AI驱动的智能开发工具，提升编程效率

# 什么是 Trae IDE？

Trae IDE 与 AI 深度集成，提供智能问答、代码自动补全以及基于 Agent 的 AI 自动编程能力。使用 Trae 开发项目时，你可以与 AI 灵活协作，提升开发效率。

* 中文版官网地址：[www.trae.com.cn](http://www.trae.com.cn)
* Trae 中文介绍视频



## 完备的 IDE 功能

提供传统的 IDE 功能，包括代码编写、项目管理、插件管理、源代码管理等。

## 强大的 AI 助手

* **智能问答**

在编写代码时，你可以随时与 AI 助手对话，获得代码解释、代码注释和错误修复等帮助。

* **实时代码建议**

AI 助手能够理解当前代码并在编辑器中实时提供代码建议，提升编程效率。

* **代码片段生成**

通过自然语言描述你的需求，AI 助手将生成相应的代码片段，甚至能够编写项目级或跨文件的代码。

* **从 0 到 1 开发项目**

告诉 AI 助手你想开发的程序，AI 助手将根据你的描述提供相关代码或自动创建所需文件。

* **支持的操作系统**

macOS 10.15 及以上

Windows 10、11

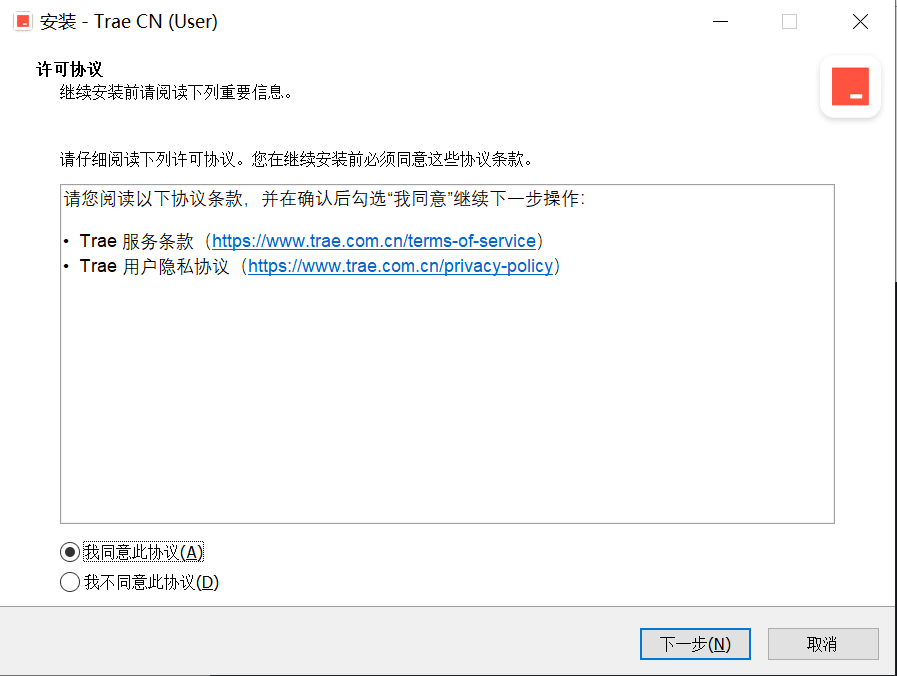
# Trae安装、配置与开始开发

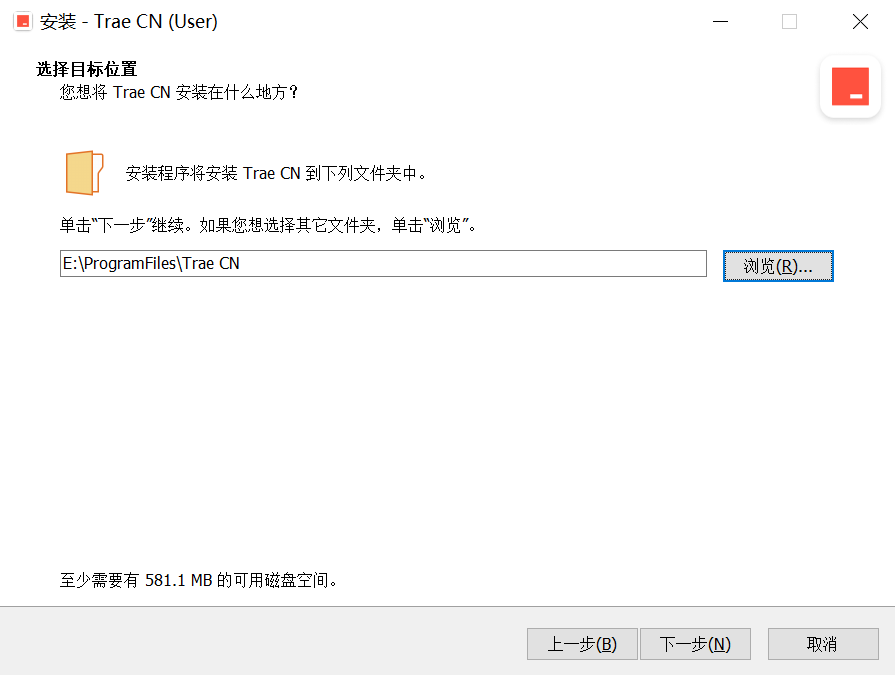
## 第一步：安装并启动 Trae

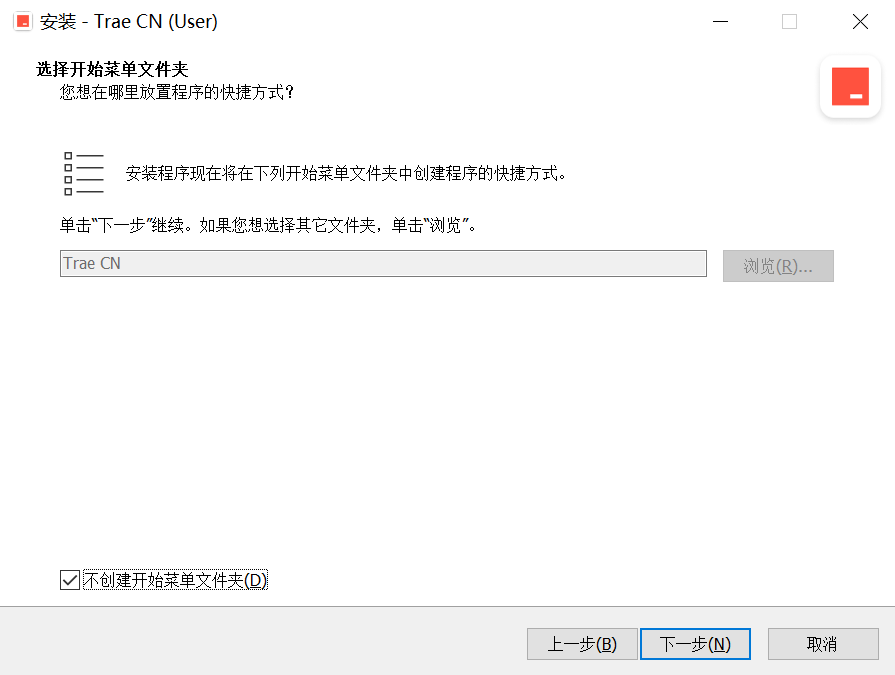
访问 Trae 的官方网站，并在你的电脑上安装 Trae 。

* Trae 中文版下载地址：[www.trae.com.cn/download](http://www.trae.com.cn/download)

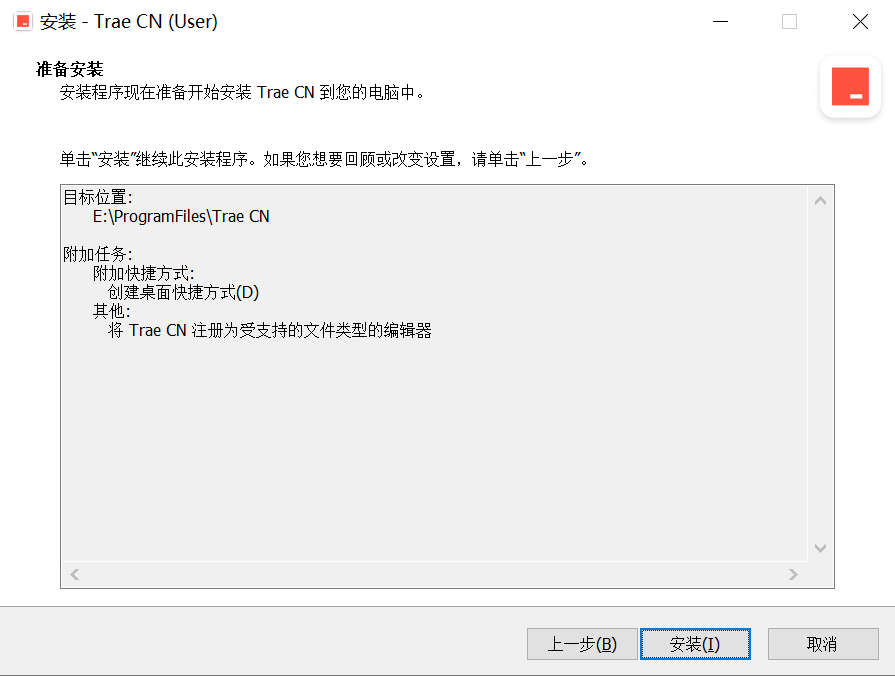




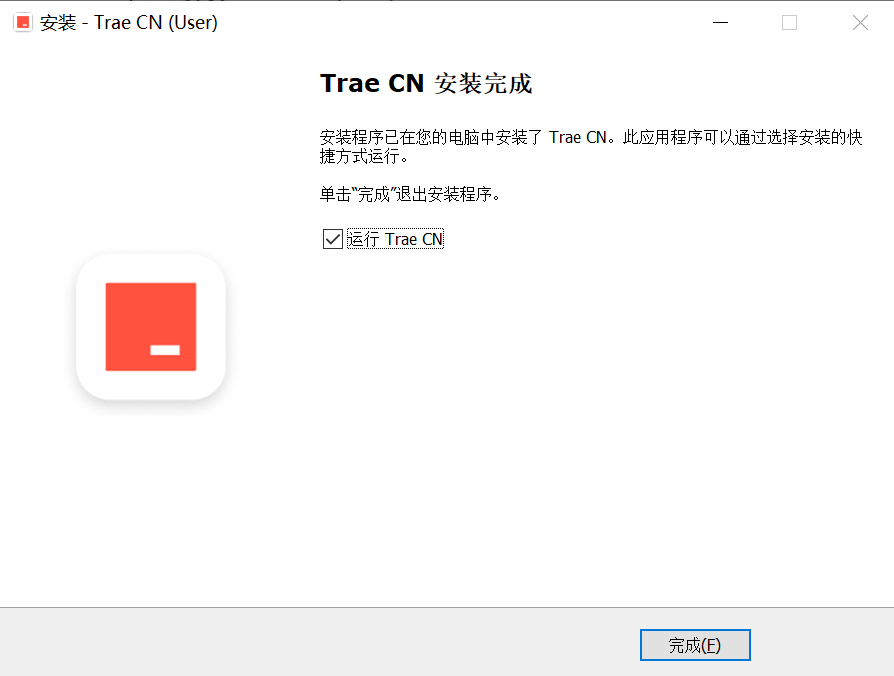












Trae 首次启动时，你会进入以下页面。



点击 开始 按钮。

Trae 初始配置流程开始。

## 第二步：选择主题和语言

1. 选择主题。可选项为 暗色、亮色 和 深蓝。
2. 选择显示语言。可选项为 简体中文、English 和 日语。
3. 点击 继续 按钮。



## **第三步：从 VS Code 或 Cursor 中导入配置**

若你的电脑中已安装并配置 VS Code 或 Cursor，你可以点击”从VS Code导入”或”从Cursor导入”按钮或从”跳过”按钮。系统将会从对应的 IDE 中获取插件、IDE 设置、快捷键设置等信息并一键导入到 Trae 中，帮助你快速从其他 IDE 切换到 Trae。



