



GitHub Copilotを VS Code で 使いこなす 💪

Shinya Yanagihara / 柳原伸弥

Developer Productivity GBB

Microsoft Corporation



Who am I ?



Shinya Yanagihara
shinyay

{ "position": "Developer Global Black Belt", "location": "Tokyo,JP", "favorites": ["Spring", "Kotlin", "Developer Productivity"] }

Edit profile

24 followers · 12 following

GitHub STAFF

Tokyo

19:23 (UTC +09:00)

shinyay@github.com

https://www.microsoft.com

@yanashin18618

@yanashin@mastodon.social

in/shinyay

shinyayREADME.md /



Java truly is an OOP language... As in: OOPs I used Java!

Shinya Yanagihara's GitHub Stats

☆ Total Stars Earned:	43
🕒 Total Commits (2024):	2.6k
🔗 Total PRs:	104
📄 Total Issues:	80
📅 Contributed to (last year):	1



Most Used Languages



views 6469 Twitter Profile

Career



Microsoft

Developer Productivity Specialist



Developer Advocate



App Modernization Specialist



Platform Architect

SNS



@shinyay



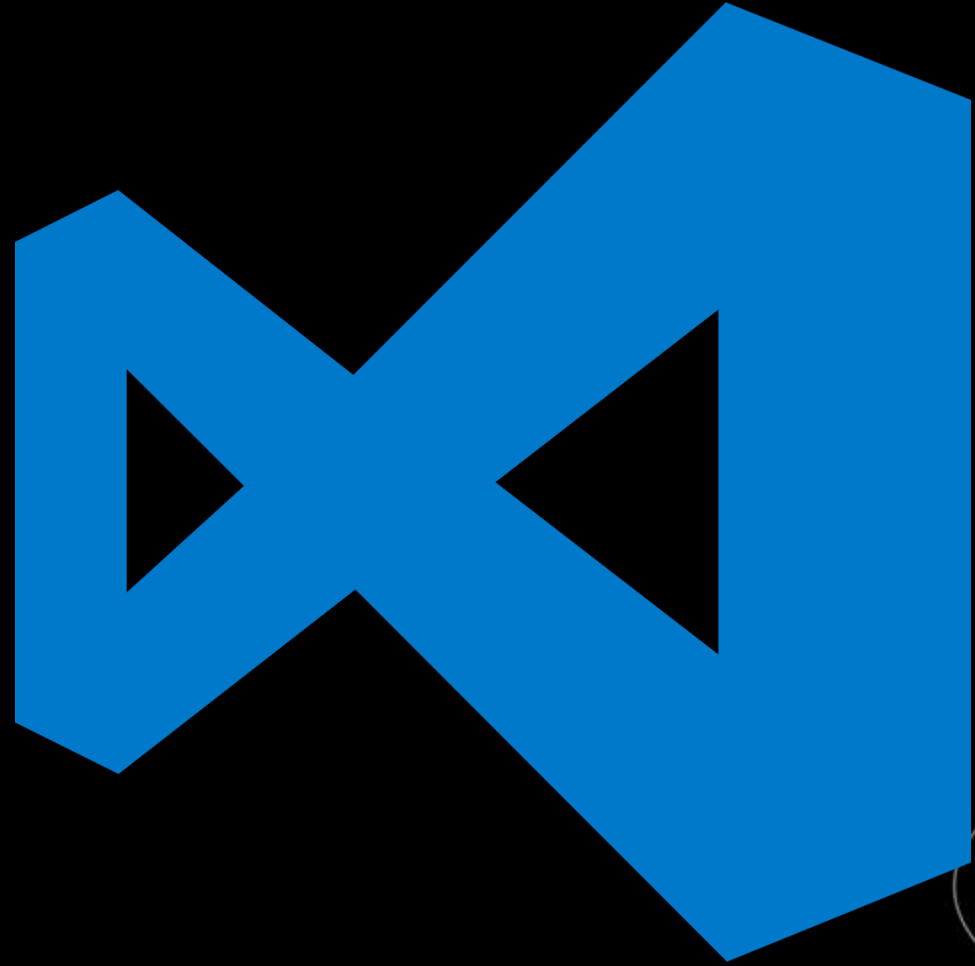
@yanashin18618



/in/shinyay







Version 1.95 is now available! Read about the new features and fixes from October.

Code faster with AI

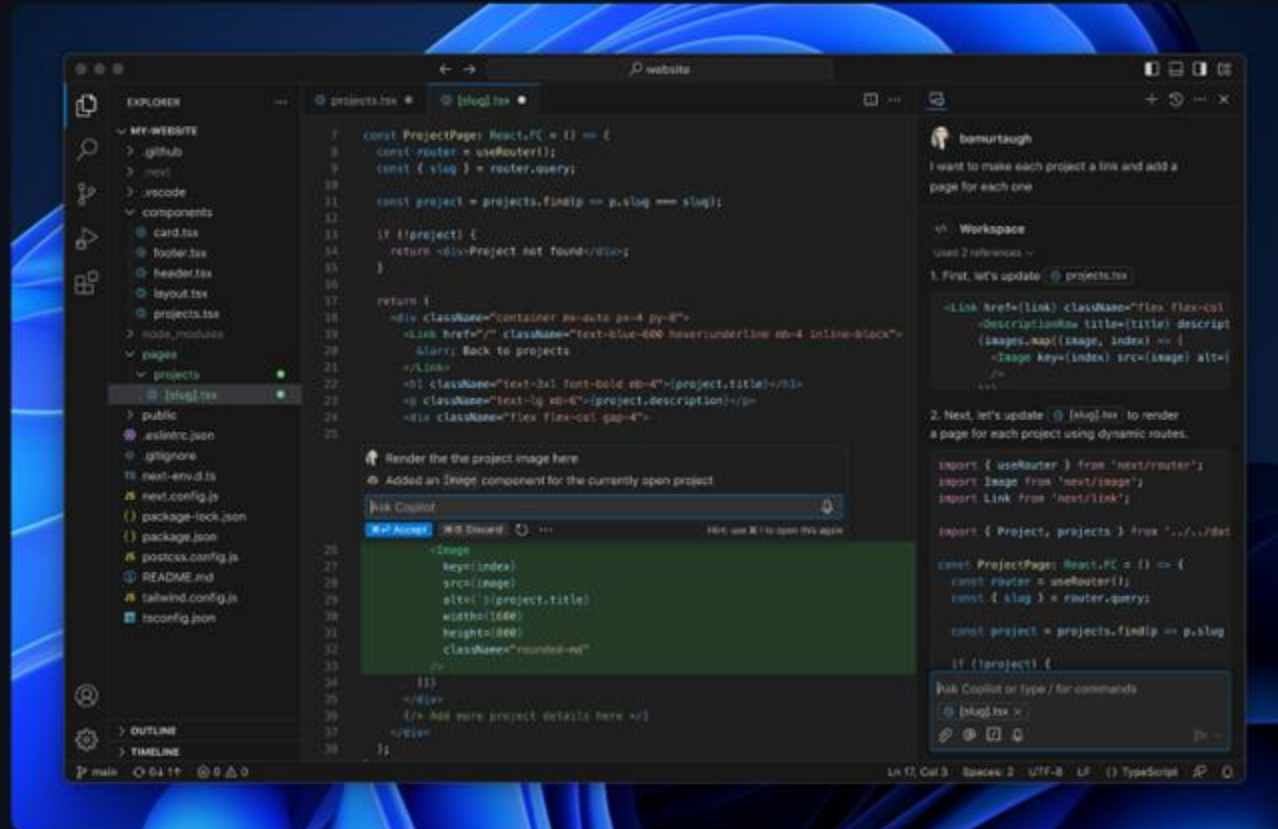
Visual Studio Code with GitHub Copilot supercharges your code with AI-powered suggestions, right in your editor.

Download for Windows

Try GitHub Copilot

Web, Insiders edition, or other platforms

Updates



UPDATES

June 2024 (version 1.91)

October 2024

September 2024

August 2024

July 2024

June 2024

May 2024

April 2024

March 2024

February 2024

January 2024

November 2023

October 2023

September 2023

August 2023

July 2023

Update 1.91.1: The update addresses these [issues](#).

Downloads: Windows: [x64 Arm64](#) | Mac: [Universal Intel silicon](#) | Linux: [deb rpm tarball Arm snap](#)

Welcome to the June 2024 release of Visual Studio Code. There are many updates in this version that we hope you'll like, some of the key highlights include:

- [Preview: Incoming/Outgoing changes graph](#) - Visualize incoming and outgoing changes in the Explorer view.
- [Python environments](#) - Enhanced environment discovery with Python environments.
- [Smart Send in native REPL](#) - Smoothly run code chunks in the native REPL.
- [GitHub Copilot extensibility](#) - Chat and Language Model APIs available for extension authors.
- [Preview: Profiles Editor](#) - Manage your profiles in a single place.
- [Custom tab labels](#) - More variable options and support for multiple tabs.
- [TypeScript 5.5](#) - Syntax checking for regular expressions and other improvements.
- [JavaScript Debugger](#) - Inspect shadowed variables while debugging.

UPDATES

October 2024

September 2024

August 2024

July 2024

June 2024

May 2024

April 2024

March 2024

February 2024

January 2024

November 2023

October 2023

September 2023

August 2023

July 2023

August 2024 (version 1.93)

Update 1.93.1: The update addresses these [issues](#).

Downloads: Windows: [x64 Arm64](#) | Mac: [Universal Intel silicon](#) | Linux: [deb rpm tarball Arm snap](#)

Welcome to the August 2024 release of Visual Studio Code. There are many updates in this version that we hope you'll like, some of the key highlights include:

- [Profiles editor](#) - Switch and manage your profiles from a single place.
- [Django unit test support](#) - Discover and run Django unit tests from the Test Explorer.
- [IntelliSense on vscode.dev](#) - Boost your JS & TS coding in vscode.dev with IntelliSense.
- [Notebook diff viewer](#) - Efficiently review changes in notebooks by collapsing unchanged cells.
- [Resize columns via the keyboard](#) - Resizing table columns in VS Code more accessible via keyboard.
- [Source Control Graph](#) - Easily hide, collapse, or move the Source Control Graph.
- [GitHub Copilot](#) - Add context in Quick Chat, improved test generation and chat history.
- [Experiment: Custom Copilot instructions](#) - Define specific code-generation instructions for Copilot.

UPDATES

October 2024

September 2024

August 2024

July 2024

June 2024

May 2024

April 2024

March 2024

February 2024

January 2024

November 2023

October 2023

September 2023

August 2023

July 2023

June 2023

May 2023

April 2023

March 2023

February 2023

January 2023

November 2022

September 2024 (version 1.94)

Update 1.94.1: The update addresses this security [issue](#).

Update 1.94.2: The update addresses these [issues](#).

Downloads: Windows: [x64 Arm64](#) | Mac: [Universal Intel silicon](#) | Linux: [deb rpm tarball Arm snap](#)

Welcome to the September 2024 release of Visual Studio Code. There are many updates in this version that we hope you'll like, some of the key highlights include:

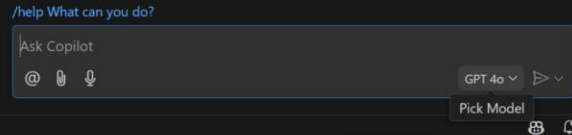
- [Find in Explorer](#) - Quickly find files in the Explorer view with the improved Find control.
- [Source Control Graph](#) - More filtering options and interactivity in the Source Control Graph.
- [Python test coverage](#) - Run Python tests with coverage and get rich results in the editor.
- [ESM](#) - Faster VS Code startup thanks to the migration to ESM.
- [Account preference](#) - Specify which account to use for an extension.
- [Copilot in Native REPL](#) - Get code completions and Inline Chat in the Native REPL.
- [Improved chat context](#) - Drag & drop files or use IntelliSense for more relevant chat context.
- [Test environment setup](#) - Get help with setting up a test framework for your workspace.

If you'd like to read these release notes online, go to [Updates](#) on [code.visualstudio.com](#). Insiders: Want to try new features as soon as possible? You can download the nightly [Insiders](#) build and try the latest updates as soon as they are available.

GitHub Copilot

Switch language models in chat

Previously, we announced that you can [sign up for early access to OpenAI o1 models](#). Once you have access, you will have a Copilot Chat model picker control in Copilot Chat in VS Code to choose which model version to use for your chat conversations.



UPDATES

October 2024

September 2024

August 2024

July 2024

June 2024

May 2024

April 2024

March 2024

February 2024

January 2024

November 2023

October 2023

September 2023

August 2023

July 2023

June 2023

May 2023

April 2023

March 2023

February 2023

January 2023

November 2022

October 2024 (version 1.95)

Edit

Security update: The following extensions have security updates: [ms-python.python](#) and [ms-vscode-remote.remote-ssh](#).

Update 1.95.1: The update addresses these [issues](#).

Update 1.95.2: The update addresses these [issues](#).

Update 1.95.3: The update addresses these [issues](#).

Downloads: Windows: [x64 Arm64](#) | Mac: [Universal Intel silicon](#) | Linux: [deb rpm tarball Arm snap](#)

Welcome to the October 2024 release of Visual Studio Code. There are many updates in this version that we hope you'll like, some of the key highlights include:

- [Copilot Edits](#) - Iterate quickly on large changes across multiple files
- [Copilot Chat in Secondary Side Bar](#) - Keep Copilot Chat open and ready to go while you work
- [Multiple GitHub accounts](#) - Log in to multiple GitHub accounts in VS Code simultaneously
- [Copilot code reviews](#) - Get a quick review pass or a deeper review of uncommitted changes
- [Docstrings with Pylance](#) - Generate docstring templates for classes or methods
- [Preview settings indicator](#) - View experimental and preview settings in the Settings editor
- [Copilot extensibility](#) - Showcasing Copilot extensibility in VS Code

If you'd like to read these release notes online, go to [Updates](#) on [code.visualstudio.com](#). Insiders: Want to try new features as soon as possible? You can download the nightly [Insiders](#) build and try the latest updates as soon as they are available.

