ctrl+F2
Format Document Shift+Alt+F
Format Selection Ctrl+K Ctrl+F
Refactor... Ctrl+Shift+R
Source Action...
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Command Palette... Ctrl+Shift+P
Editor Inline Chat Ctrl+I
Explain
Fix
Review and Comment
Generate Docs
Generate Tests
Add Selection to Chat
Add File to Chat
Add Selection to Copilot Edits
Add File to Copilot Edits
0 ⚠ 0
0
ol 1 (201 selected) Spaces: 4 UTF-8 CRLF {} TypeScript Go Live
xplain stroke styles?
oilot
ed:43-51 Current file
Claude 3.5 Sonnet (Previ...

```
document.addEventListener('DOM

    next.addEventListener('cli
        nextSlide();
        resetInterval();
    });

    // Show first slide
    showSlide(0);

    const canvas = document.ge
    const ctx = canvas.getCont

    // Draw a rectangle
    ctx.fillStyle = 'blue';
    ctx.fillRect(10, 10, 100,

    // Draw a line
    ctx.beginPath();
    ctx.moveTo(0, 0);
    ctx.lineTo(200, 100);
    ctx.stroke();
});
```

Program Analysis

This code appears to be a slideshow implementation with the following components:

1. **Slide Controls**
   - Has event listeners for previous/next buttons
   - prevSlide() and nextSlide() functions handle navigation
   - showSlide(0) initializes first slide

o Advance Feature

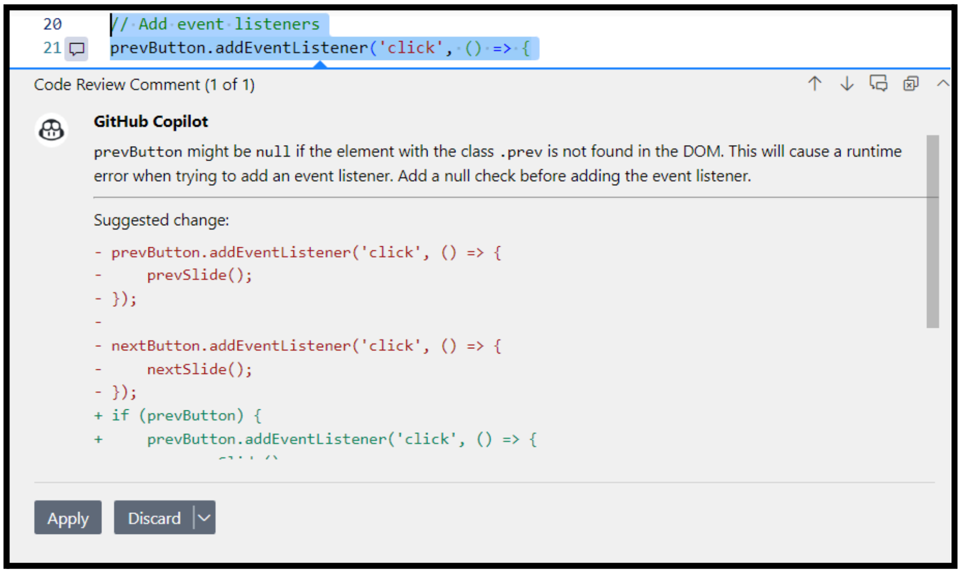
Uses setInterval(nextSlide, 2000)

Automatically advances slides every 2 seconds

vas Setup

Gets canvas reference using document.getElementById('myC vas')

2. Ask copilot to try and fix the issue.
   Select the code > Right click > Copilot > Fix
   *(**Note:** The fix might or might not solve the problem as per your logic. Be cautious while accepting the changes)*
3. Ask copilot to review and comment
   Select the code > Right click > Copilot > Review and Comment



```
// Add event listeners
prevButton.addEventListener('click', () => {
```

prevButton might be null if the element with the class .prev is not found in the DOM. This will cause a runtime error when trying to add an event listener. Add a null check before adding the event listener.

Suggested change:

```
- prevButton.addEventListener('click', () => {
-     prevSlide();
- });
-
- nextButton.addEventListener('click', () => {
-     nextSlide();
- });
+ if (prevButton) {
+     prevButton.addEventListener('click', () => {
```

This file is meant for personal use by javier.gusart@gmail.com only.
Sharing or publishing the contents in part or full is liable for legal action.