## **第四步：添加 Trae 相关的命令行**

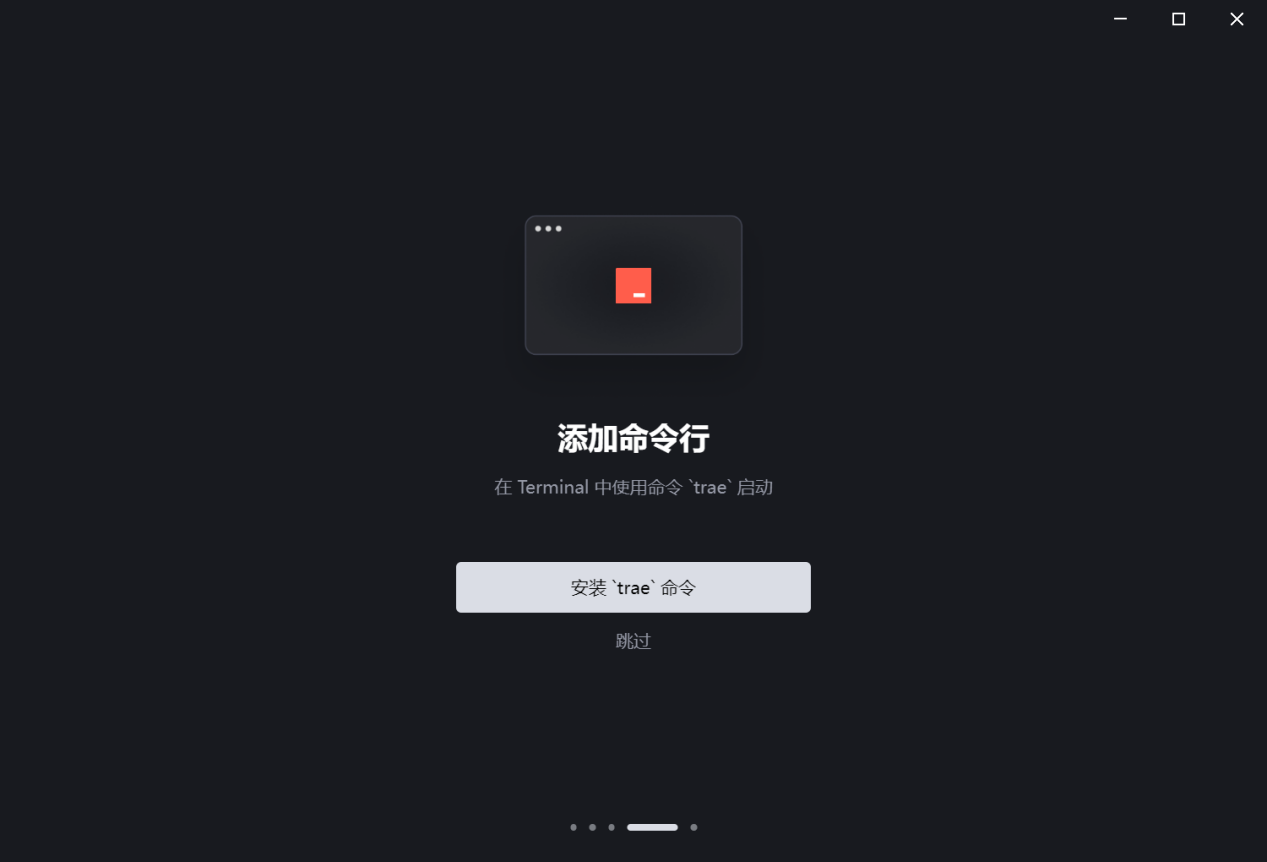
添加 Trae 相关的命令行后，你可以在终端中使用命令行更快速地完成 Trae 相关的操作。例如：

* 使用 trae 命令快速唤起 Trae。
* 使用 trae my-react-app 命令在 Trae 中打开一个项目。

点击 安装 `trae` 命令 按钮，然后完成授权流程。

提示：

若未在此处安装 Trae 相关的命令行，后续可在 IDE 中使用快捷键（macOS：Command + Shift + P；Windows：Ctrl + Shift + P）打开命令面板，然后使用 install 'trae' command in PATH 命令进行安装。