GitHub Copilot

Copilot features might go through different early access stages, which are typically enabled and configured through settings.

- **Experimental** - view the [experimental features](#) (@tag:experimental)

This setting controls a new feature that is actively being developed and may be unstable. It is subject to change or removal.

- **Preview** - view the [preview features](#) (@tag:preview)

This setting controls a new feature that is still under refinement yet ready to use. Feedback is welcome.



UPDATES

July 2024 (version 1.92)

October 2024

September 2024

August 2024

July 2024

June 2024

May 2024

April 2024

March 2024

February 2024

January 2024

November 2023

October 2023

September 2023

August 2023

July 2023

Update 1.92.1: The update addresses these [issues](#).

Update 1.92.2: The update addresses these [issues](#).

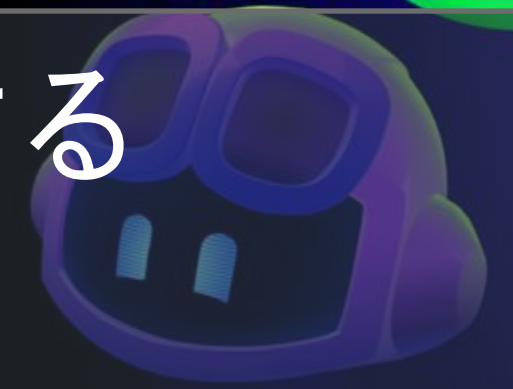
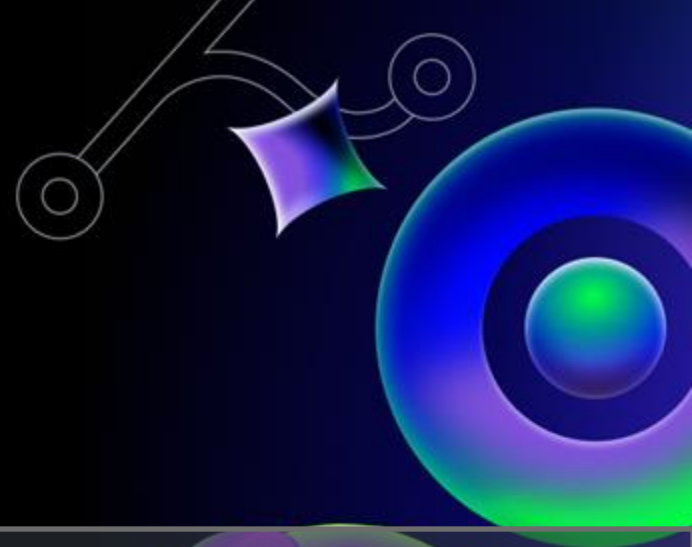
Downloads: Windows: [x64 Arm64](#) | Mac: [Universal Intel silicon](#) | Linux: [deb rpm tarball Arm snap](#)

Welcome to the July 2024 release of Visual Studio Code. There are many updates in this version that we hope you'll like, some of the key highlights include:

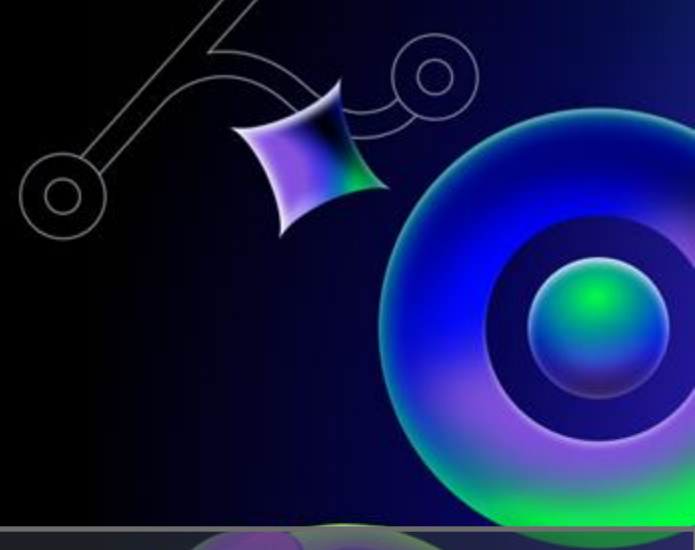
- [Default browser](#) - Configure which browser to use for opening links in VS Code.
- [Revert PRs](#) - Easily create a revert PR for a merged PR.
- [Extension updates](#) - More easily configure auto updating of extensions.
- [Override profiles](#) - Override an existing profile with the Profiles Editor.
- [Paste files in CSS](#) - Quickly add CSS file references with paste or drag and drop.
- [Move Panel to top](#) - Position the Panel section at the top of the workbench.
- [Copilot uses GPT-4o](#) - [GitHub Copilot](#) Chat upgraded to OpenAI's GPT-4o.

If you'd like to read these release notes online, go to [Updates](#) on [code.visualstudio.com](#). Insiders: Want to try new features as soon as possible? You can download the nightly [Insiders](#) build and try the latest updates as soon as they are available.

VS Code の利用を便利にする GitHub Copilot の機能



October 2024
(version 1.95)



Chat in the Secondary Side Bar

The image shows a screenshot of the Visual Studio Code (VS Code) interface. On the left, a code editor displays Python code for a unit test. On the right, the Explorer sidebar shows a project structure. In the bottom right corner, the GitHub Copilot chat interface is open in the secondary sidebar, showing a conversation with the AI. A pink arrow points from the chat interface to the secondary sidebar area.

VS Code の右側に位置するセカンダリサイドバーが、GitHub Copilot チャットビューのデフォルトの表示場所

GitHub Copilot チャットを開いたままエクスプローラやソース管理などその他のビューを利用可能

Copilot Edits

Commonly Used

- > Text Editor
- > Workbench
- > Window
- > Features
- > Application
- > Security
- ▼ Extensions
 - .ipynb Support
 - .NET Install Tool
 - Azure Repos
 - Copilot Next Edit Sug...
 - > CSS Language Features
 - Dev Containers
 - Emmet
 - Extension Pack for Java
 - Git
 - GitHub
 - GitHub Actions
 - GitHub Codespaces
 - GitHub Copilot
 - ▼ GitHub Copilot Chat
 - Preview**
 - Experimental
 - GitHub Copilot for Az...
 - GitHub Copilot Works...
 - GitHub Enterprise Ser...
 - GitHub Pull Requests
 - GitHub Repositories
 - Gradle

Extensions

GitHub Copilot Chat

Preview

Github > Copilot > Chat > Code Generation: Use Instruction Files [Preview](#)

Controls whether code instructions from `.github/copilot-instructions.md` are added to Copilot requests.

Note: Keep your instructions short and precise. Poor instructions can degrade Copilot's quality and performance.

Github > Copilot > Chat > Edits: Enabled [Preview](#)

Whether to enable the Copilot Edits feature.

Github > Copilot > Chat > Review Selection: Enabled [Preview](#)

Enables code review on current selection.

Github > Copilot > Chat > Review Selection: Instructions [Preview](#)

A set of instructions that will be added to Copilot requests that provide code review for the current selection. Instructions can come from:

- a file in the workspace: `{ "file": "fileName" }`
- text in natural language: `{ "text": "Use underscore for field names." }`

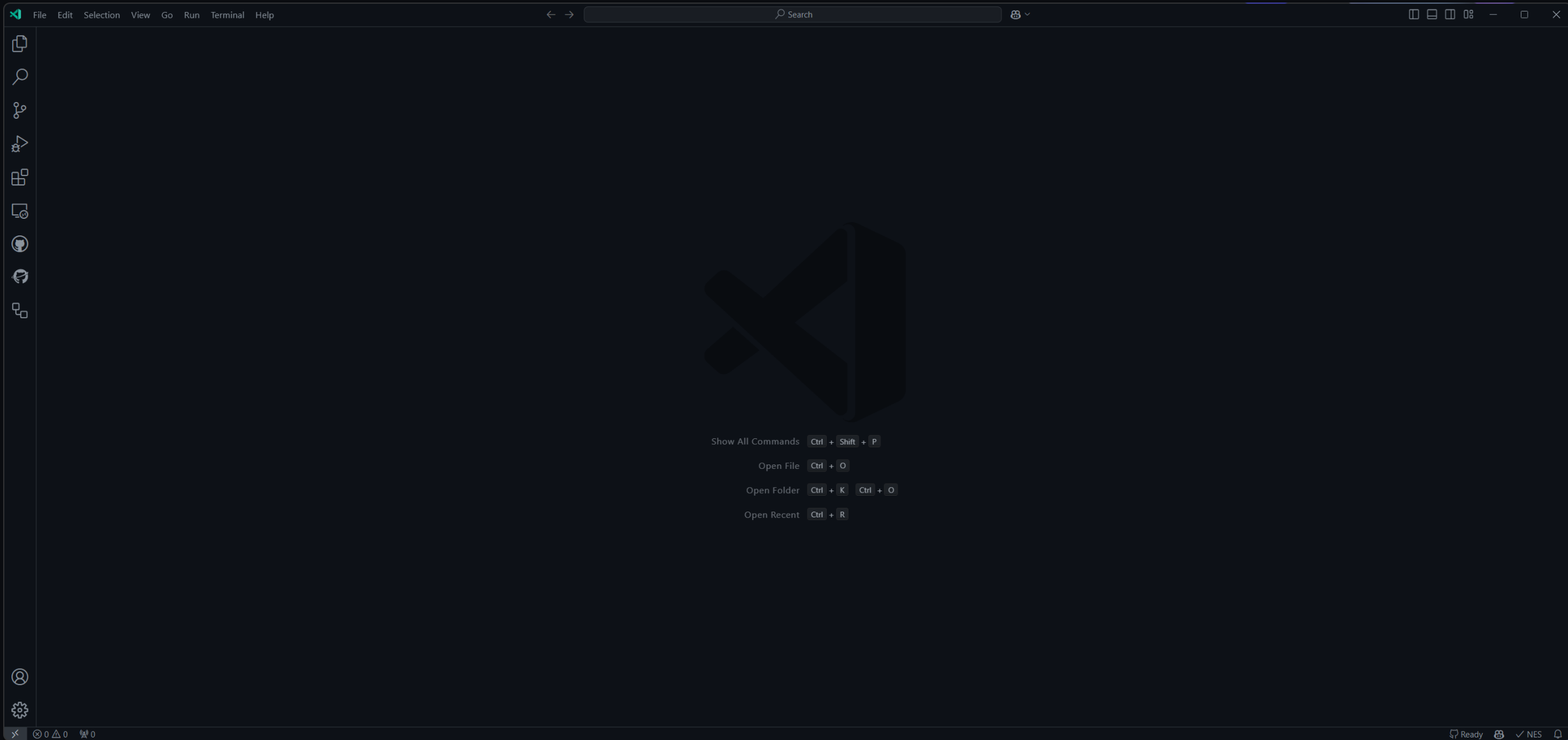
Note: Keep your instructions short and precise. Poor instructions can degrade Copilot's effectiveness.

[Edit in settings.json](#)

複数のファイルに跨って、
GitHub Copilot が
コードの変更や生成を実施



デモンストレーション



Copilot code reviews

The screenshot displays the Visual Studio Code interface with a code review window open. The code editor shows the following Java code for `Owner.java`:

```
48 public class Owner extends Person {  
59     @Column(name = "telephone")  
60     @NotBlank  
61     @Pattern(regexp = "\\d{10}", message = "Telephone must be a 10-digit number")  
62     private String telephone;  
63  
64     @Column(name = "zipcode")  
65     @NotBlank
```

The code review comment from GitHub Copilot is:

Code Review Comment (1 of 1)

GitHub Copilot

The 'zipcode' field should have a pattern validation to ensure it meets expected formats, such as a 5-digit or 9-digit ZIP code.

Suggested change:

```
+     @Pattern(regexp = "\\d{5}(-\\d{4})?", message = "Zipcode must be a 5-digit or 9-digit number")
```

Buttons: **Apply** | **Discard** | **▼**

The interface also shows the Source Control panel on the left with a 'Commit' button and a 'Copilot Code Review - Uncommitted Changes' section. A search bar at the top right contains the text 'zip'.