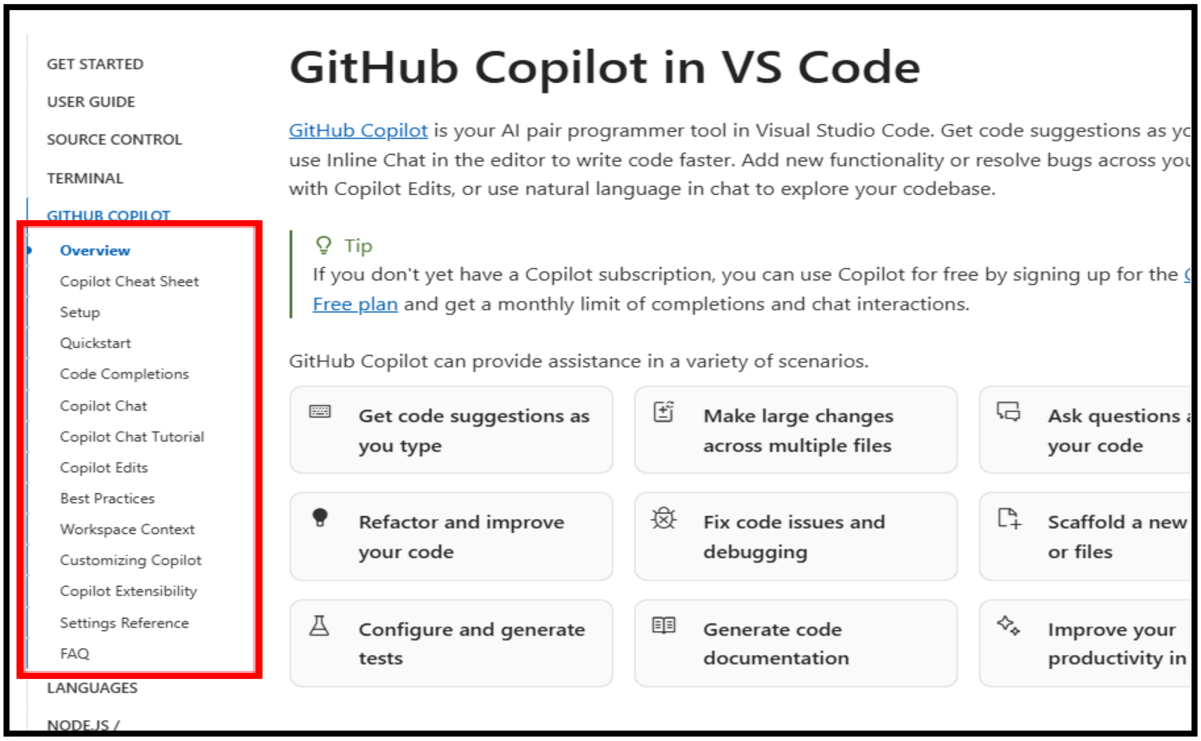
Best Practices for Using Copilot

Tips:

1. Always review AI-generated code for accuracy and security.
2. Use Copilot to learn new coding techniques by analyzing its suggestions.
3. If someone is new to a concept:
   - **Learn the basics first** through videos, tutorials, documentation, or hands-on practice.
   - Use tools like Copilot as a **supplementary aid** rather than the primary resource.
4. For experienced individuals:
   - Use Copilot to **speed up workflows**, but validate the generated code based on your expertise.
5. Combine Copilot with official documentation for a comprehensive development experience.

Explanation: While Copilot is a powerful tool, human oversight ensures quality and prevents errors.

If you want to learn more about copilot and all its features, find the official documentation here.



GitHub Copilot in VS Code

GitHub Copilot is your AI pair programmer tool in Visual Studio Code. Get code suggestions as yc use Inline Chat in the editor to write code faster. Add new functionality or resolve bugs across you with Copilot Edits, or use natural language in chat to explore your codebase.

Tip
If you don't yet have a Copilot subscription, you can use Copilot for free by signing up for the (Free plan and get a monthly limit of completions and chat interactions.

GitHub Copilot can provide assistance in a variety of scenarios.

Get code suggestions as you type
Make large changes across multiple files
Ask questions a your code
Refactor and improve your code
Fix code issues and debugging
Scaffold a new or files
Configure and generate tests
Generate code documentation
Improve your productivity in

---------------------- END -------------------------