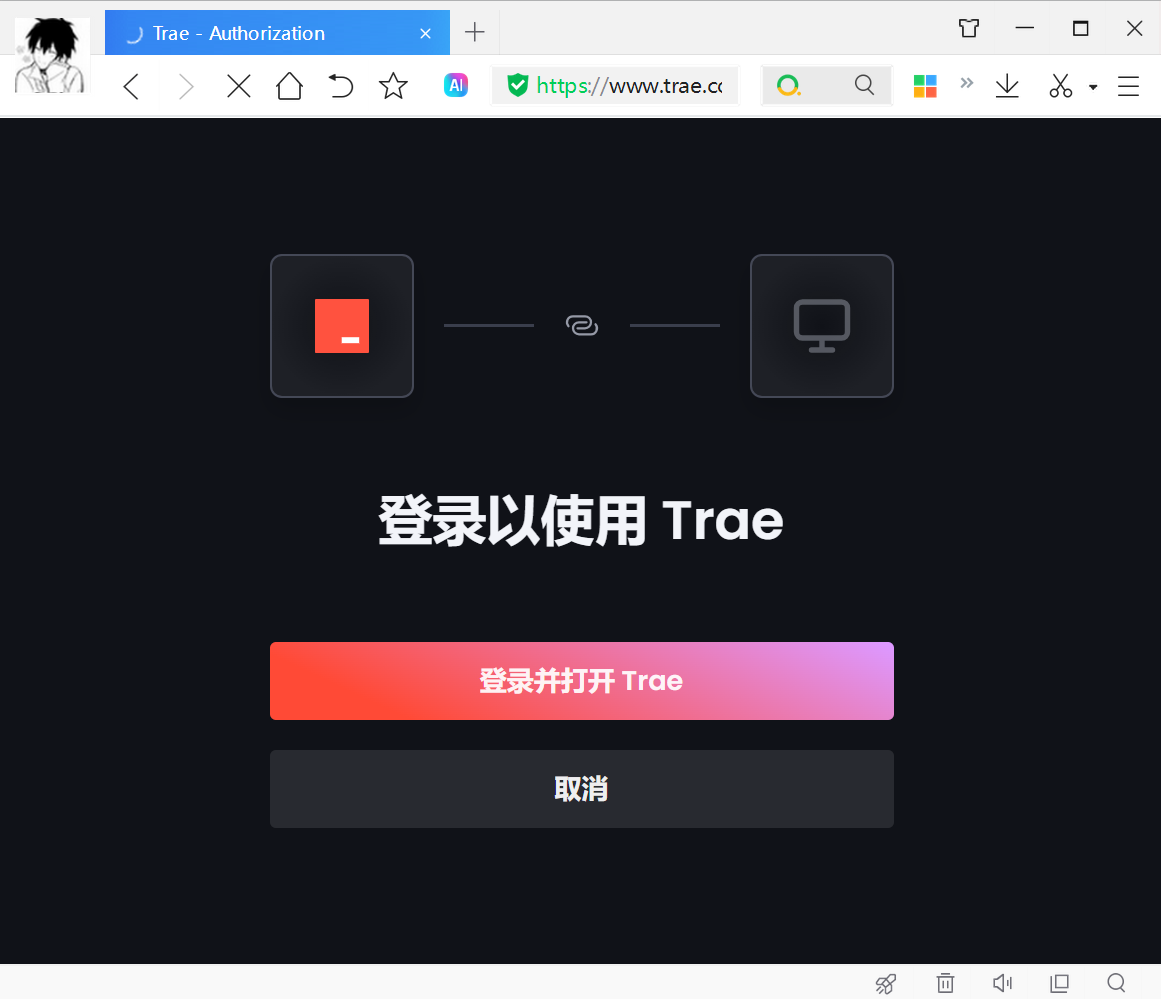


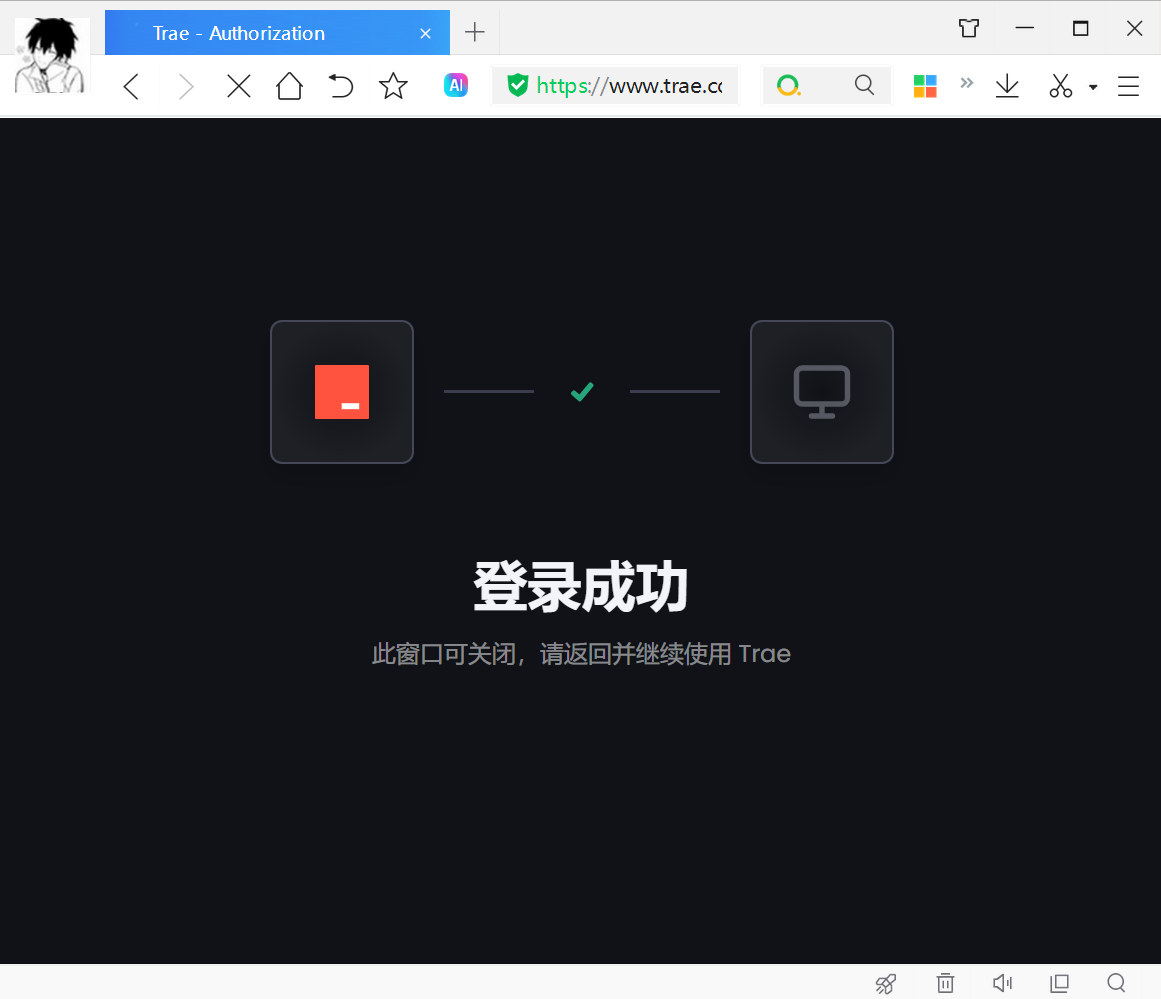
## **第五步：登录账号**

使用手机号或稀土掘金帐号登录 Trae。完成登录后，你才可以在 Trae 中使用 AI 服务。





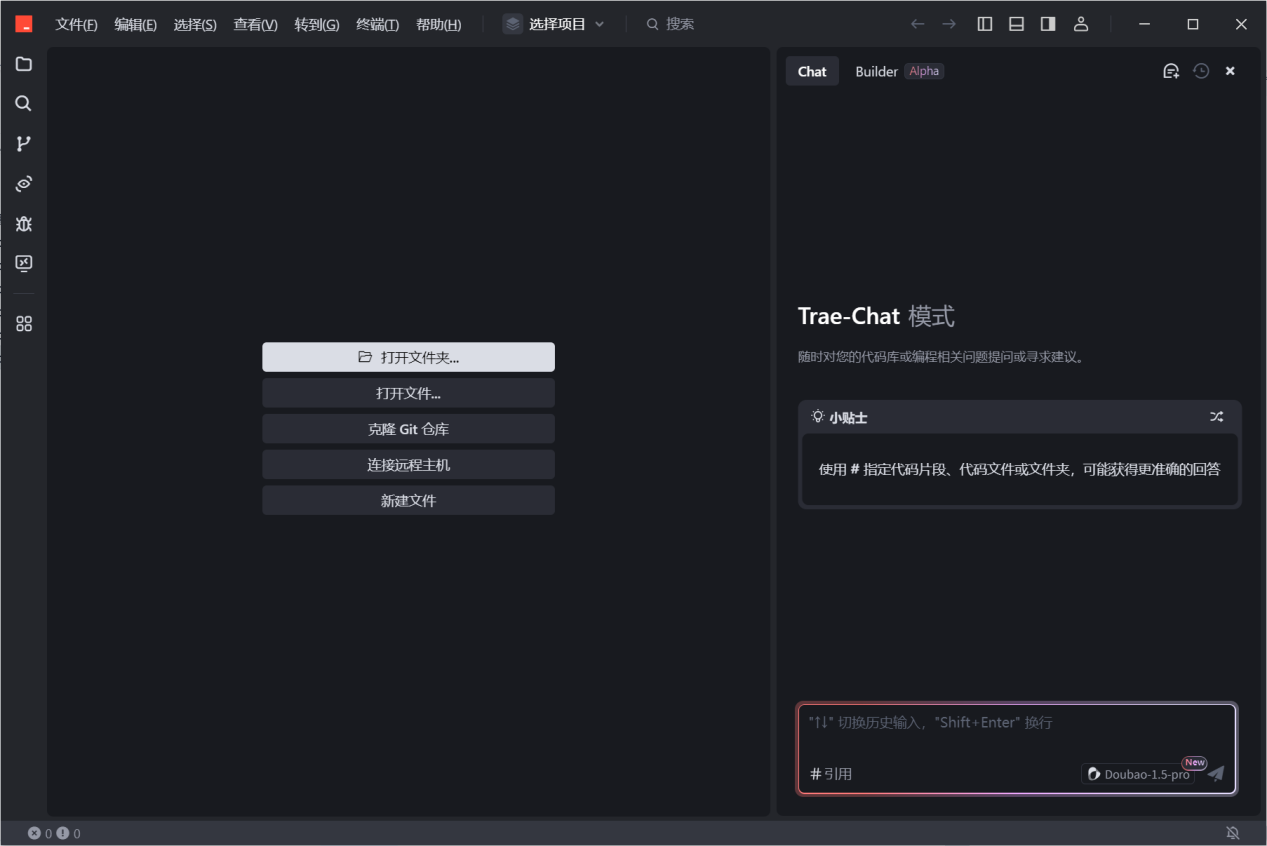




## 第六步：开始使用Trae进行开发

在 Trae 中，您可以通过以下两种方式之一开始开发项目：

* 打开现有文件夹或克隆现有 Git 仓库。
* 使用 AI 助手从0到1开始创建一个新项目。



# Trae基本使用

## 智能问答

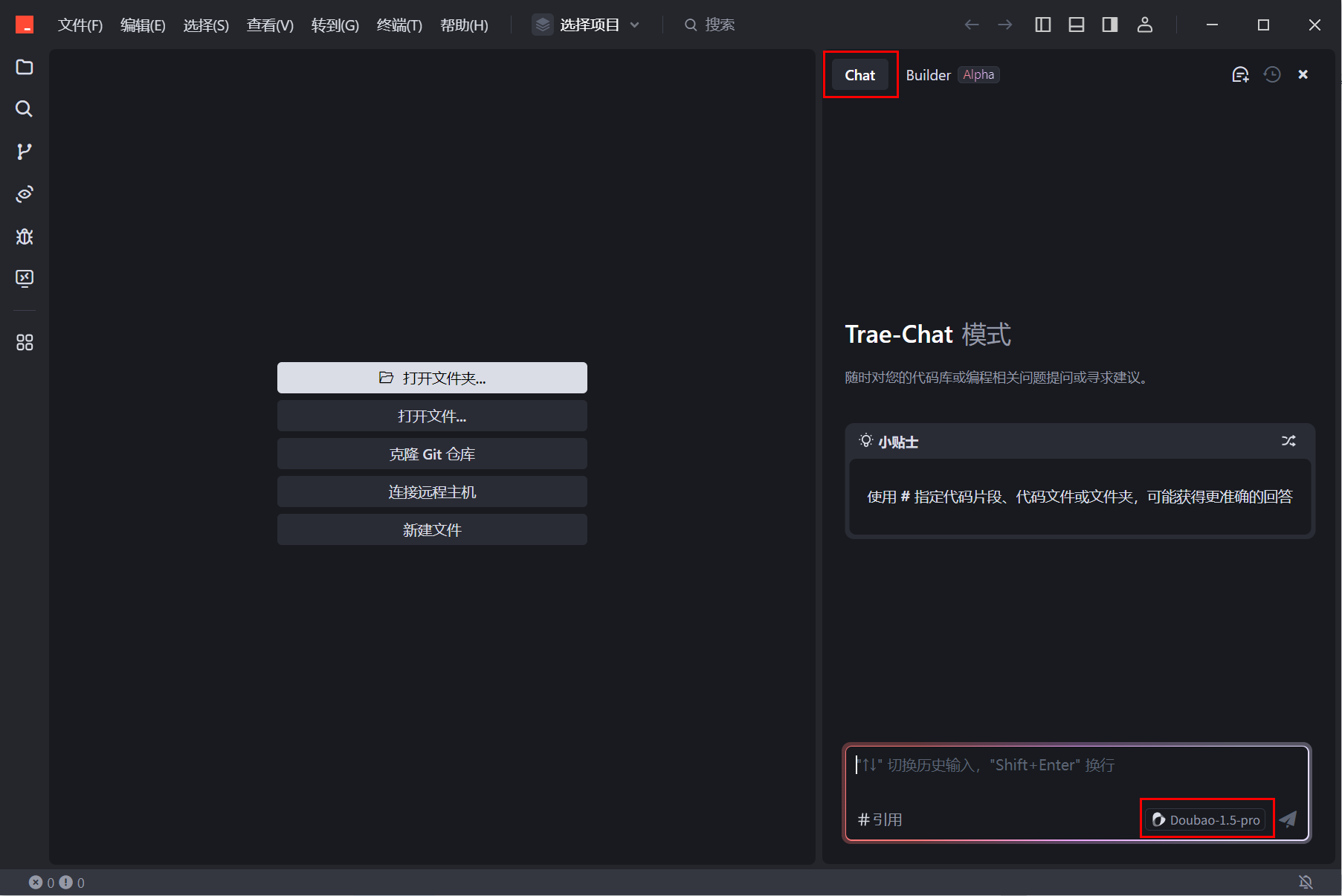
### 侧边对话（Chat 模式）

侧边对话（Chat 模式）是编码过程中的全能 AI 伙伴，可以用来回答编码问题、讲解代码仓库、生成代码片段、修复错误等。

进入 Trae 后，一般默认进入 Chat 模式。若非如此，可以参考以下步骤打开 Chat 模式。

1. 使用快捷键（macOS：Command + U；Windows：Ctrl + U）打开侧边对话框。
2. 在侧边对话框的左上角，点击 Chat。
3. 模型选择，Chat 模式默认使用 “Doubao-1.5-pro”。你可以点击模型名称，然后选择切换至 DeepSeek-Reasoner (R1) 或 DeepSeek-Chat (V3) 模型。

**提示：选择 DeepSeek-Reasoner (R1) 模型会自动启用其深度思考模式，并展示其思考过程。**



### 内嵌对话

Trae 提供了内嵌在代码编辑器中的内嵌对话（Inline Chat）。你能够在编码的过程中随时唤起内嵌对话。使用内嵌对话可以让你在与 AI 助手对话的同时更好地保持编码心流。

你可以通过以下两种方式唤起内嵌对话：

* 在编辑器内的光标处，使用快捷键（macOS：Command + I；Windows：Ctrl + I）。
* 在编辑器内，选中任意代码，然后使用快捷键（macOS：Command + I；Windows：Ctrl + I）或点击悬浮菜单中的 编辑 按钮。