Automatic chat participant detection

チャット参加者の例

@azure
@docker
@github
@hubber-lookup
@issue-matcher-agent
@teamsagent
@teamsapp
@terminal
@vscode
@workspace
@azure /changeTenant
@azure /cost

チャット参加者を現在行われているチャットの
コンテキストから判断して自動で追加

@workspace

ワークスペース内のコードに関するコンテキストを集められるため、コードに関するナビゲーションや、関連するクラスやファイルなどの情報を確認するサポートをしてくれる

@github

GitHubリポジトリやコミット、Issue、プルリクエストなどについて情報を確認するサポートをしてくれる



shinyay

データの登録操作はどこで行われていますか？

`</>` **Workspace** (rerun without)

> Used 40 references

✓ Collecting workspace information

データの登録操作は、以下のファイルで行われています。

1. `Owner` の登録操作は、`OwnerController` クラスの `processCreationForm` メソッドで行われています。
2. `Pet` の登録操作は、`PetController` クラスの `processCreationForm` メソッドで行われています。
3. `Visit` の登録操作は、`VisitController` クラスの `processNewVisitForm` メソッドで行われています。



shinyay

直近のコミットログを教えてください。

`GitHub` (rerun without)

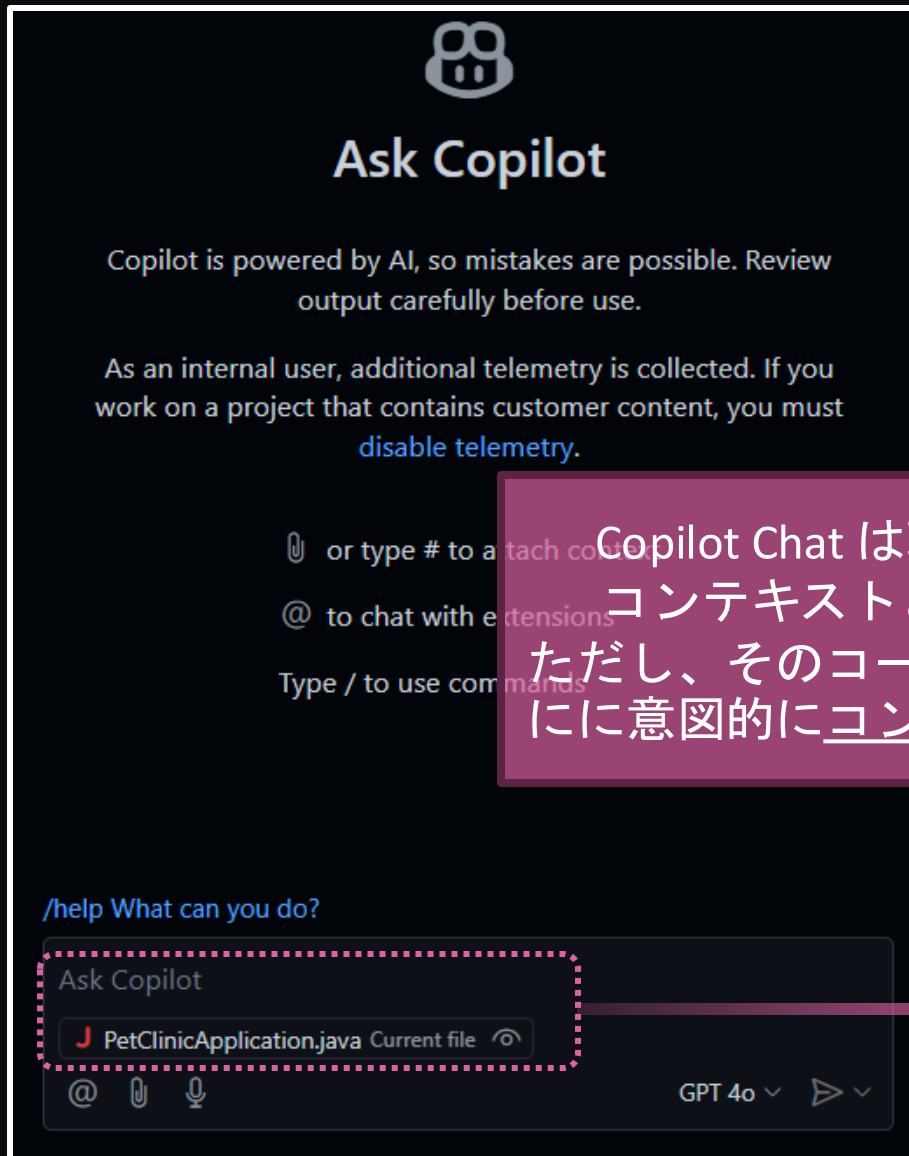
> Used 2 references


直近のコミットログは以下の通りです：

1. `Modify` - shinyay
2. `Initial Commit` - shinyay





Control current editor context




Ask Copilot



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






As an internal user, additional telemetry is collected. If you work on a project that contains customer content, you must [disable telemetry](#).

 or type # to attach code
 to chat with extensions
Type / to use commands

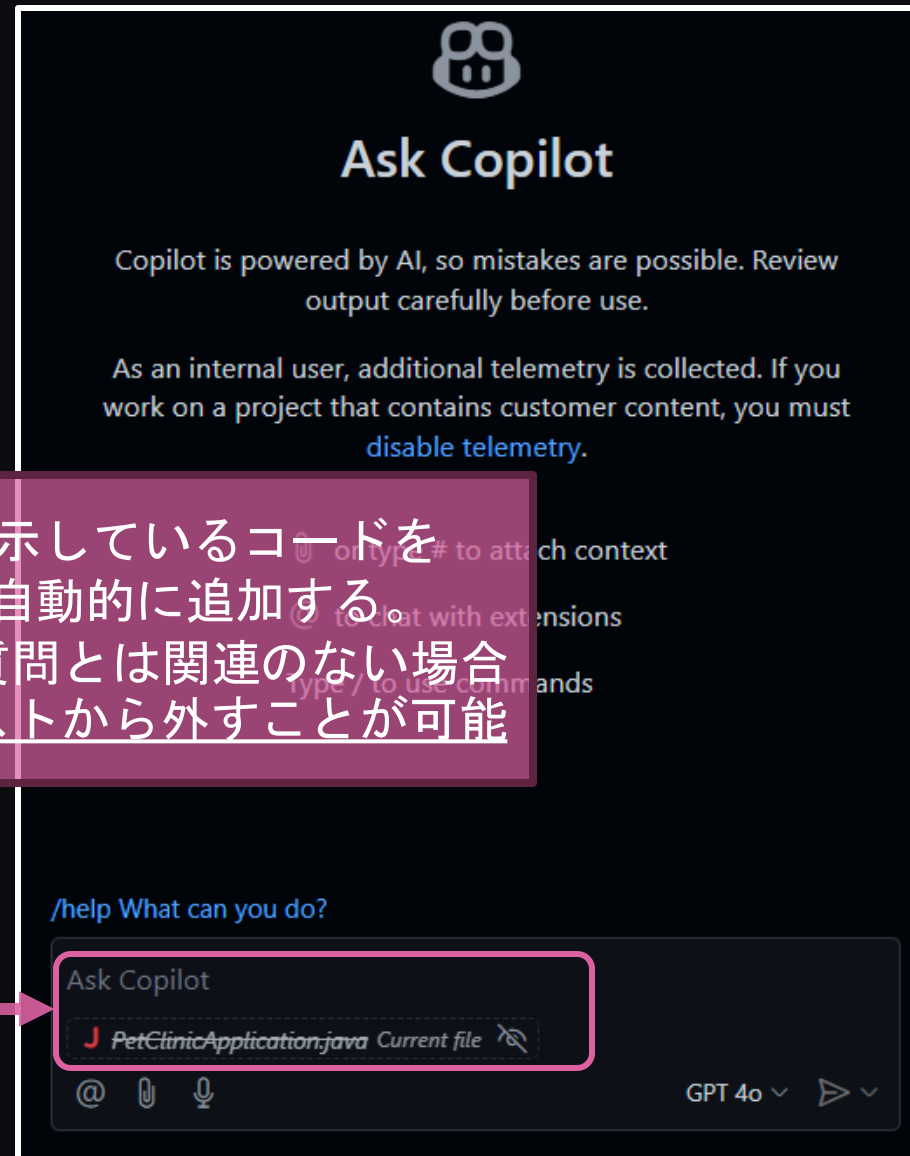
/help What can you do?


Ask Copilot

 PetClinicApplication.java Current file 

   GPT 4o  


Copilot Chat は現在表示しているコードを
コンテキストとして自動的に追加する。
ただし、そのコードが質問とは関連のない場合に
意図的にコンテキストから外すことが可能




Ask Copilot


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






As an internal user, additional telemetry is collected. If you work on a project that contains customer content, you must [disable telemetry](#).

 or type # to attach context
 to chat with extensions
Type / to use commands

/help What can you do?

Ask Copilot

 PetClinicApplication.java Current file 

   GPT 4o  



Interactive workspace symbol links

shinyay

@workspace ManyToManyはどこでどのように使われていますか？

Workspace

Used 46 references

Collecting workspace information

ManyToMany

アノテーションは、src/main/java/org/springframework/samples/petclinic/vet/Vet.javaファイルで使用されています。このアノテーションは、VetクラスとSpecialtyクラスとの多対多の関係を定義しています。

具体的には、

- Open to the Side (Ctrl+Enter)
- Go to Definition
- Go to Type Definitions
- Go to Implementations
- Go to References

@ManyToOne
@JoinTable
private Set<Specialty> specialties;

このコードは、VetクラスがSpecialtyクラスと多対多の関係を持ち、vet_specialtiesという結合テーブルを介して関連付けられていることを示しています。

```
jakarta.persistence-api-3.1.0.jar\jakarta.persistence\ManyToMany.class ×  
102 */  
103 @Target({METHOD, FIELD})  
104 @Retention(RUNTIME)  
105 public @interface ManyToMany {  
106
```

Copilot Chat の応答文の中で指し示している該当コードに対してシンボリックリンクの追加

```
Vet.java ×  
src > main > java > org > springframework > samples > petclinic > vet > Vet.java > Vet  
44 @Entity  
45 @Table(name = "vets")  
46 public class Vet extends Person {  
47  
48     @ManyToOne(fetch = FetchType.EAGER)  
49     @JoinTable(name = "vet_specialties", joinColumns = @JoinColumn(name = "vet_id"),  
50         inverseJoinColumns = @JoinColumn(name = "specialty_id"))  
51     private Set<Specialty> specialties;
```

```
Specialty.java ×  
src > main > java > org > springframework > samples > petclinic > vet > Specialty.java > Specialty  
28 @Entity  
29 @Table(name = "specialties")  
30 public class Specialty extends NamedEntity {  
31  
32 }
```



Fix using Copilot action in the Problem hover

This is a value-based class; programmers should treat instances that are equal as interchangeable and should not use instances for synchronization, or unpredictable behavior may occur. For example, in a future release, synchronization may fail.

API Note:

`Optional` is primarily intended for use as a method return type where there is a clear need to represent "no result," and where using `null` is likely to cause errors. A variable should always point to an `Optional` instance.


Since:


1.8

View Problem (Alt+F8) Quick Fix... (Ctrl+.) **Fix using Copilot (Ctrl+I)**

`Optional<Owner> findById(@`




エディタ上で問題表示されている箇所にカーソルを合わせることで Copilot を使用して問題修正するアクションを表示



 `/fix` The return type is incompatible with '@NonNull Optional<Owner>' returned from `CrudRepository<Owner,Integer>.findById(Integer)` (mismatching null constraints)

 > Used 1 reference

To fix the problem, you need to remove the `@NonNull` annotation from the `findById` method since the `Optional` type already indicates the presence or absence of a value.

Ask Copilot

  GPT 4o 

Accept Discard  

`Optional<Owner> findById(Integer id);`



Workspace indexing

“*GitHub Copilot: Build Local Workspace index*” コマンドにより
明示的にインデックスの作成を開始可能

```
> GitHub Copilot BuildLocal Workspace Index  
GitHub Copilot: Build Local Workspace Index  
Developer: GitHub Copilot Chat Diagnostics  
Developer: Log Working Copies
```

NOTIFICATIONS

\$(copilot) Indexing workspace: 110 of 113 files

Source: GitHub Copilot Chat

Ready

shinyay
@workspace データの登録操作はどこで行われていますか？

Workspace
Used 16 references
Collecting workspace information
Filtering to most relevant information

⚠ Still building the workspace index, response may be less accurate.

データの登録操作は、OwnerController クラスの processCreationForm メソッドで行われています。

```
this.owners.save(owner);  
redirectAttributes.addFlashAttribute("me:  
return "redirect:/owners/" + owner.getId  
}
```

このメソッドは、新しいオーナーのデータを受け取り、データベースに保存します。
Where is the data saved?

NOTIFICATIONS
\$(copilot) Indexing workspace: 116 of 117 files

数多くのファイルを含むワークスペースの
インデックスには時間がかかる場合があり、
インデックス作成中に質問すると警告表示

Semantic search results (September Update) / Sort by relevance in Semantic Search (Experimental)

SEARCH

> owner Aa ab .*

736 results in 38 files - Open in editor

- > LICENSE.txt 8
- > mvnw X
- > mvnw.cmd 1
- ▼ maven-wrapper.propertie... 1
- # regarding copyright ownership....
- ▼ Owner.java src/main/java... 11
- ...springframework.samples.petcli...
- ...domain object representing an ...
- @Table(name = "owners")
- public class Owner extends Perso...
- @JoinColumn(name = "owner_id")
- ...null if none found for this Owner

SEARCH

> save Aa ab .*

26 results in 7 files - Open in editor

- > mvnw.cmd 3
- ▼ Owner.java src/main/java/... 1
- ...new pets (pets that are not save..
- ▼ OwnerController.java src/... 2
- this.owners.save(owner);
- this.owners.save(owner);
- ▼ PetController.java src/mai... 2
- this.owners.save(owner);
- this.owners.save(owner);
- ▼ VisitController.java sr... 1
- this.owners.save(owner);

Github > Copilot > Chat > Search: Semantic Text Results **Experimental**

Enables semantic search results in the search view. Disabling this setting requires reloading the window for the change to take effect.

SEARCH

> owner save Aa ab .*

No results found. Review your settings for configured exclusions and check your gitignore files - [Open Settings](#) - [Learn More](#)

GitHub Copilotにより
意味的に関連性のある検索
(セマンティック検索) を実施

SEARCH

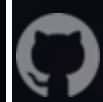
> owner save Aa ab .*

292 results in 24 files - Open in editor

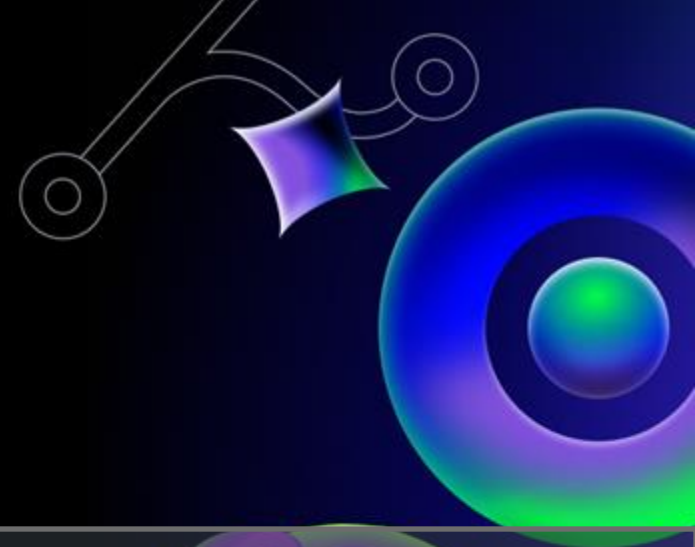
i Results are from Copilot

▼ Text Results

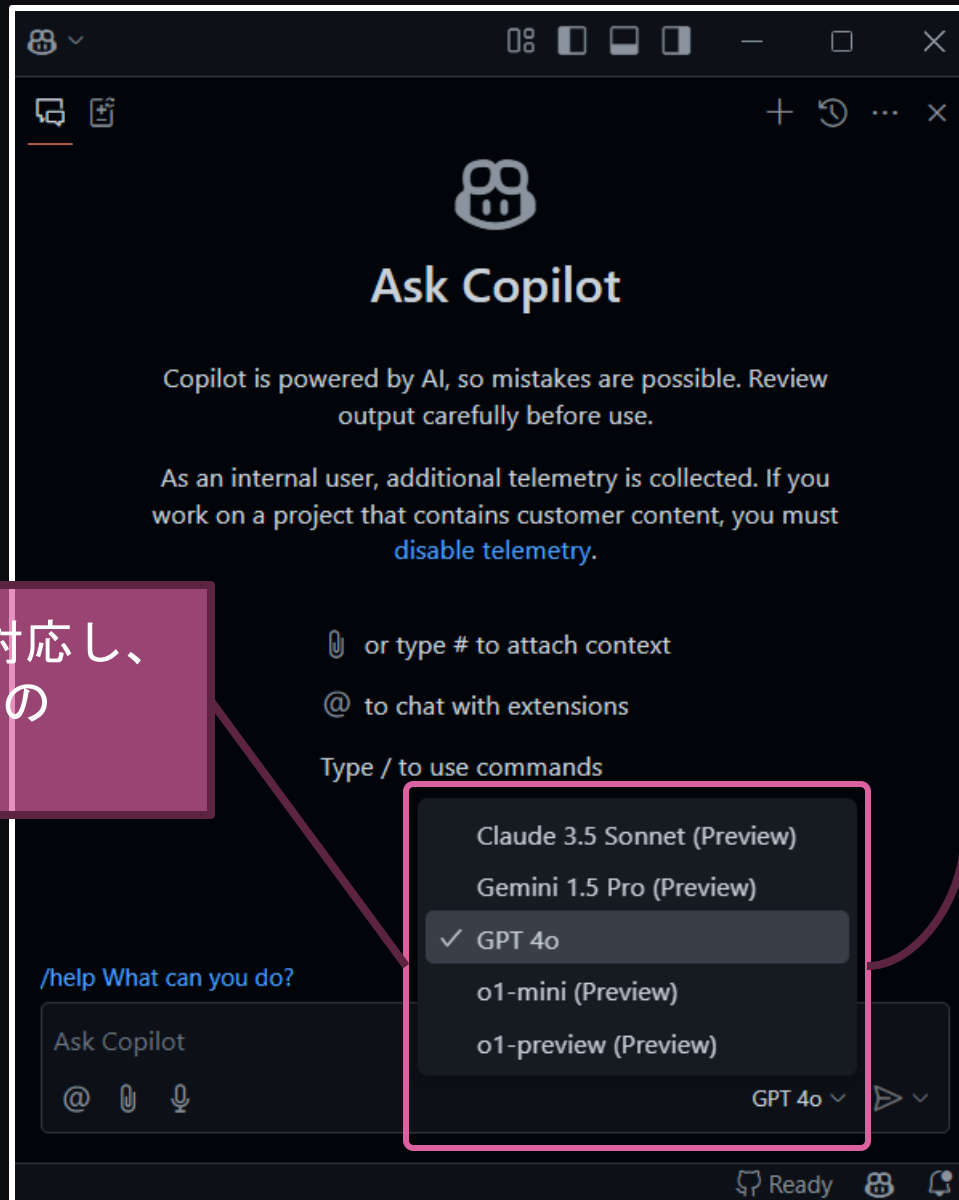
- ▼ Owner.java:13-172 src/main/java/org/s... 15
- ...specific language governing permissions and
- ...springframework.samples.petclinic.owner;
- @JoinColumn(name = "owner_id")
- ...null if none found for this Owner.
- ...such Pet exists for this Owner
- ...null if none found for this Owner.
- ...such Pet exists for this Owner
- ...null if none found for this Owner.
- ...such Pet exists for this Owner
- ...null if none found for this Owner.
- ...new pets (pets that are not saved yet)



September 2024
(version 1.94)



Switch language models in chat



GitHub Copilotがマルチモデル対応し、Anthropic、Google、OpenAI の主要モデルを選択可能

以下のモデルを使用可能
(As of Nov 17, 2024)

Anthropic

❖ Claude 3.5 Sonnet

Google

❖ Gemini 1.5 Pro

OpenAI

❖ GPT-4o

❖ o1-preview

❖ o1-mini



Switch language models in chat (Sample 1/2)

shinyay
@workspace このアプリケーションの概要についておしえてください。

Workspace
Used 50 references
Collecting workspace information...

このアプリケーションは、Spring Frameworkを使用して構築された「Spring PetClinic」サンプルアプリケーションです。ペットクリニックの管理システムを模倣しており、ペットの所有者、ペット、獣医、訪問記録などの管理を行います。

主な機能

- **ペットの所有者管理:** 所有者の情報を追加、編集、削除できます。
- **ペット管理:** ペットの情報を追加、編集、削除できます。
- **獣医管理:** 獣医の情報を表示します。
- **訪問記録管理:** ペットの訪問記録を追加、表示します。

技術スタック

- **Spring Boot:** アプリケーションのベースフレームワーク。
- **Spring MVC:** Webアプリケーションの構築に使用。
- **Spring Data JPA:** データアクセス層の構築に使用。
- **Thymeleaf:** サーバーサイドテンプレートエンジン。
- **H2 Database:** デフォルトのインメモリデータベース。
- **JUnit:** 単体テストフレームワーク。
- **Mockito:** モックオブジェクトの作成に使用。

プロジェクト構成

- **src/main/java:** アプリケーションのメインソースコード。
- **src/test/java:** テストコード。
- **src/main/resources:** アプリケーションの設定ファイルやテンプレート。
- **build.gradle:** Gradleビルド設定ファイル。
- **pom.xml:** Mavenビルド設定ファイル。

What features can we add next?

Ask Copilot

GPT-4o

shinyay
@workspace このアプリケーションの概要についておしえてください。

Workspace
Used 50 references
Collecting workspace information...

Spring PetClinicの概要

Spring PetClinicは、Spring Frameworkの機能を実践的に学ぶためのサンプルアプリケーションです。

主な特徴

- Spring Boot 3.3.5をベースに構築されたWebアプリケーション
- 動物病院の管理システムを模した業務アプリケーション
- Java 17以上の動作をサポート

主な機能

- 飼い主(Owner)の管理
- ペット(Pet)の管理
- 獣医(Vet)の管理
- 診療記録(Visit)の管理

技術スタック

- **フレームワーク:** Spring Boot 3.3.5
- **フロントエンド:** Thymeleaf + Bootstrap
- **データベース:**
 - H2 (デフォルト、インメモリDB)
 - MySQL (オプション)
 - PostgreSQL (オプション)
- **ビルドツール:**
 - Maven
 - Gradle (代替オプション)

What features should we implement next?

Ask Copilot

Claude 3.5 Sonnet (Preview)

shinyay
@workspace このアプリケーションの概要についておしえてください。

Workspace
Used 50 references
Collecting workspace information...

The Spring PetClinic application is a sample application built using the Spring Framework. It demonstrates various Spring features in the context of a pet clinic management system. Key features and technologies used are detailed below. For more information, see the [README.md](#) file.

Core Functionality:

- **Pet Owner Management:** Add, edit, and delete pet owner information.
- **Pet Management:** Add, edit, and delete pet information, including type and birth date.
- **Vet Management:** View information about veterinarians.
- **Visit Management:** Add and view pet visit records.

Technical Stack:

- **Spring Boot:** Provides a simplified and convention-over-configuration approach to building Spring applications. The main application class is `PetClinicApplication`.
- **Spring MVC:** Used for building the web application and handling user interactions. Controllers like `PetController`, `OwnerController`, `VetController`, and `VisitController` manage different aspects of the application.
- **Spring Data JPA:** Simplifies database access using repositories like `OwnerRepository`.
- **Thymeleaf:** Server-side template engine for rendering the user interface.
- **H2 Database (Default):** An in-memory database for easy setup and testing. Persistent database options like MySQL and PostgreSQL are also supported.
- **Caching:** Uses Spring's caching abstraction with a cache for vets defined.

What features should we implement next?

Ask Copilot

Gemini 1.5 Pro (Preview)

Switch language models in chat (Sample 2/2)

The screenshot shows a chat window with the user 'shinyay' asking for a summary of an application. The Copilot response is in Japanese and includes a summary, main features, and a technical stack. A pink box highlights the model selection dropdown in the input field, which is set to 'o1-mini (Preview)'. The 'Workspace' section shows 'Used 18 references'.

shinyay
@workspace このアプリケーションの概要についておしえてください。

Workspace
Used 18 references
Collecting workspace information

このアプリケーションの概要

このアプリケーションは、**Spring PetClinic** をベースに構築されたサンプルアプリケーションで、Spring Frameworkの主要機能を実践的に学ぶためのものです。ペットクリニックの管理システムを模倣し、以下のような機能を提供します。

主な機能

- 飼い主管理:** 飼い主の情報を追加、編集、削除できます。管理は `OwnerRepository` を通じて行われます。
- ペット管理:** ペットの情報（種類、誕生日など）の追加、編集、削除が可能です。
- 獣医管理:** 獣医の情報を表示・管理します。 `VetRepository` がデータアクセス層を担当します。
- 訪問記録管理:** ペットの訪問記録を追加・表示する機能があります。
- キャッシュ管理:** `CacheConfiguration` によるキャッシュ設定を行います。

技術スタック

- フレームワーク:** Spring Boot
- ビルドツール:** Gradle、Maven
- データベース:**
 - H2（デフォルト、インメモリデータベース）
 - MySQL、PostgreSQL（オプション）
- テンプレートエンジン:** Thymeleaf
- テストフレームワーク:** JUnit、Mockito、Testcontainers
- コンテナ管理:** Docker Compose (`docker-compose.yml`)

具体的な機能を教えてください。

Ask Copilot
@ o1-mini (Preview)

The screenshot shows the same chat window as the first image, but with the model selection dropdown set to 'o1-preview (Preview)'. The 'Workspace' section now shows 'Used 19 references'. The Copilot response is in Japanese and includes a summary, main features, and a technical stack.

shinyay
@workspace このアプリケーションの概要についておしえてください。

Workspace
Used 19 references
Collecting workspace information

このアプリケーションの概要

このアプリケーションは、Spring Frameworkを使用して構築されたペットクリニック管理システムのサンプルアプリケーションです。以下に概要をまとめます。

主な機能:

- 飼い主管理:** 飼い主の情報を追加、編集、削除できます。詳細は `OwnerRepository` をご参照ください。
- ペット管理:** ペットの情報（種類、誕生日など）を管理します。
- 獣医管理:** 獣医の情報を表示します。詳細は `VetRepository` をご参照ください。
- 訪問記録管理:** ペットの訪問記録を追加・表示する機能があります。

技術スタック:

- フレームワーク:** Spring Bootを使用しており、エントリーポイントは `PetClinicApplication` です。
- ビルドツール:** GradleとMavenの両方に対応しています。
- データベース:**
 - デフォルト: H2（インメモリデータベース）
 - オプション: MySQL、PostgreSQL (`docker-compose.yml` を使用して起動可能)

プロジェクト構成:

- ソースコード:** `java` にアプリケーションのメインコードがあります。
- テストコード:** `java` にテストコードがあります。例えば、統合テストは `PetClinicIntegrationTests` で行われています。
- 設定ファイル:**
 - アプリケーションプロパティ: `application.properties`

具体的な機能を教えてください。

Ask Copilot
@ o1-preview (Preview)



GPT-4o in Inline Chat

Inline Chat (Ctrl + I) で GPT-4o 利用可能

```
25 public class PetClinicRuntimeHints implements RuntimeHintsRegistrar {
```

このコードは何を処理しますか？



GPT 4o



Claude 3.5 Sonnet (Preview)

Gemini 1.5 Pro (Preview)

✓ GPT 4o

o1-mini (Preview)

o1-preview (Preview)

```
26
```

```
@Override
```

```
27
```

```
public void registerHints(RuntimeHints hints, ClassLoader classLoader) {
```

```
28
```

```
    hints.resources().registerPattern(include:"db/*"); // https://github.com/spring-project
```

```
29
```

```
    hints.resources().registerPattern(include:"messages/*");
```

```
30
```

```
    hints.resources().registerPattern(include:"META-INF/resources/webjars/*");
```

```
31
```

```
    hints.resources().registerPattern(include:"mysql-default-conf");
```

```
32
```