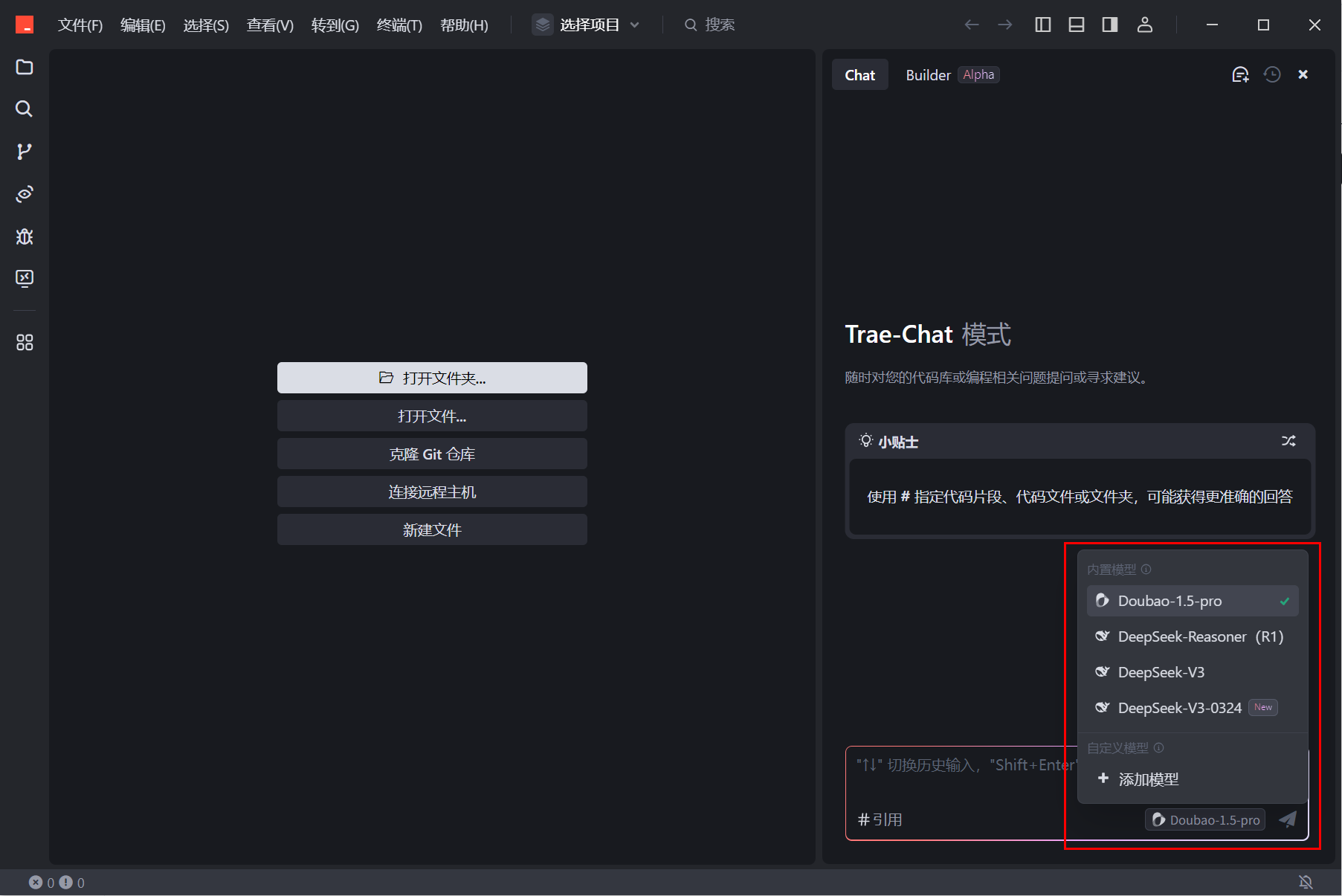
## 模型选择

Trae 预置了一系列业内表现比较出色的模型，你可以直接切换不同的模型进行使用。此外，Trae 还支持通过 API 密钥（API Key）接入自定义模型，从而满足个性化的需求。

Chat 和 Builder 模式可用的模型有所差异，详见下表。



在输入框的右下角，点击当前模型名称，打开模型列表，然后选择你想使用的模型。



## Builder 模式的工作流程

Builder 模式可以帮助你从 0 到 1 开发一个完整的项目。你可以将它无缝融入到项目构建的流程中。

在 Builder 模式下，AI 助手在回答时会根据需求调用不同的工具，包括分析代码文件的工具、编辑代码文件的工具、运行命令的工具等等。从而让回答更精确、更有效。

### Builder 模式的工作流程

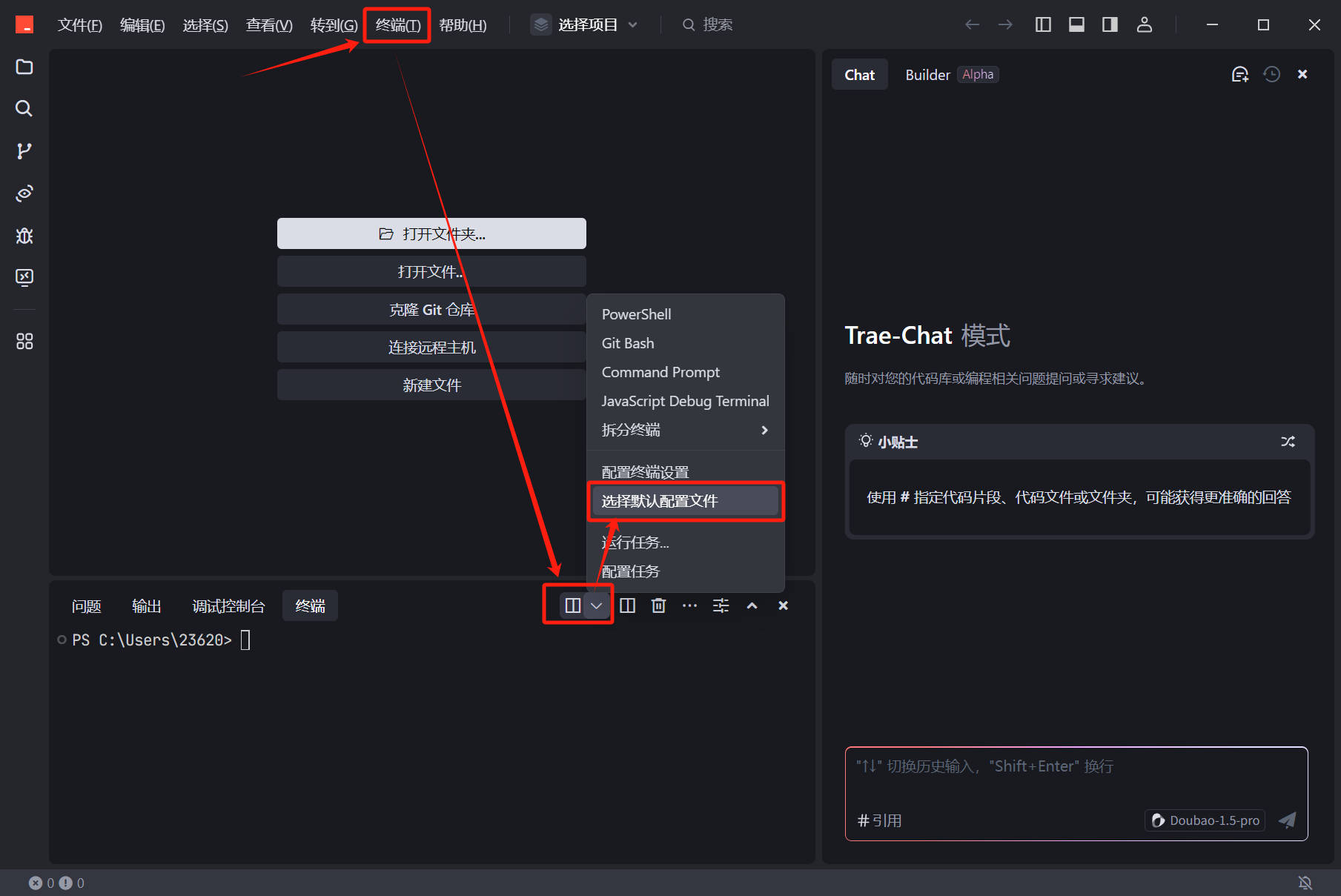
Builder 模式下，AI 助手会主动读取当前项目中的文件内容，拆解任务并逐步执行，包括：

* 提取相关上下文
* 创建或修改文件
* 生成并运行终端命令
* 分析命令运行状态

### 配置 PowerShell（Windows）

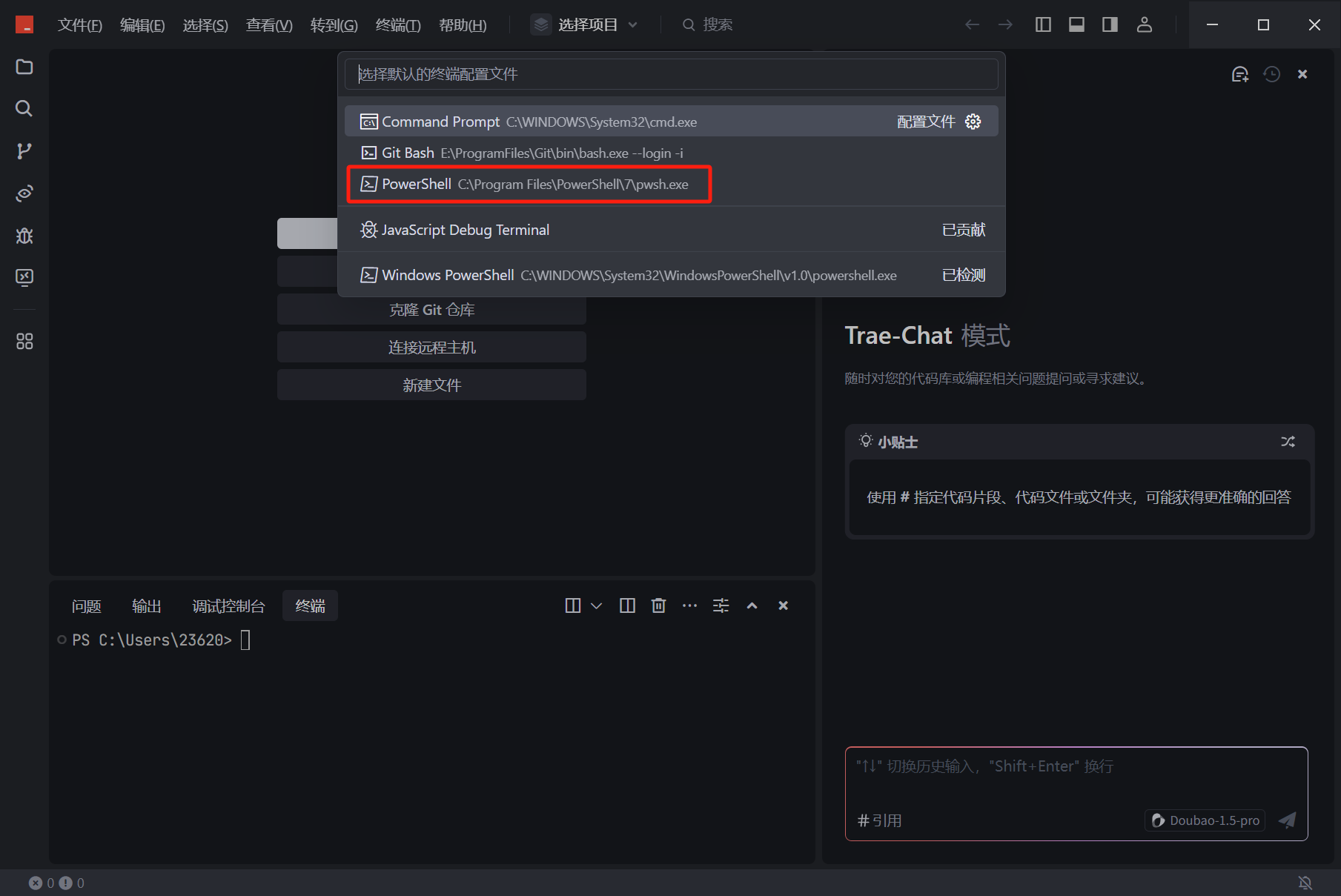
对于 Windows 操作系统，若需要使用 Builder 模式，则必须为 Trae 配置 PowerShell 6 或更高版本。若当前非此配置，参考以下步骤完成配置：

1. 安装 PowerShell 6 或更高版本。[安装说明参考此文档](https://learn.microsoft.com/en-us/powershell/scripting/install/installing-powershell-on-windows?view=powershell-7.5)。
2. 返回 Trae 并打开 终端 面板。
3. 点击右上方区域的 + > 选择默认配置文件。



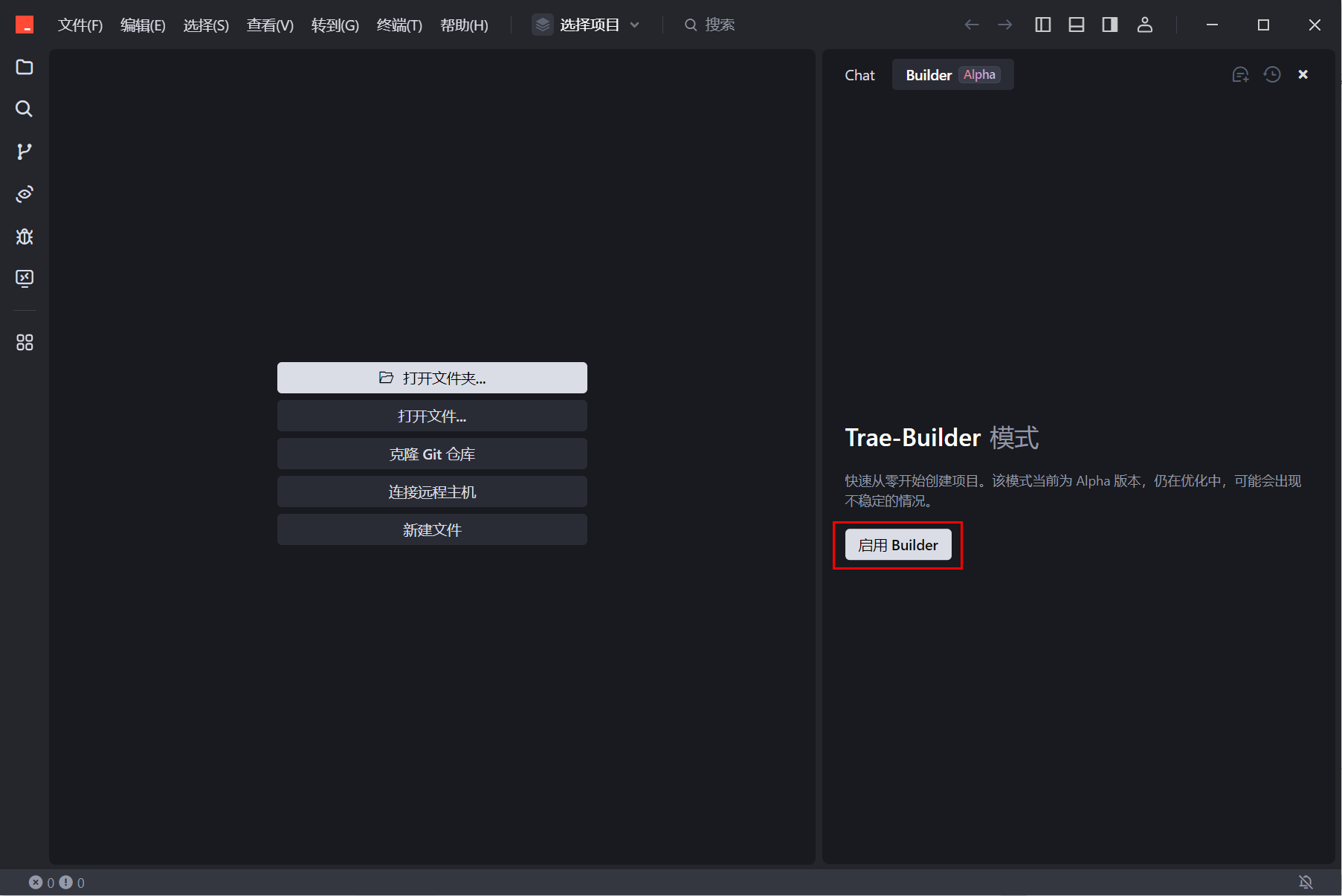
1. 界面上方显示配置文件选择列表，选择带有 \PowerShell\{版本号}\xxx 字样的配置文件（一般会自动检测到 ）。

**注意：选择的{版本号} 大于等于 6。勿选择系统内置的旧版本 PowerShell，例如图中的 Windows PowerShell。**

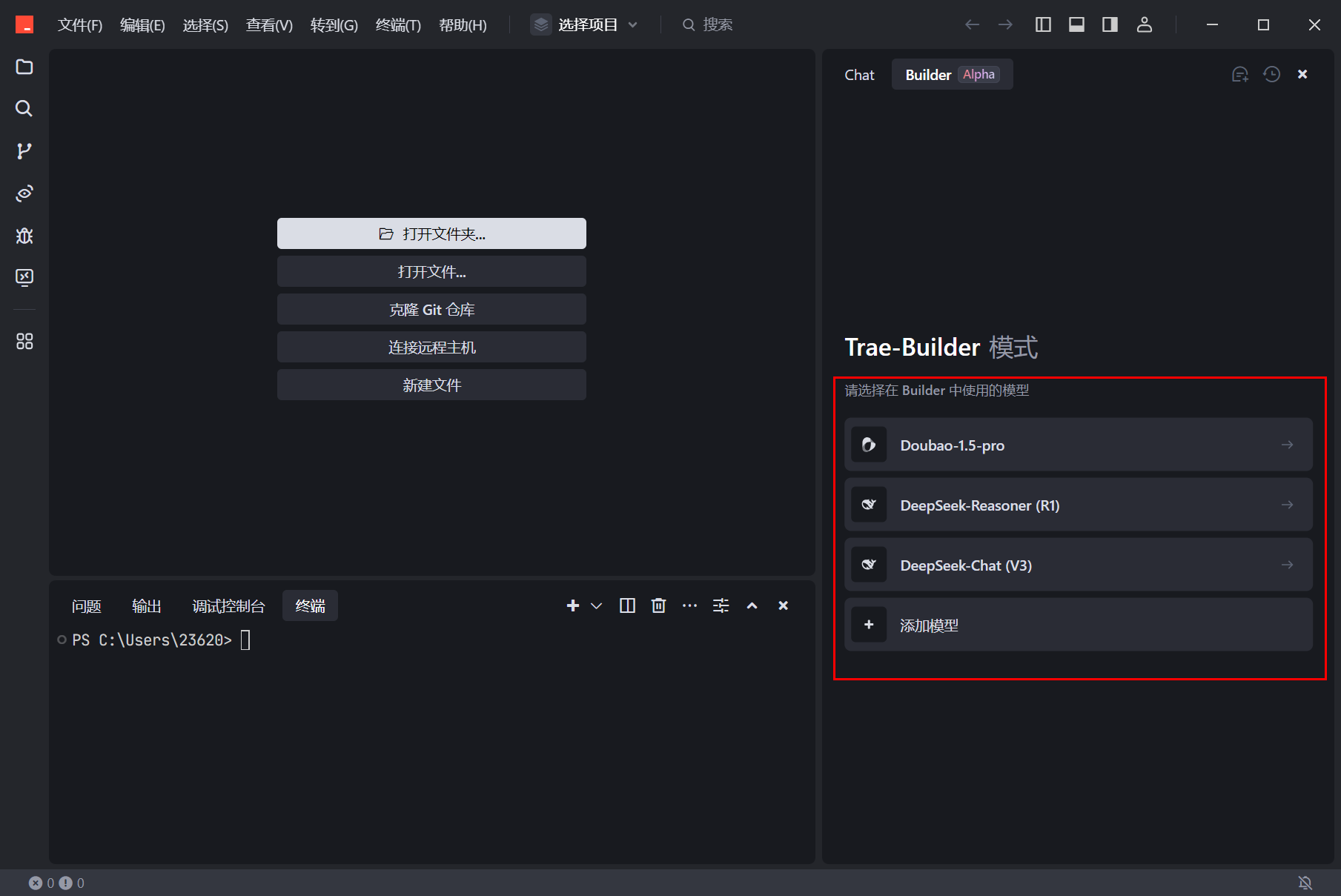


### 启用 Builder 模式

1. 使用快捷键（macOS：Command + U；Windows：Ctrl + U）打开侧边对话框。
2. 在侧边对话框的左上角，点击 Builder。
3. 在 Builder 对话框中，点击 启用 Builder 按钮。



1. 选择你想使用的大语言模型。



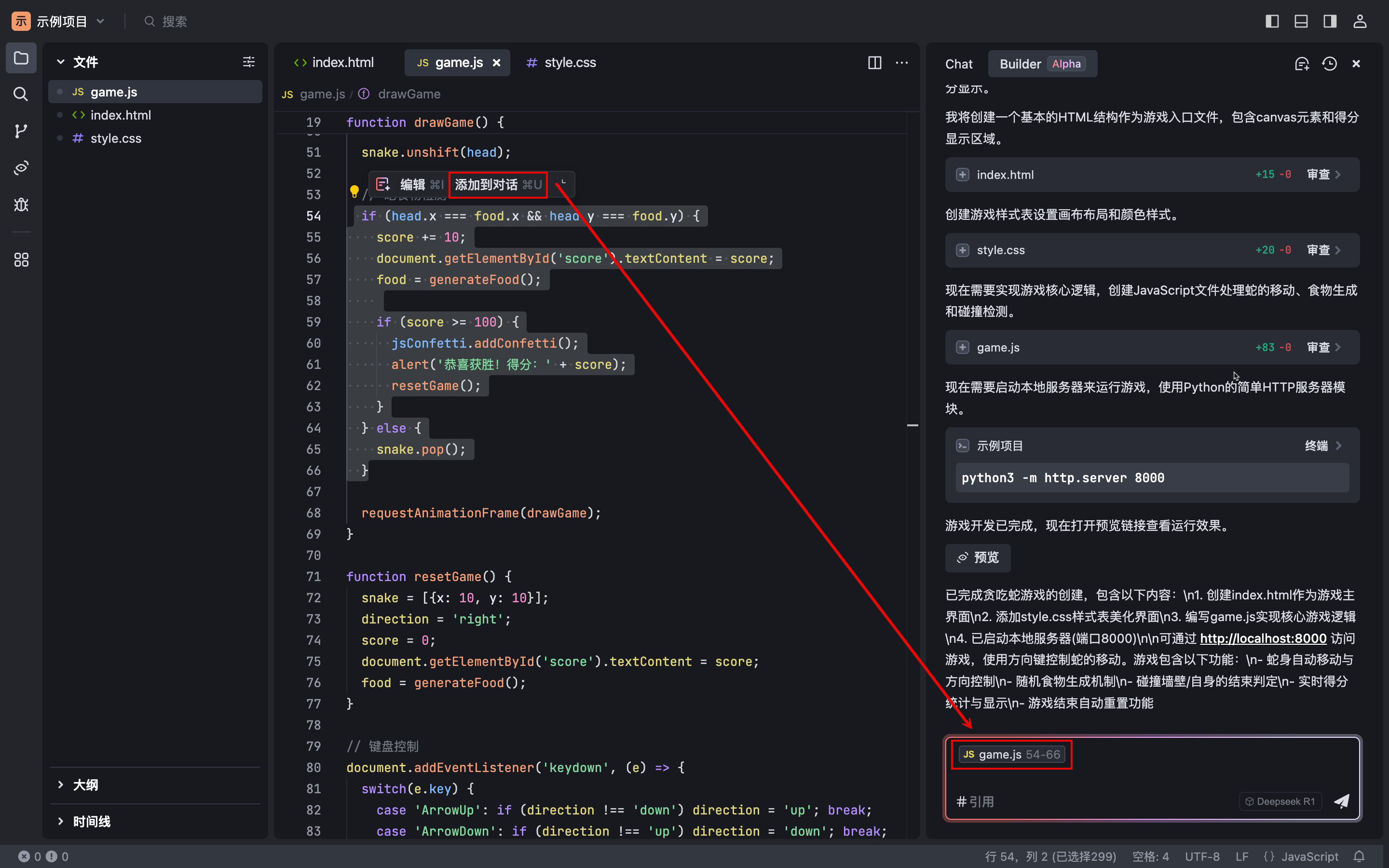
## 上下文：提升 AI 助手响应精准度的三种方法

AI 助手对话时，你可以指定 Trae 中的内容（包括代码、文件、文件夹、工作空间）作为上下文供 AI 助手阅读和理解，从而确保 AI 助手的答复更符合你的要求。本文介绍指定上下文的三种方式。