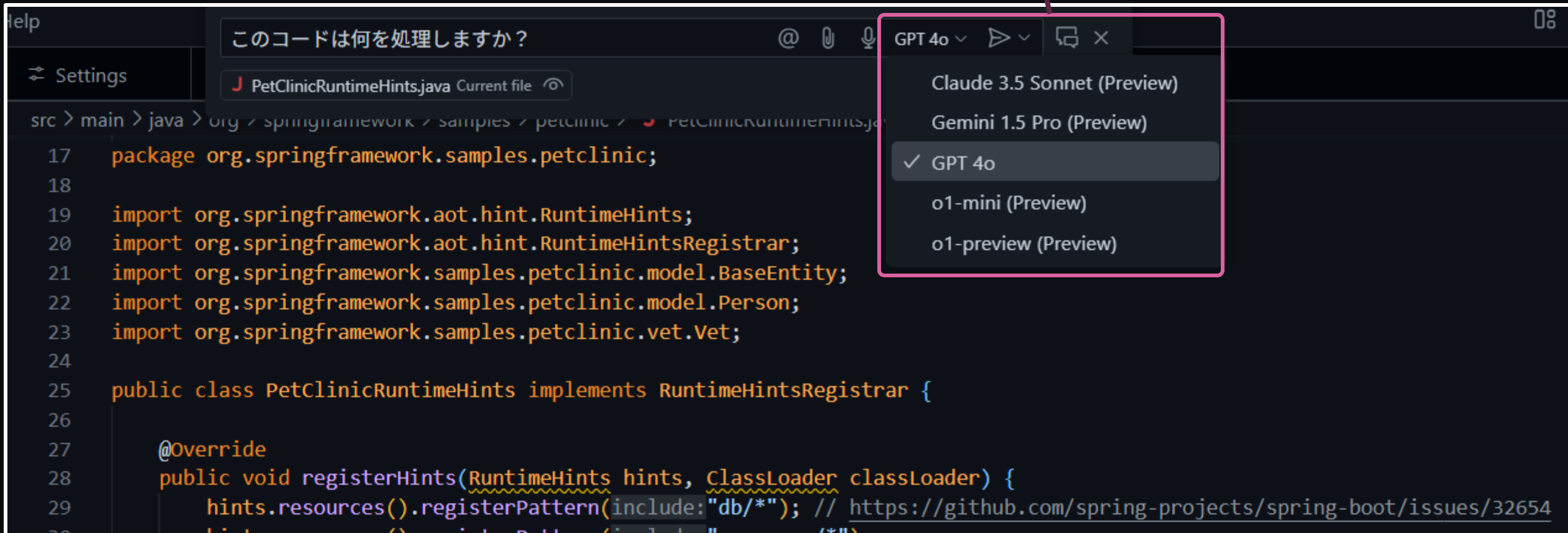
```
    hints.serialization().registerType(type:BaseEntity.class);
```

```
33
```



GPT-4o in Quick Chat

Quick Chat (Ctrl+Shift+Alt+L) で GPT-4o 利用可能



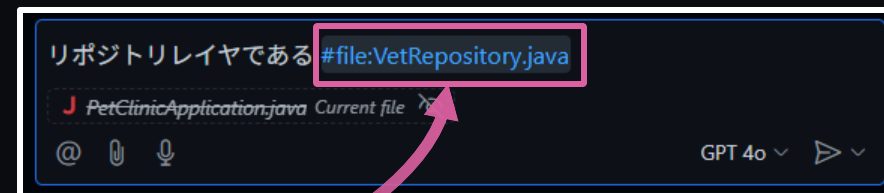
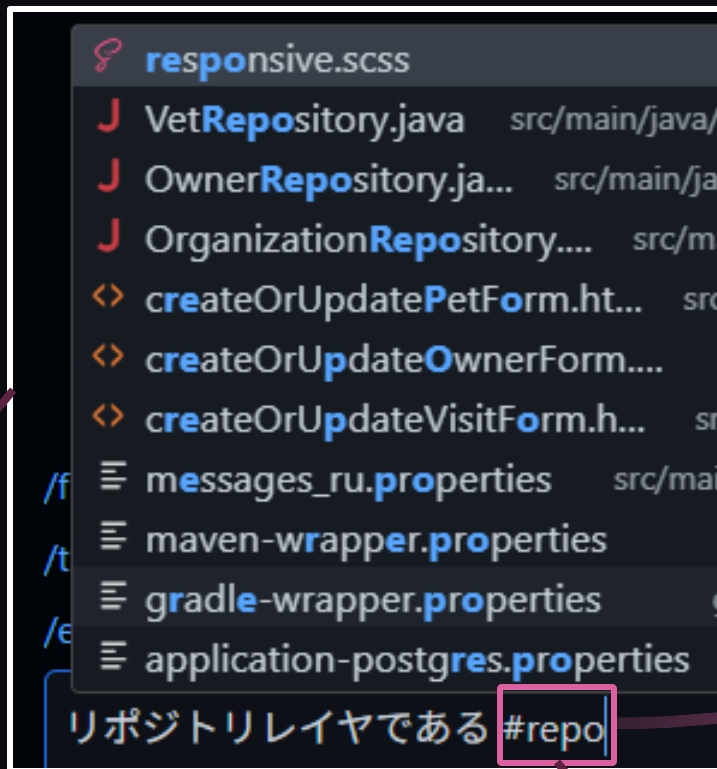
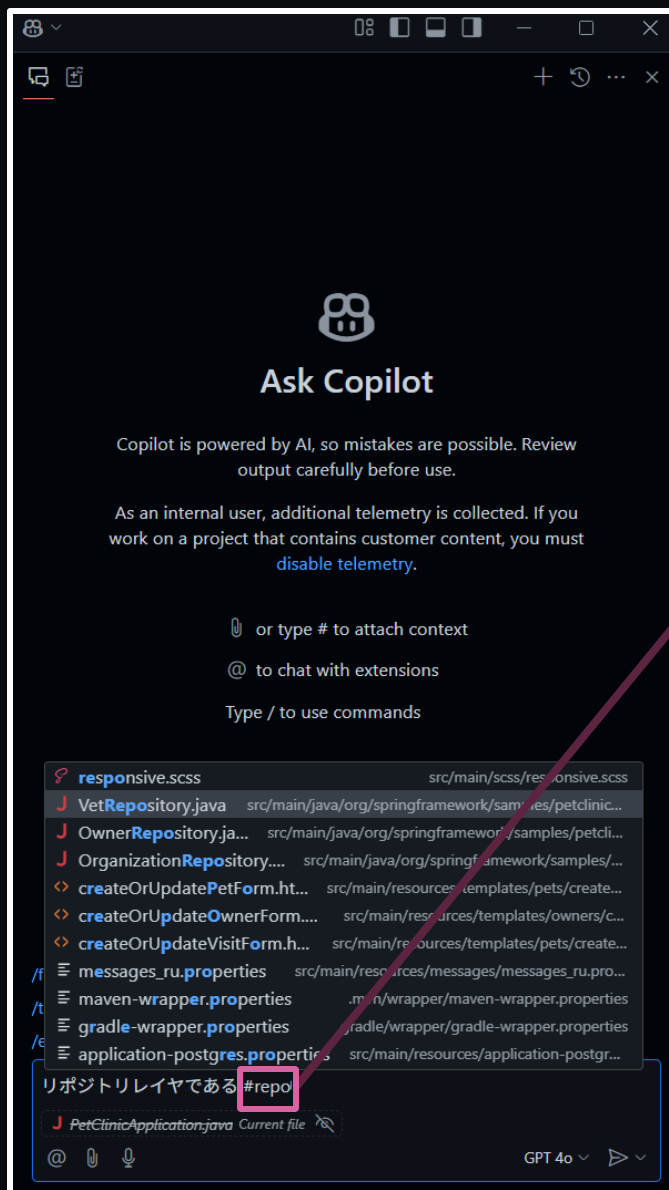
The screenshot shows an IDE interface with a code editor and a Quick Chat window. The code editor displays the following Java code:

```
17 package org.springframework.samples.petclinic;
18
19 import org.springframework.aot.hint.RuntimeHints;
20 import org.springframework.aot.hint.RuntimeHintsRegistrar;
21 import org.springframework.samples.petclinic.model.BaseEntity;
22 import org.springframework.samples.petclinic.model.Person;
23 import org.springframework.samples.petclinic.vet.Vet;
24
25 public class PetClinicRuntimeHints implements RuntimeHintsRegistrar {
26
27     @Override
28     public void registerHints(RuntimeHints hints, ClassLoader classLoader) {
29         hints.resources().registerPattern(include:"db/*"); // https://github.com/spring-projects/spring-boot/issues/32654
30     }
31 }
```

The Quick Chat window is open, showing a list of AI models. The GPT 4o model is selected and highlighted with a checkmark. The other models listed are Claude 3.5 Sonnet (Preview), Gemini 1.5 Pro (Preview), o1-mini (Preview), and o1-preview (Preview).



File suggestions in chat



チャット入力欄でファイル名の提案 (#ファイル名の一部) を入力するとファイル候補一覧が表示される



Improved file links in chat responses

The image shows a chat window with a list of files and folders. A pink box labeled "アイコン表示" (Icon display) highlights the "pom.xml" entry in the list. A second pink box labeled "ドラッグ&ドロップ" (Drag & Drop) shows the "pom.xml" file being dragged from the list to a code editor window. The code editor shows the content of "pom.xml". A third pink box labeled "完全なファイルパスのフローティング表示" (Floating display of the full file path) shows the path "src/main/java/org/springframework/samples/petclinic/owner/OwnerController.java" in a separate window. A fourth pink box labeled "右クリックからのファイル操作" (File operations from right-click) shows a context menu with options like "Open to the Side", "Reveal in Explorer View", "Copy Path", "Copy Relative Path", and "Add File to Chat".

shinyay
@workspace このプロジェクトの構造について教えてください。

Workspace
Used 40 references
Collecting workspace information

このプロジェクトは、Spring PetClinicアプリケーションのソースコードを含むディレクトリです。以下に、プロジェクトの主要なディレクトリとファイルの構造を

- .devcontainer : DevContainer設定ファイルを含むディレクトリ。
- .editorconfig : エディタ設定ファイル。
- .gitattributes : Git属性ファイル。
- .github : GitHub Actionsのワークフロー設定ファイルを含むディレクトリ。
- .gitignore : Gitで無視するファイルを指定するファイル。
- !.gitpod.yml : Gitpod設定ファイル。
- .gradle : Gradleのキャッシュや設定ファイルを含むディレクトリ。
- .mvn : Mavenの設定ファイルを含むディレクトリ。
- .vscode : Visual Studio Codeの設定ファイルを含むディレクトリ。
- build : ビルド成果物を含むディレクトリ。
- build.gradle : Gradleビルド設定ファイル。
- docker-compose.yml : Docker Compose設定ファイル。
- gradle : Gradleラッパー設定ファイルを含むディレクトリ。
- gradlew , gradlew.bat : Gradleラッパースクリプト。
- k8s : Kubernetes設定ファイルを含むディレクトリ。
- LICENSE.txt : ライセンスファイル。
- mvnw , mvnw.cmd : Mavenラッパースクリプト。
- pom.xml : Mavenビルド設定ファイル。
- README.md : プロジェクトのREADMEファイル。
- settings.gradle : Gradle設定ファイル。
- src : ソースコードを含むディレクトリ。
 - main/ : アプリケーションのメインソースコードを含むディレクトリ。
 - java/ : Javaソースコードを含むディレクトリ。
 - resources/ : リソースファイルを含むディレクトリ。
 - test/ : テストコードを含むディレクトリ。
 - java/ : Javaテストコードを含むディレクトリ。
 - resources/ : テストリソースファイルを含むディレクトリ。

このプロジェクトは、Spring Bootを使用して構築されており、MavenまたはGradleを使用してビルドおよび依存関係管理を行います。プロジェクトには、アプリケーションのソースコード、テストコード、設定ファイル、およびビルドスクリプトが含まれています。

pom.xml

ドラッグ&ドロップ

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:
3 <modelVersion>4.0.0</modelVersion>
4
5 <parent>
6 <groupId>org.springframework.boot</groupId>
7 <artifactId>spring-boot-starter-parent</artifactId>
8 <version>3.3.5</version>
```

java/ : Javaソースコードを含むディレクトリ

src/main/java/org/springframework/samples/petclinic/owner/OwnerController.java

完全なファイルパスのフローティング表示

- build : ビルド成果物を含むディレクトリ。
- build.gradle
- docker-com
- gradle : Gradle
- gradlew
- k8s : Kubernetes
- LICENSE.txt
- mvnw ,
- pom.xml

右クリックからのファイル操作

- Open to the Side Ctrl+Enter
- Reveal in Explorer View
- Copy Path Shift+Alt+C
- Copy Relative Path Ctrl+K Ctrl+Shift+C
- Add File to Chat

Refreshed chat user experience

Ask Copilot

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

As an internal user, additional telemetry is collected. If you work on a project that contains customer content, you must [disable telemetry](#).

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

Ask Copilot

@

GPT 4o

'@' をクリックして
チャット参加者や
スラッシュコマンドの入力

- @terminal /explain
- @vscode /search
- @workspace /explain
- @workspace /fix
- @workspace /fixTestFailure
- @workspace /new
- @workspace /newNotebook
- @workspace /setupTests
- @workspace /tests
- /help
- /clear
- Install Chat Extensions...



Semantic search results

SEARCH

> owner Aa ab .*

736 results in 38 files - Open in editor

- > LICENSE.txt 8
- > mvnw X
- > mvnw.cmd 1
- ▼ maven-wrapper.propertie... 1
- # regarding copyright ownership...
- ▼ Owner.java src/main/java... 11
- ...springframework.samples.petcli...
- ...domain object representing an ...
- @Table(name = "owners")
- public class Owner extends Perso...
- @JoinColumn(name = "owner_id")
- ...null if none found for this Owner

SEARCH

> save Aa ab .*

26 results in 7 files - Open in editor

- > mvnw.cmd 3
- ▼ Owner.java src/main/java/... 1
- ...new pets (pets that are not save..
- ▼ OwnerController.java src/... 2
- this.owners.save(owner);
- this.owners.save(owner);
- ▼ PetController.java src/mai... 2
- this.owners.save(owner);
- this.owners.save(owner);
- ▼ VisitController.java sr... 1
- this.owners.save(owner);

Github > Copilot > Chat > Search: Semantic Text Results **Experimental**

Enables semantic search results in the search view. Disabling this setting requires reloading the window for the change to take effect.

SEARCH

> owner save Aa ab .*

No results found. Review your settings for configured exclusions and check your gitignore files - Open Settings - Learn More

GitHub Copilotにより
意味的に関連性のある検索
(セマンティック検索)を実施

SEARCH

> owner save Aa ab .*

292 results in 24 files - Open in editor

Results are from Copilot

▼ Text Results

- ▼ Owner.java:13-172 src/main/java/org/s... 15
- ...specific language governing permissions and
- ...springframework.samples.petclinic.owner;
- @JoinColumn(name = "owner_id")
- ...null if none found for this Owner.
- ...such Pet exists for this Owner
- ...null if none found for this Owner.
- ...such Pet exists for this Owner
- ...null if none found for this Owner.
- ...such Pet exists for this Owner
- ...null if none found for this Owner.
- ...new pets (pets that are not saved yet)



Automated test setup (Experimental)

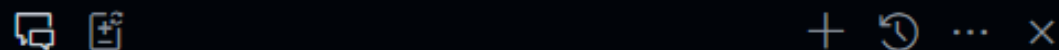
Github > Copilot > Chat > Setup Tests: Enabled **Experimental**
✓ Enables the `/setupTests` intent and prompting in `/tests` generation.

@workspace /setupTests Set up tests in your project (Experimer



GPT 4o

ワークスペース内のコードに対して
自動テスト構成を設定するサポート



shinyay

@workspace /setupTests

Workspace

Given the structure of your workspace, it appears you are working on a Java project with Spring Boot. For testing frameworks, I recommend the following:

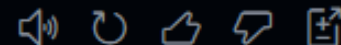
Pick a testing framework

Pick from these options, or use chat to tell me what you'd prefer:

JUnit

TestNG

Spring Test



Start debugging from Chat (Preview)

Github > Copilot > Chat > Start Debugging: Enabled **Preview**

- ✓ Enables the `/startDebugging` intent in panel chat. Generates or finds launch config to match the query (if any), project structure, and more.

Ask Copilot

or type # to attach context

@ to chat with extensions

Type / to use commands

/help What can you do?

@vscode /startDebugging gradle

PetClinicApplication.java:31 Current file

GPT 4o

@vscode/startDebugging コマンドにより起動構成を作成し Copilot チャットからシームレスにデバッグ実行が可能

```
{
  "configurations": [
    {
      "type": "java",
      "request": "launch",
      "name": "Launch Gradle Program",
      "mainClass": "${input:mainClass}",
      "projectName": "${input:projectName}",
      "preLaunchTask": "build"
    }
  ],
  "inputs": [
    {
      "type": "promptString",
      "id": "mainClass",
      "description": "Enter the fully qualified main class name"
    },
    {
      "type": "promptString",
      "id": "projectName",
      "description": "Enter the project name"
    }
  ]
},
{
  "tasks": [
    {
      "type": "shell",
      "label": "build",
      "command": "./gradlew",
      "args": [
        "build"
      ]
    }
  ]
}
}
```

Run Task and Start Debugging

Save Task and Configuration

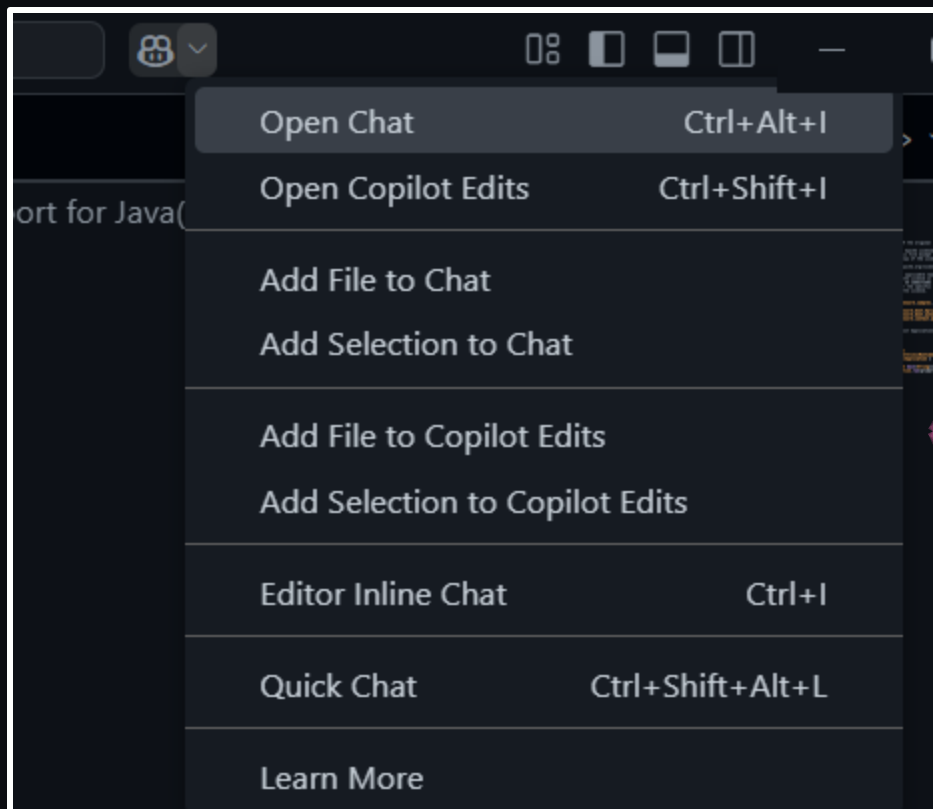
Debugging can be started in the Debug View or by using the Start Debugging Command.

Start Debugging

Save Configuration



Chat in Command Center (Preview)



様々なチャットの開始や
プロンプトへのコンテキストの
アタッチに素早くアクセス可能

Copilot チャット (セカンダリサイドバー)

Copilot エディット (セカンダリサイドバー)

指定ファイルをチャットのコンテキストに追加

指定範囲をチャットのコンテキストに追加

指定ファイルを Edit のコンテキストに追加

指定範囲を Edit のコンテキストに追加

インラインチャット

クイックチャット



Custom instructions (Experimental)

GitHub Copilot Chat

Experimental

Github > Copilot > Chat > Code Generation: Instructions Experimental

A set of instructions that will be added to Copilot requests that generate code. Instructions can come from:

- a file in the workspace: `{ "file": "fileName" }`
- text in natural language: `{ "text": "Use underscore for field names." }`

Note: Keep your instructions short and precise. Poor instructions can degrade Copilot's quality and performance.

Edit in settings.json

コードを生成する
全ての Copilot リクエストに追加される
一連の命令を定義

```
32 ..github.copilot.chat.commitMessageGeneration.instructions
33 ..github.copilot.chat.codeGeneration.instructions":
34   {
35     "file": ".copilot-instructions.md"
36   }
37 ],
38 }
```

settings.json

Preview copilot-instructions.md ×

13. **[Recommended]** Carefully check whether a *k/v collection* can store *null* value, refer to the table below:

Collection	Key	Value	Super	Note
Hashtable	Null is not allowed	Null is not allowed	Dictionary	Thread-safe
ConcurrentHashMap	Null is not allowed	Null is not allowed	AbstractMap	Segment lock
TreeMap	Null is not allowed	Null is allowed	AbstractMap	Thread-unsafe
HashMap	Null is allowed	Null is allowed	AbstractMap	Thread-unsafe

Counter example: Confused by `HashMap`, lots of people think *null* is allowed in `ConcurrentHashMap`. Actually, `NullPointerException` will be thrown when putting in *null* value.

14. **[For Reference]** Properly use sort and order of a collection to avoid negative influence of unsorted and unordered one.

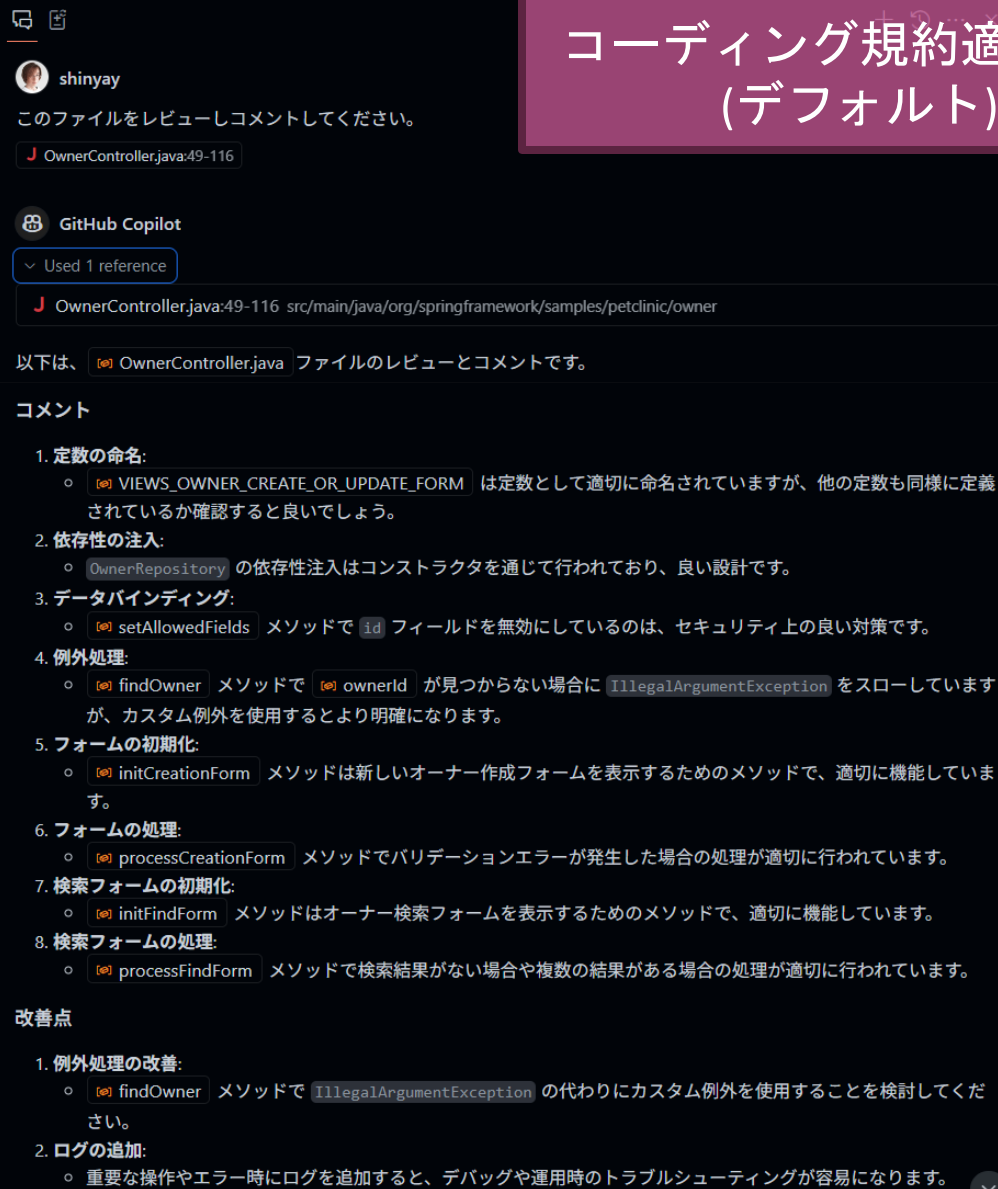
Note: *Sorted* means that its iteration follows specific sorting rule. *Ordered* means the order of elements in each traverse is stable. e.g. `ArrayList` is ordered and unsorted, `HashMap` is unordered and unsorted, `TreeSet` is ordered and sorted.

コード規約例



Custom instructions (Experimental)

コーディング規約適用なし
(デフォルト)



shinyay
このファイルをレビューしコメントしてください。

OwnerController.java:49-116

GitHub Copilot
Used 1 reference

OwnerController.java:49-116 src/main/java/org/springframework/samples/petclinic/owner

以下は、OwnerController.java ファイルのレビューとコメントです。

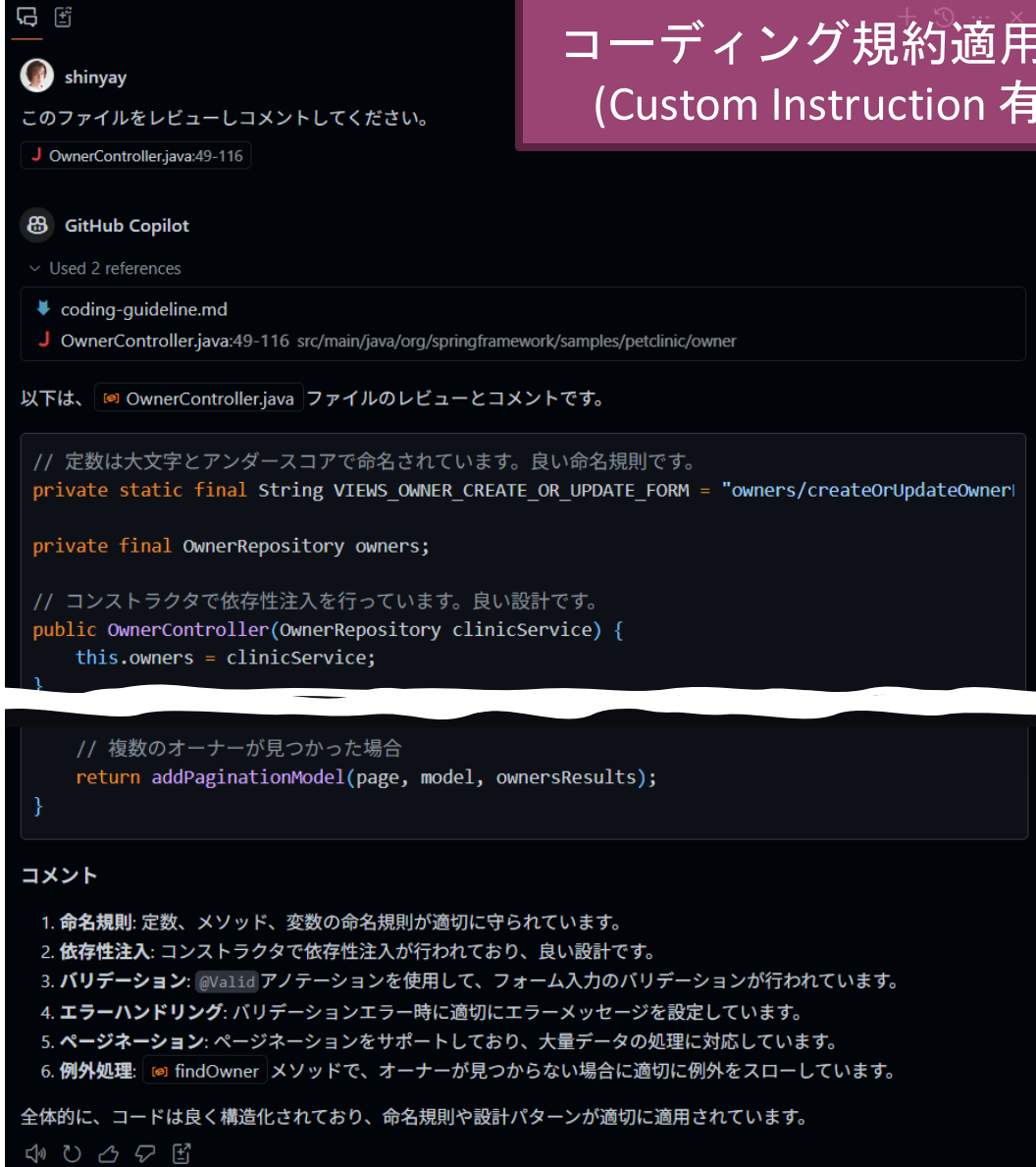
コメント

- 定数の命名:
 - VIEWS_OWNER_CREATE_OR_UPDATE_FORM は定数として適切に命名されていますが、他の定数も同様に定義されているか確認すると良いでしょう。
- 依存性の注入:
 - OwnerRepository の依存性注入はコンストラクタを通じて行われており、良い設計です。
- データバインディング:
 - setAllowedFields メソッドで id フィールドを無効にしているのは、セキュリティ上の良い対策です。
- 例外処理:
 - findOwner メソッドで ownerId が見つからない場合に IllegalArgumentException をスローしていますが、カスタム例外を使用するとより明確になります。
- フォームの初期化:
 - initCreationForm メソッドは新しいオーナー作成フォームを表示するためのメソッドで、適切に機能しています。
- フォームの処理:
 - processCreationForm メソッドでバリデーションエラーが発生した場合の処理が適切に行われています。
- 検索フォームの初期化:
 - initFindForm メソッドはオーナー検索フォームを表示するためのメソッドで、適切に機能しています。
- 検索フォームの処理:
 - processFindForm メソッドで検索結果がない場合や複数の結果がある場合の処理が適切に行われています。

改善点

- 例外処理の改善:
 - findOwner メソッドで IllegalArgumentException の代わりにカスタム例外を使用することを検討してください。
- ログの追加:
 - 重要な操作やエラー時にログを追加すると、デバッグや運用時のトラブルシューティングが容易になります。

コーディング規約適用あり
(Custom Instruction 有効)



shinyay
このファイルをレビューしコメントしてください。

OwnerController.java:49-116

GitHub Copilot
Used 2 references

coding-guideline.md
OwnerController.java:49-116 src/main/java/org/springframework/samples/petclinic/owner

以下は、OwnerController.java ファイルのレビューとコメントです。