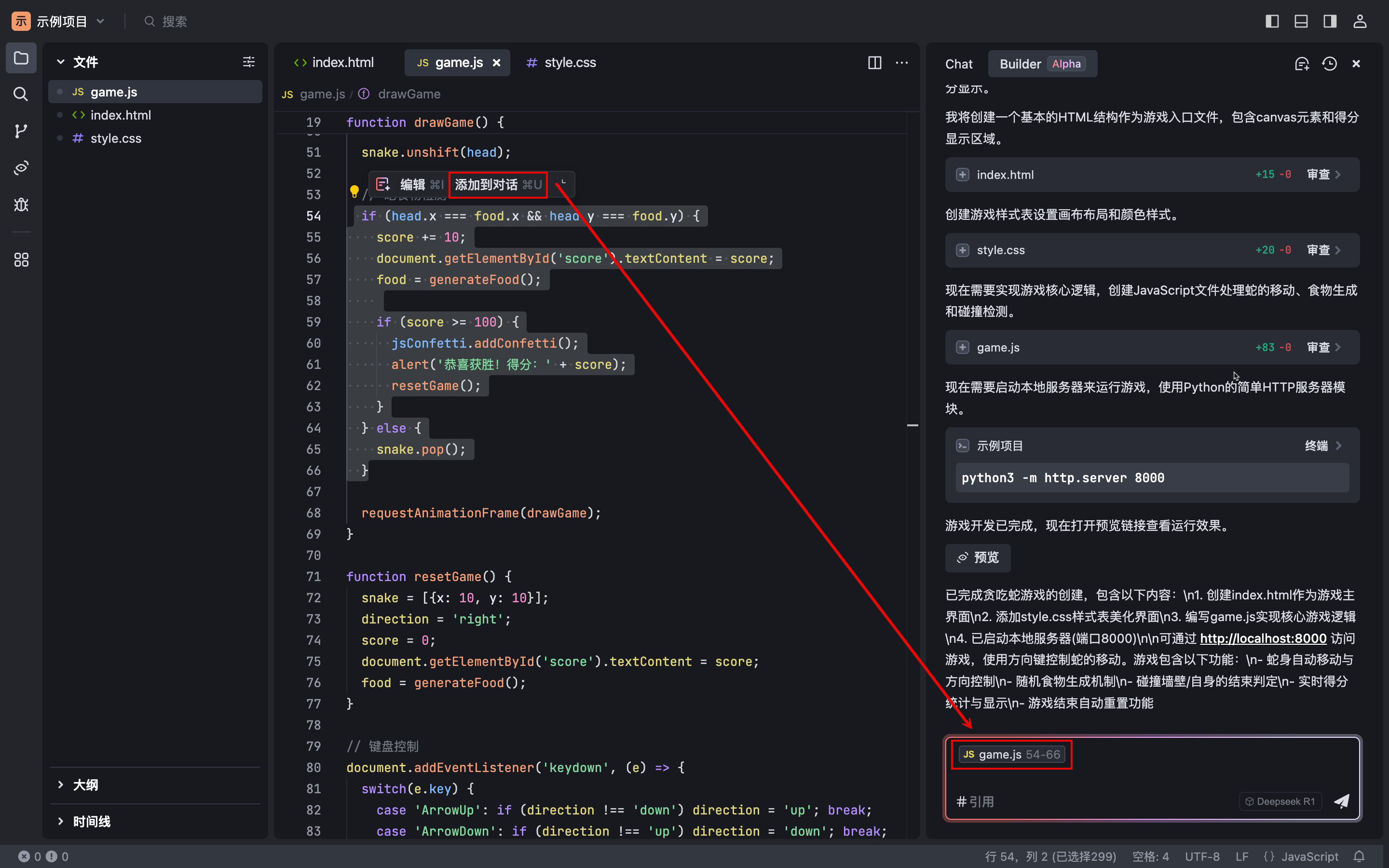
**方式一：将编辑器内的内容作为上下文**

当你的编辑器中存在正在编辑的代码文件时，AI 助手默认能够看到当前文件。你可以直接向 AI 助手提问与当前文件相关的问题。若期望对文件中的某一段代码进行提问，使用以下步骤：

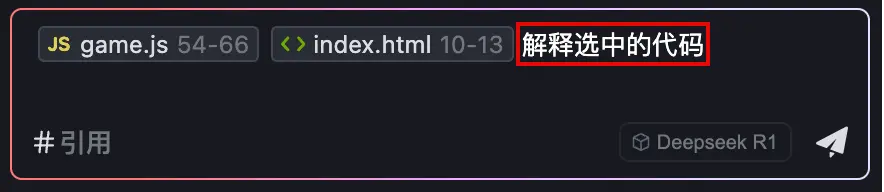
1. 选中代码。
2. 点击悬浮菜单中的 添加到对话 按钮，将选中的内容作为上下文添加至侧边对话框。指定的上下文会显示在侧边对话底部的输入框。以下图为例，输入框内显示所选内容所属的文件名称，以及所选的代码行编号。



1. (可选) 继续添加编辑器中的其他内容片段，或同时添加其他来源的上下文。



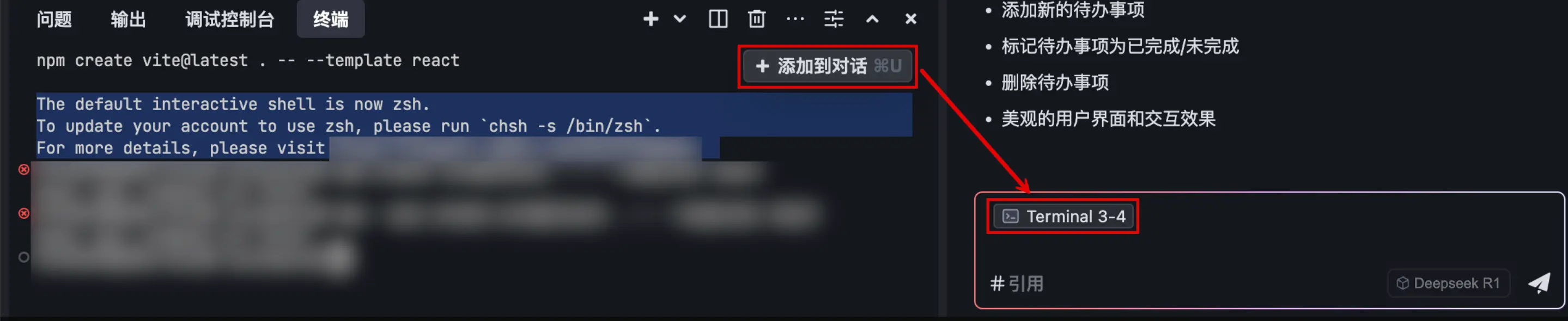
1. 在上下文旁，输入你的问题，然后发送给 AI 助手。



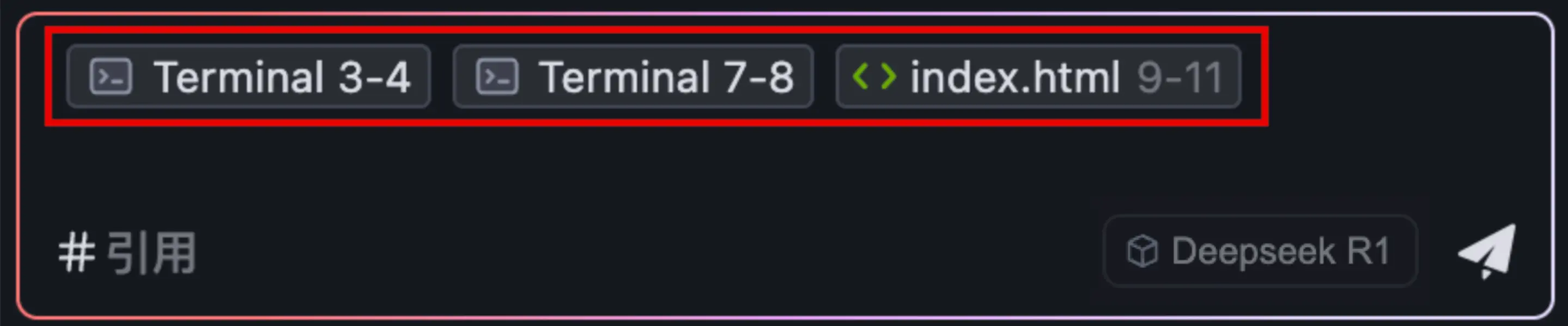
**方式二：将终端中的内容作为上下文**

若你希望对终端中的输出内容进行提问（如帮助你修复报错），使用以下步骤：

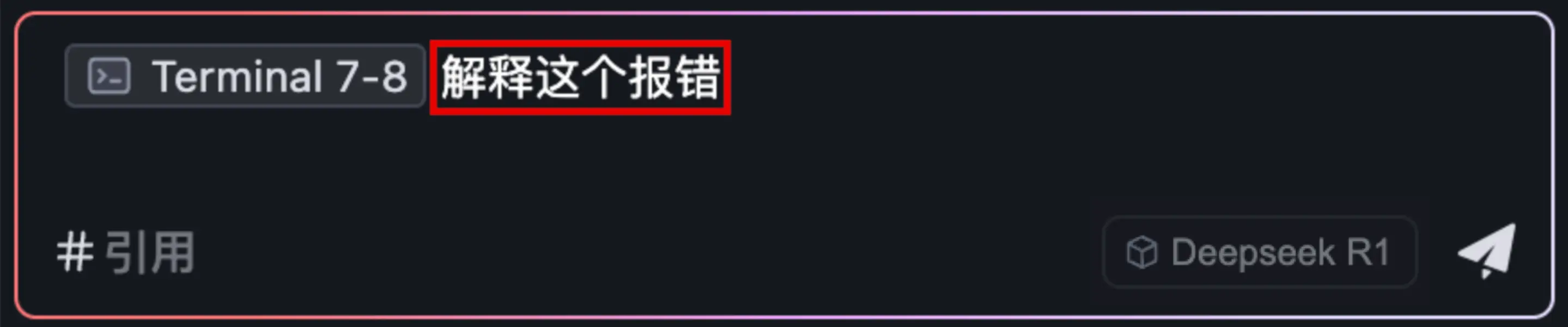
1. 在终端中，点击输出内容片段。
2. 在内容片段区域的右上角，点击 添加到对话 按钮，将选中的内容作为上下文添加至侧边对话框。  
   指定的上下文会显示在侧边对话底部的输入框。以下图为例，输入框内显示上下文的来源以及行号。