```
// 定数は大文字とアンダースコアで命名されています。良い命名規則です。  
private static final String VIEWS_OWNER_CREATE_OR_UPDATE_FORM = "owners/createOrUpdateOwnerForm";  
  
private final OwnerRepository owners;  
  
// コンストラクタで依存性注入を行っています。良い設計です。  
public OwnerController(OwnerRepository clinicService) {  
    this.owners = clinicService;  
}
```

```
// 複数のオーナーが見つかった場合  
return addPaginationModel(page, model, ownersResults);  
}
```

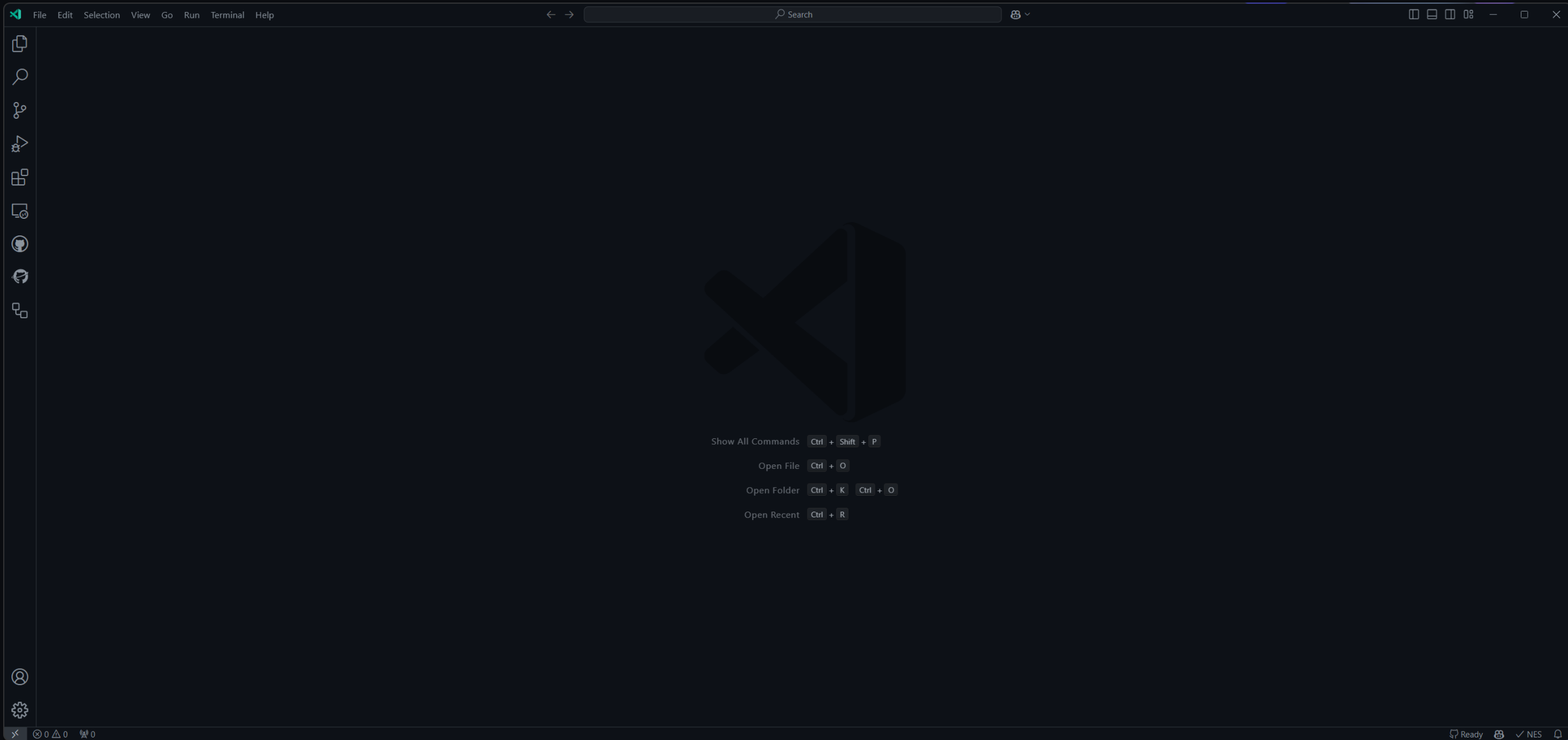
コメント

- 命名規則: 定数、メソッド、変数の命名規則が適切に守られています。
- 依存性注入: コンストラクタで依存性注入が行われており、良い設計です。
- バリデーション: @Valid アノテーションを使用して、フォーム入力のバリデーションが行われています。
- エラーハンドリング: バリデーションエラー時に適切にエラーメッセージを設定しています。
- ページネーション: ページネーションをサポートしており、大量データの処理に対応しています。
- 例外処理: findOwner メソッドで、オーナーが見つからない場合に適切に例外をスローしています。

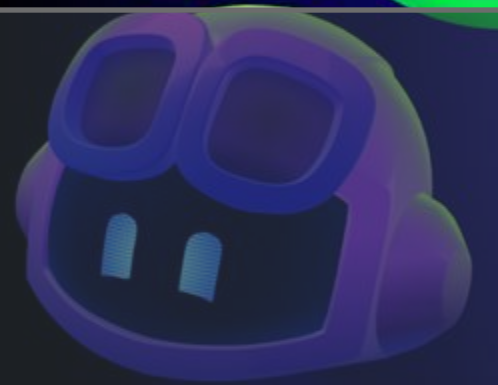
全体的に、コードは良く構造化されており、命名規則や設計パターンが適切に適用されています。



デモンストレーション

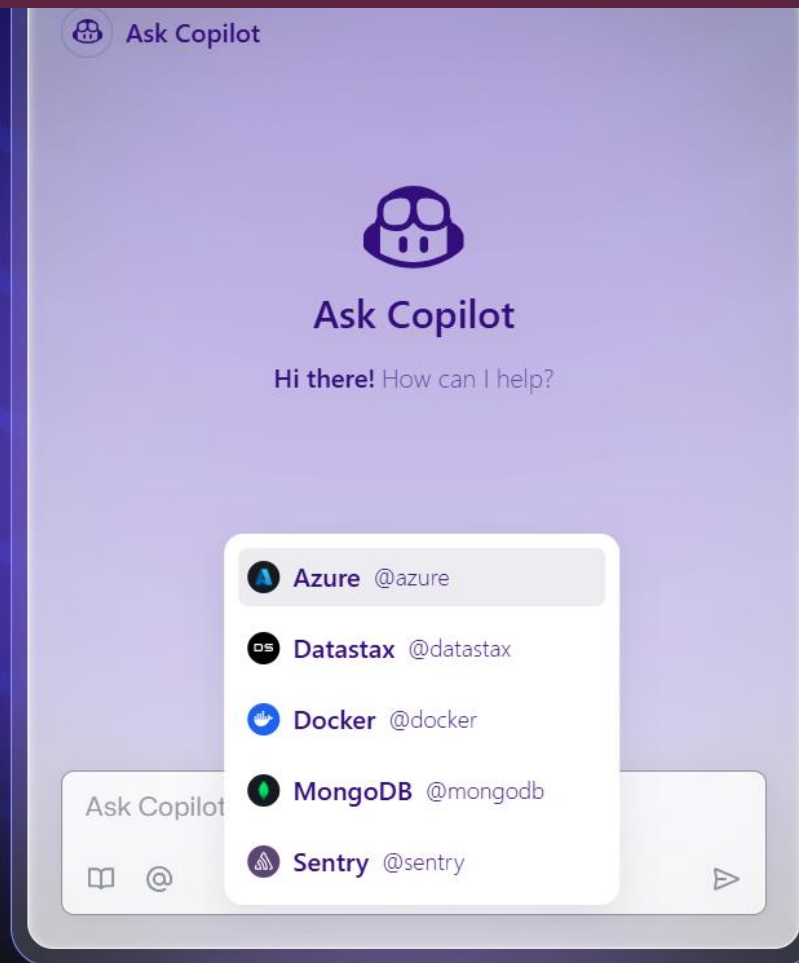


GitHub Copilot Extensions



Build and deploy with the tools you love in Copilot Chat

GitHub Copilot を拡張し API、ドキュメント、ガイドを備えた独自の開発者プラットフォームを構築



Explore Extensions



Copilot Extensions

Extend Copilot capabilities using third party tools, services, and

GitHub Marketplace

Filter: All

By: All creators

- Docker for GitHub Copilot** Copilot
Learn about containerization, generate Docker assets and analyze project vulnerabilities in GitHub Copilot
- PerplexityAI** Copilot
Perplexity answers questions as you code by searching the web
- ReadMe API** Copilot
Ask questions about the ReadMe API and get help with code, directly in VS code
- Mermaid Chart** Copilot
Provides advanced and powerful diagramming and visualization to GitHub Copilot Chat
- Stack Overflow** Copilot
Get answers to your most complex coding questions right where you're already working
- Models (GitHub)** Copilot
Copilot Extension to connect and chat with GitHub Models
- Product Science** Copilot
Get help with performance optimization techniques to make your code faster
- Sentry for GitHub Copilot** Copilot
The Sentry Copilot Extension allows developers to fix broken code from the GitHub UI
- GitBook for GitHub Copilot** Copilot
Leverage your GitBook documentation to answer questions, providing instant responses in your workflow
- Octopus for GitHub Copilot** Copilot
The Octopus extension allows a cloud hosted Octopus instance to be queried via the GitHub Copilot chat interface
- LambdaTest** Copilot
Design, implement and analyze test automation right where you code
- Teams Toolkit** Copilot
Chat with GitHub Copilot extension for Teams Toolkit to build Teams apps or customize and extend Microsoft 365 Copilot
- DataStax** Copilot
Chat with your data! Write queries and code with natural language
- Pinecone AI** Copilot
Developers can leverage this app to gain crucial context or build AI applications with vector databases
- Atlassian Rovo** Copilot
Atlassian Rovo for GitHub Copilot brings your company's knowledge to your code
- LaunchDarkly for GitHub Copilot** Copilot
The LaunchDarkly Copilot Extension, available under @launchdarkly-app
- New Relic Copilot Extension** Copilot
New Relic AI-powered insights for app performance, monitoring, and deployment optimization
- Agentic Search Ai** Copilot
This extension leverages the power of agentic systems to enhance your coding experience. Agentic systems are intelligent
- Spice.ai for GitHub Copilot** Copilot
Chat with repositories, external databases, data warehouses, and data lakes directly in GitHub Copilot
- Skyramp for GitHub Copilot** Copilot
Your Personal Quality Engineer

Visual Studio Marketplace

Visual Studio | Visual Studio Code

Visual Studio Marketplace

tag:chat-participant

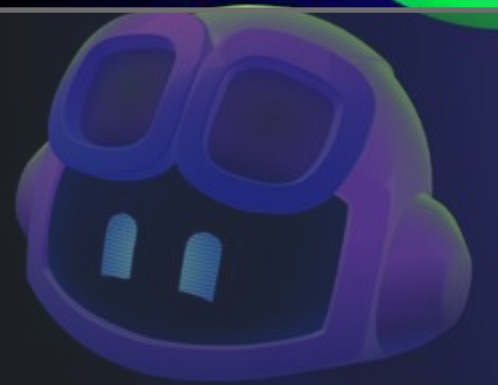
32 Results

Showing: All categories | Sort By: Relevance

- Remote - SSH** Microsoft
Open any folder on a remote machine using SSH and take advantage of VS Code's full...
- GitHub Pull Requests** GitHub
Pull Request and Issue Provider for GitHub
- GitHub Copilot Chat** GitHub
AI chat features powered by Copilot
- MongoDB for VS Code** MongoDB
Connect to MongoDB and Atlas directly from your VS Code environment, navigat...
- Ruby LSP** Shopify
VS Code plugin for connecting with the Ruby LSP
- Regex Previewer** Christof Marti
Regex matches previewer for JavaScript, TypeScript, PHP and Haxe in Visual Studio...
- Stripe** Stripe
Build, test, and use Stripe inside your editor.
- GitHub Copilot for Azure** Microsoft
GitHub Copilot for Azure is the @azure extension. It's designed to help streamline...
- DBCode** DBCode
Database management in Visual Studio Code
- vscode-mermaid** Microsoft
Exploration into Copilot Chat-powered Diagram Generation
- Flexpivot** Flexpivot
Open-Source, Native and a True GitHub Copilot Alternative for VS Code
- LaunchDarkly** LaunchDarkly
Manage LaunchDarkly feature flags directly from within your code
- Vision for Copilot Pre** Microsoft
A chat extension providing vision capabilities in VS Code, with a focus on accessibility.
- GenAIScript** Microsoft
Generative AI Scripting.
- Data Analysis for Cop** Microsoft
This tool extends the LLM's capabilities by allowing it to run Python code in a...
- Web Search for Copil** Microsoft
Gives access to search engines from within Copilot.
- VS Code Commander** Microsoft
Enables users to configure VS Code using chat. The extension allows to search f...
- Custom Prompts for C** Julien Tanay
GitHub Copilot chat extension enabling re-use of prompts across developer teams.
- Grug Brained Develop** shouples
A layman's chat participant, helping to think like the self-aware, small brained.
- Code Runner Copilot** Jun Han
Your Copilot to let you use "Code Runner" extension to run code in VS Code with...
- Multicoder - Claude S** Multicoder
Add Claude 3.5 Sonnet to Multicoder (BYOK)
- GitHub Copilot Code** Code review for g...
An extension to assist developers with code reviews using GitHub Copilot
- Copilot Chronicle** M365 Chronicle A...
Copilot Chronicle assistant to apply code changes, migrations, deprecations an...
- CoCover** Maya Shavin Studi...
Your AI Assistant in enhancing your markdown content with generated professional cove...
- Babylon Copilot Agen** MackeyKinard
CODEWRX - Babylon Toolkit Copilot Chat Agent
- inline-test** YanQianyu
行内测试
- Unity Copilot Agent** CODEWRX - Unity Toolk...
CODEWRX - Unity Toolkit Copilot Chat Agent
- LGTM - AI Code Review** Christian Puverm...
Review source code changes using GitHub Copilot Chat
- codest** CodeLinkID
code generator
- Unreal Copilot Agent** MackeyKinard
CODEWRX - Unreal Toolkit Copilot Chat Agent



GitHub Copilot for Azure



GitHub Copilot for Azure (Preview)

Introducing GitHub Copilot for Azure (preview)

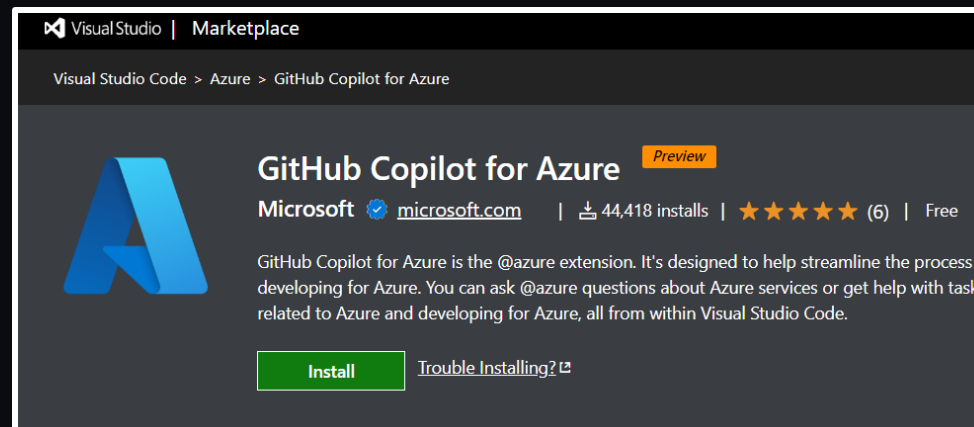
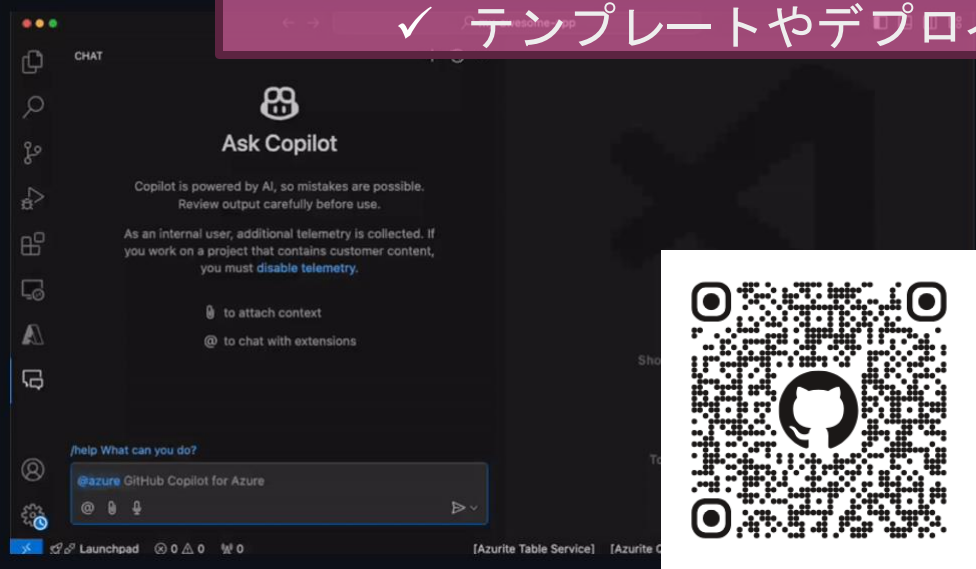
November 15, 2024 by Chris Harris, Product Manager

I'm thrilled to introduce the preview of GitHub Copilot for Azure - a new tool that integrates effortlessly with GitHub Copilot Chat in VS Code. Imagine it as your personal guide for navigating the Azure cloud. No more toggling between your IDE and the Azure portal to manage infrastructure or look up commands and arguments. Now, you can concentrate on your core task - coding. Whether you're setting up services or deploying applications, simply prompt `@azure` in the Chat view and manage everything directly within your editor.

You can get started right now by installing [GitHub Copilot for Azure](#) from the Visual Studio Marketplace!

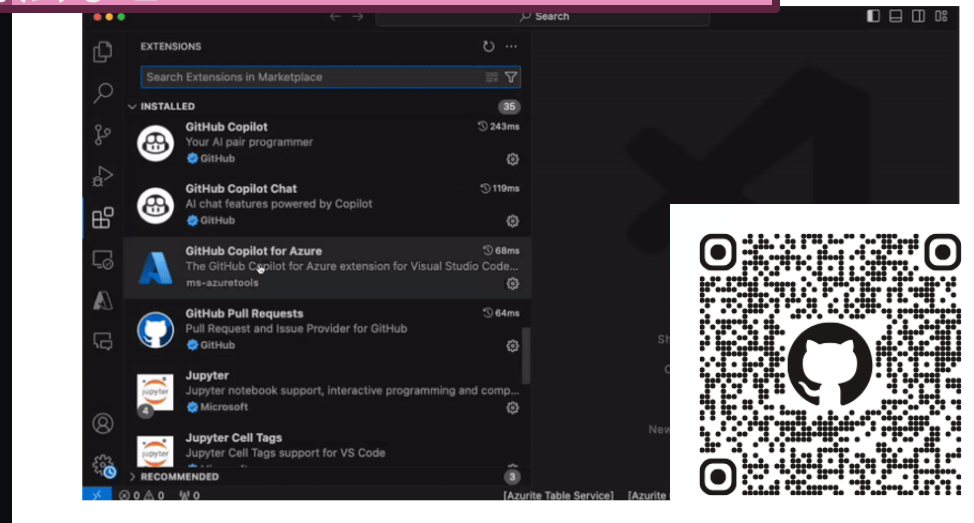
GitHub Copilot for Azure can help with different tasks throughout the lifecycle of building and running your application.

Learning about Azure



GitHub Copilot for Azureはユーザーのコンテキストを理解した上で回答

- ✓ 利用可能なMicrosoft Azureのリソースや使用状況
- ✓ システムに適したMicrosoft Azureのサービス
- ✓ テンプレートやデプロイの方法など



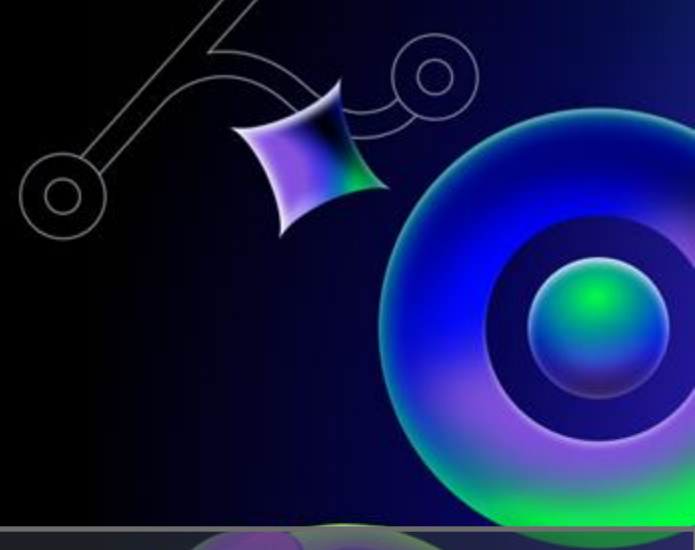
デモンストレーション

The image shows a screenshot of a chat application interface with a dark theme. The interface includes a top menu bar with 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', 'Terminal', and 'Help'. A search bar is located in the top right. On the left side, there is a vertical toolbar with icons for copy, search, refresh, and other functions. The main area contains six chat messages, each in a white-bordered box. Each message consists of a user prompt starting with '@azure' and an AI response. The AI responses are generated by 'GPT 4o', as indicated by the model name and a play button icon at the end of each message. The prompts and responses are as follows:

- Message 1:** Prompt: '@azure サンプルアプリケーション chatgpt-quickstart をazd テンプレートを使用して作成とデプロイをするサポートをしてください。' Response: '@azure 'azd init' を実行したあと、何をしたらいいですか？'
- Message 2:** Prompt: '@azure 'azd init' コマンドとは何をするものですか？' Response: '@azure 今作られているリソースのランニングコストについて教えてください。'
- Message 3:** Prompt: '@azure このテンプレートで何が作成されますか？' Response: '@azure クリーンアップするにはどうしたらいいですか？'

At the bottom of the window, there is a system tray with icons for network, volume, and battery, and the text 'Ready NES'.

Key Takeaways



本日伝えたかったこと