1. (可选) 继续添加终端中的其他内容片段，或同时添加其他来源的上下文。



1. 在上下文旁，输入你的问题，然后发送给 AI 助手。



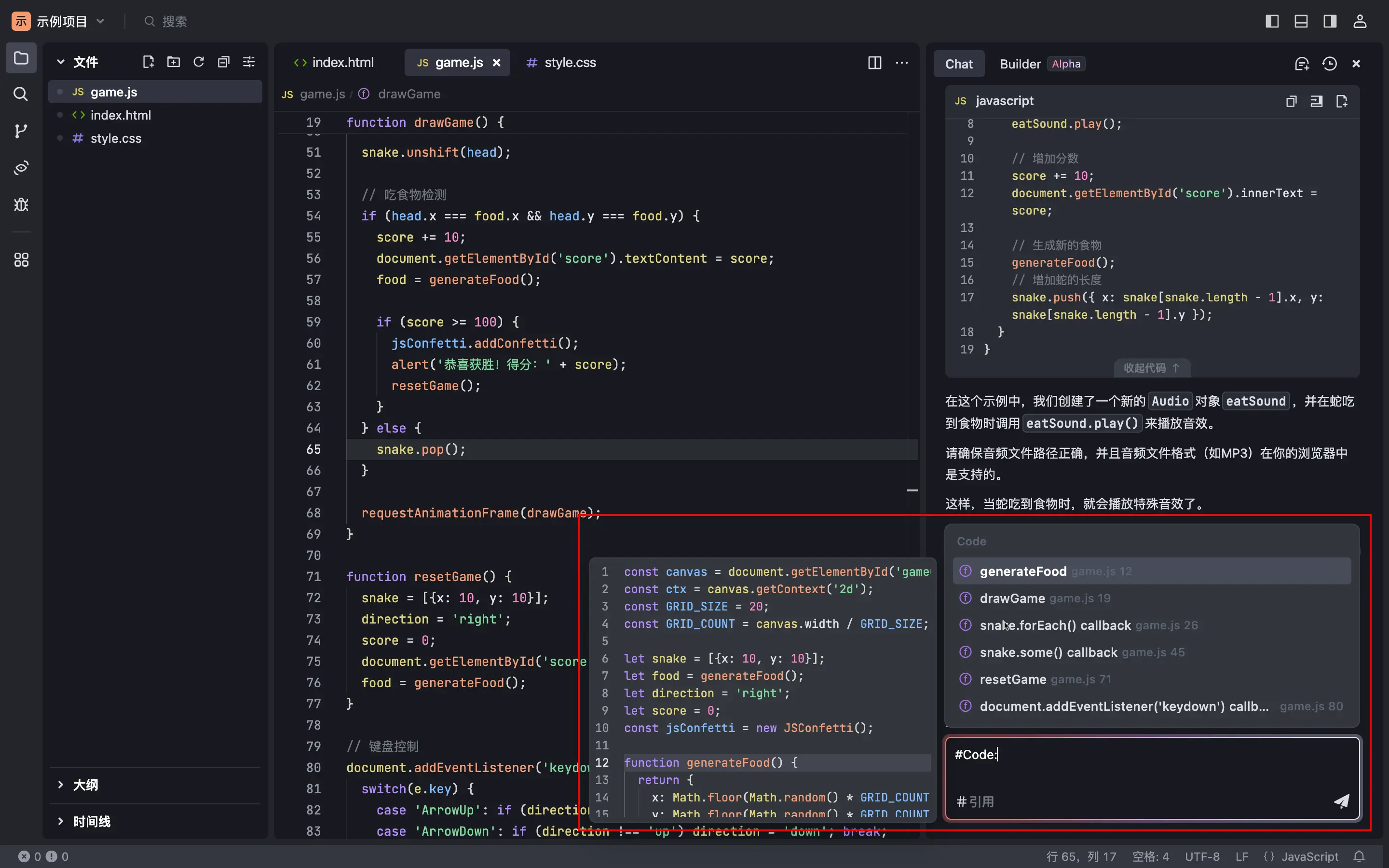
**方式三：使用 # 键添加上下文**

在侧边对话的输入框中，你可以通过 # 符号添加多种类的上下文，包括代码、文件、文件夹和工作区。通常情况下，列表中将展示与编辑器中当前打开文件相关的内容作为推荐的上下文，但你仍然可以自行搜索所需的上下文并将其添加到输入框中。基于你的问题，你可以组合添加各种来源的相关上下文（例如同时添加代码和文件）。

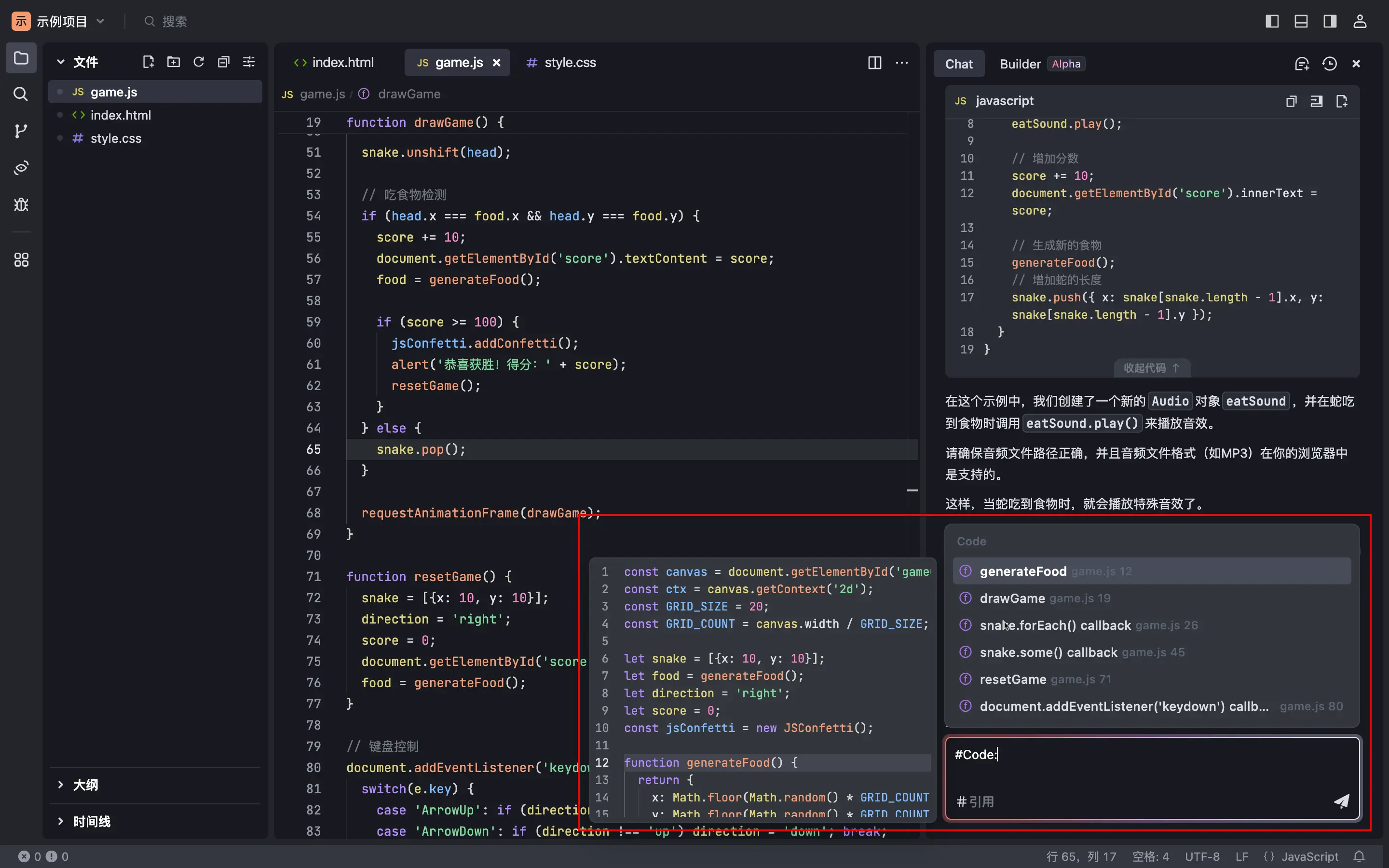
#Code

通过 #Code，你可以将函数或类的相关代码作为与 AI 助手对话的上下文。列表中默认展示当前编辑器内打开的文件中的函数或类。选择前，你可以预览列表中推荐的函数或类的相关代码。若推荐的内容非你所需，你可以通过关键词搜索所需的函数或类。

1. 在输入框中输入 #，或直接点击输入框左下角的 # 引用 按钮。 输入框上方显示上下文类型选择列表。 Trae 上下文#Code 1
2. 在列表中选择 Code（或在 # 符号后手动输入 Code），然后按下回车键。列表将展示编辑器中当前打开的文件中存在的函数和类。将鼠标悬浮在列表中的某个条目后，左侧会展示该函数或类的代码内容，供你预览。



1. 若推荐的函数和类非你所需，在 #Code: 后输入你想要的函数或类的名称或关键词。 AI 助手将在项目内搜索相关函数和类，并展示在列表中，你可以进行预览。



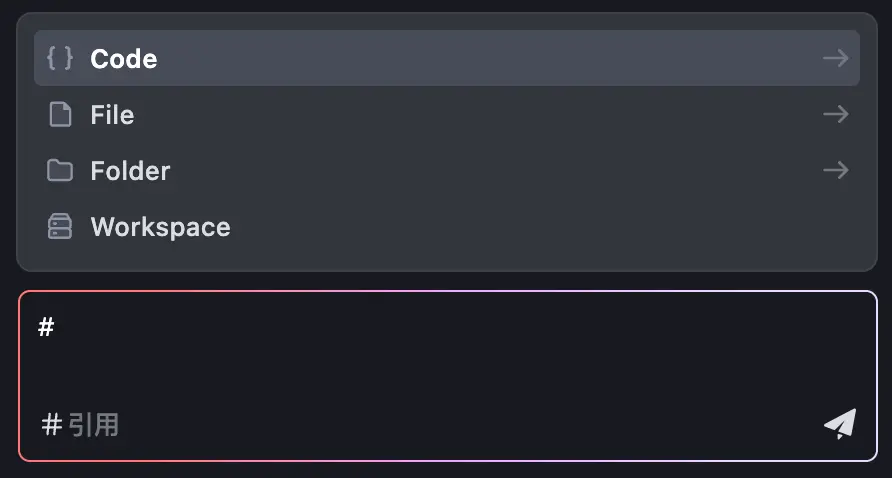
1. 从列表中选择需指定为上下文的函数或类。
2. 在输入框的代码标识后，输入你的问题并发送。



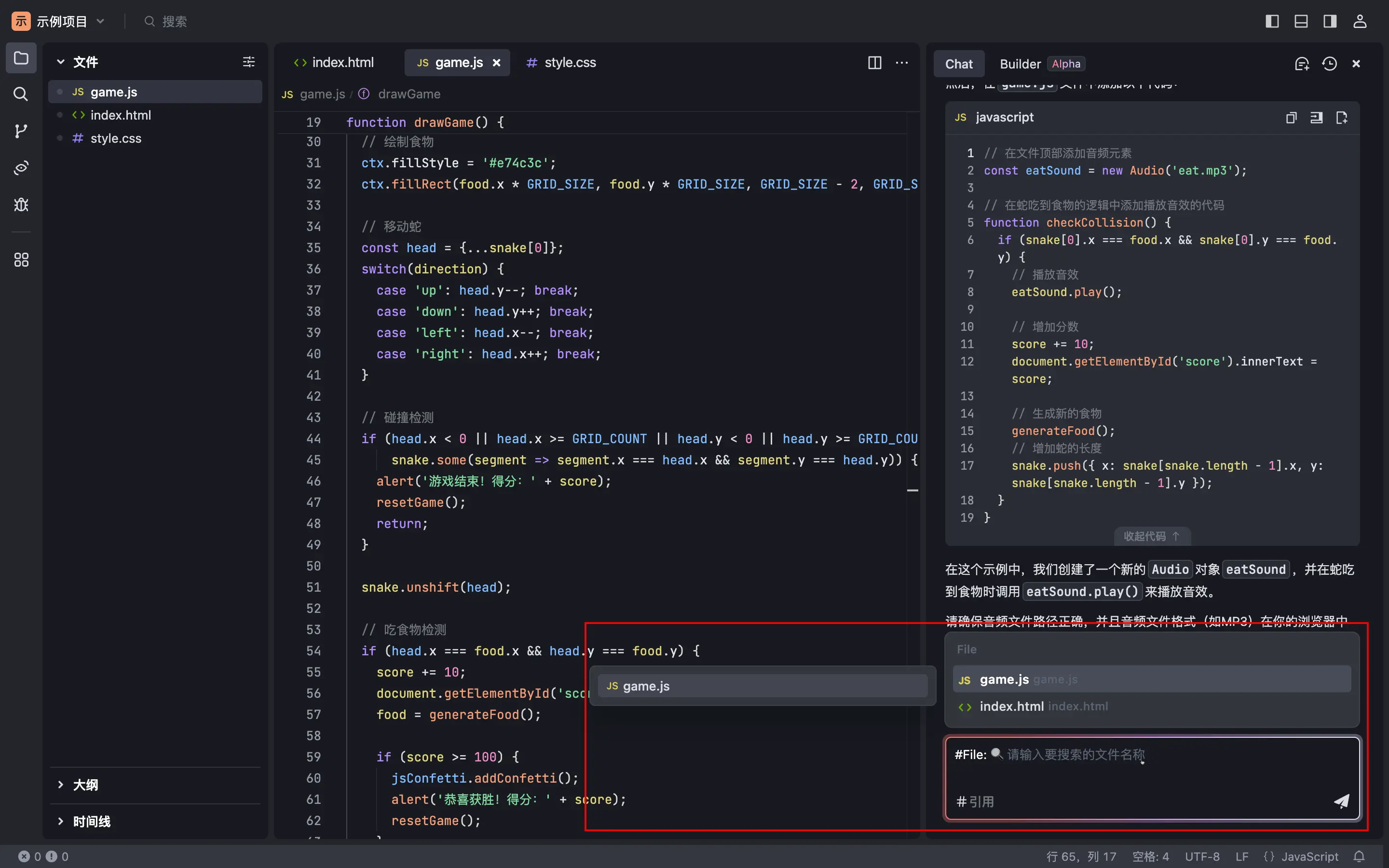
**#File**

通过 #File，你可以将指定文件中的所有内容作为与 AI 助手对话的上下文。列表中默认展示近期在编辑器中打开过的文件。你可以预览这些文件所在的目录，以免因存在同名文件而导致错选。若展示的文件非你所需，你可以通过关键词搜索所需文件。

1. 在输入框中输入 #，或直接点击输入框左下角的 # 引用 按钮。 输入框上方显示上下文类型选择列表。



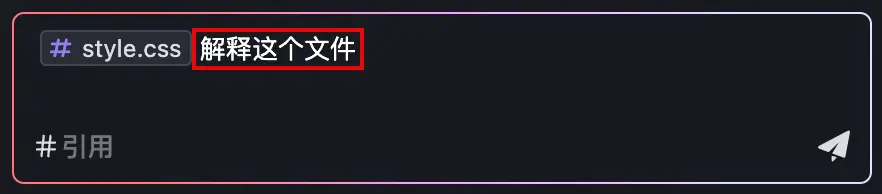
1. 在列表中选择 File（或在 # 符号后手动输入 File），然后按下回车键。列表展示近期在编辑器中打开过的文件。将鼠标悬浮在列表中的某个条目后，左侧会展示该文件所在的目录。



1. 若推荐的文件非你所需，在 #File: 后输入你想要的文件的名称或关键词。AI 助手将在项目内搜索相关文件并展示在列表中。



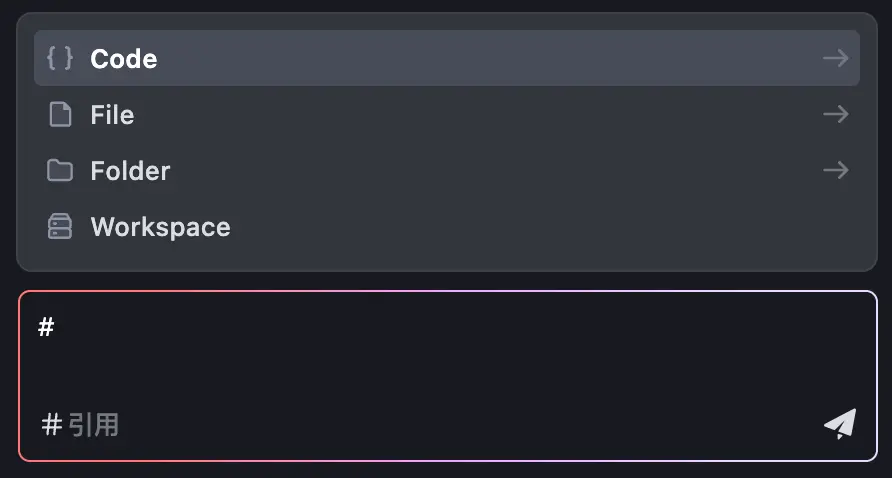
1. 从列表中选择需指定为上下文的文件。
2. 在输入框的文件标识后，输入你的问题并发送。



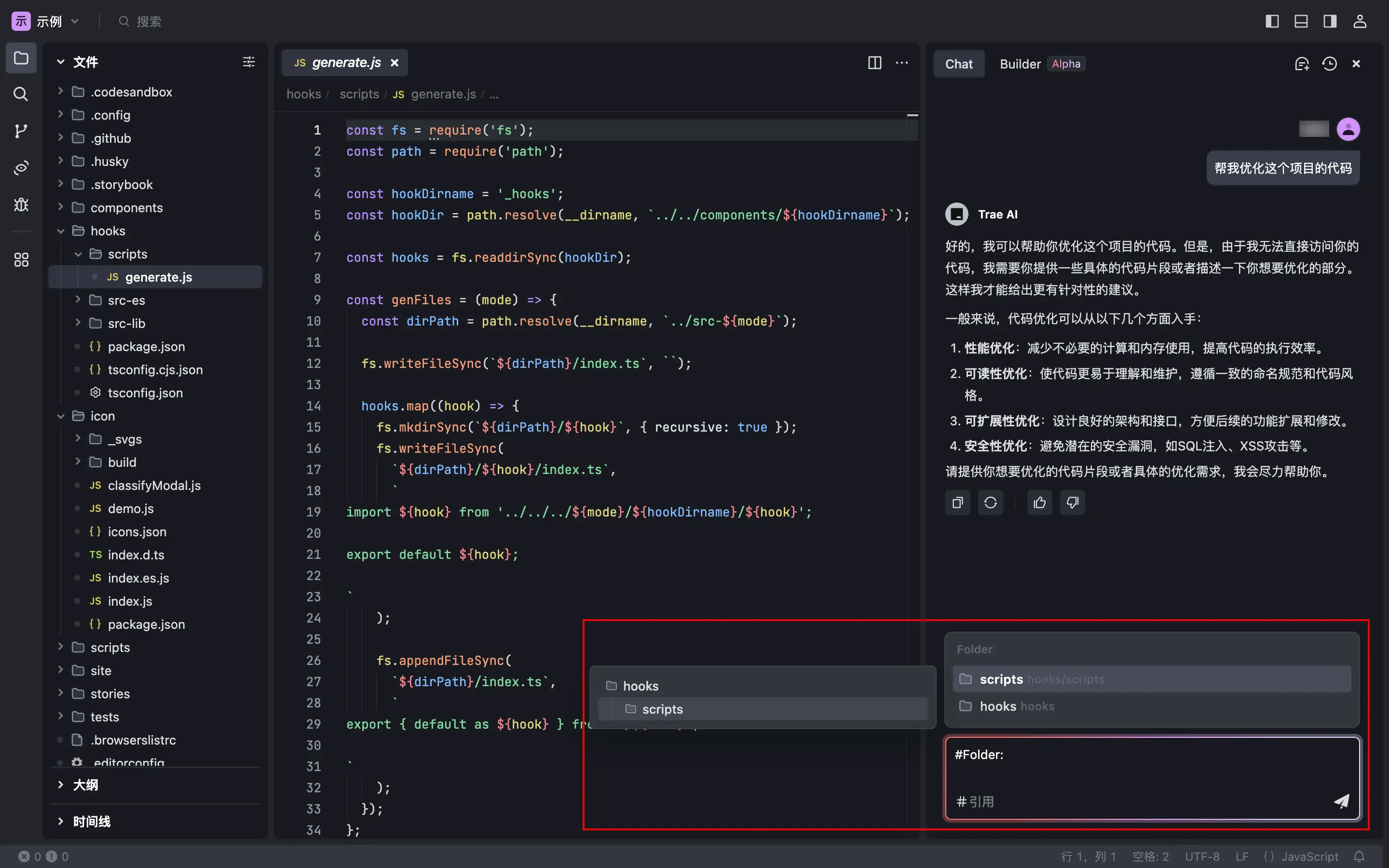
**#Folder**

通过 #Folder，你可以将指定文件夹中的所有内容作为与 AI 助手对话的上下文。列表中默认展示与编辑器中当前所打开文件相关的文件夹。你可以预览这些文件夹所在的目录，以免因存在同名文件夹而导致错选。若展示的文件夹非你所需，你可以通过关键词搜索所需文件夹。

1. 在输入框中输入 #，或直接点击输入框左下角的 # 引用 按钮。  
   输入框上方显示上下文类型选择列表。



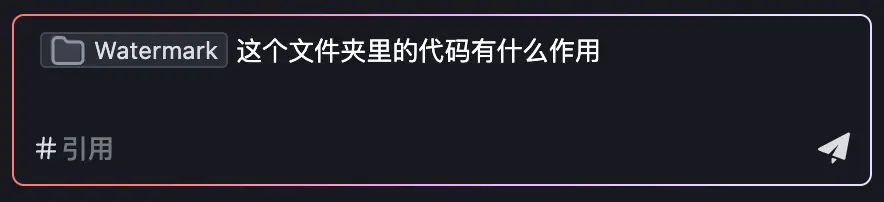
1. 在列表中选择 Folder（或在 # 符号后手动输入 Folder），然后按下回车键。 列表展示与编辑器中当前所打开文件相关的文件夹。将鼠标悬浮在列表中的某个条目后，左侧会展示该文件夹所在的目录。



1. 若推荐的文件夹非你所需，在 #Folder: 后输入你想要的文件夹的名称或关键词。 AI 助手将在项目内搜索相关文件夹并展示在列表中。



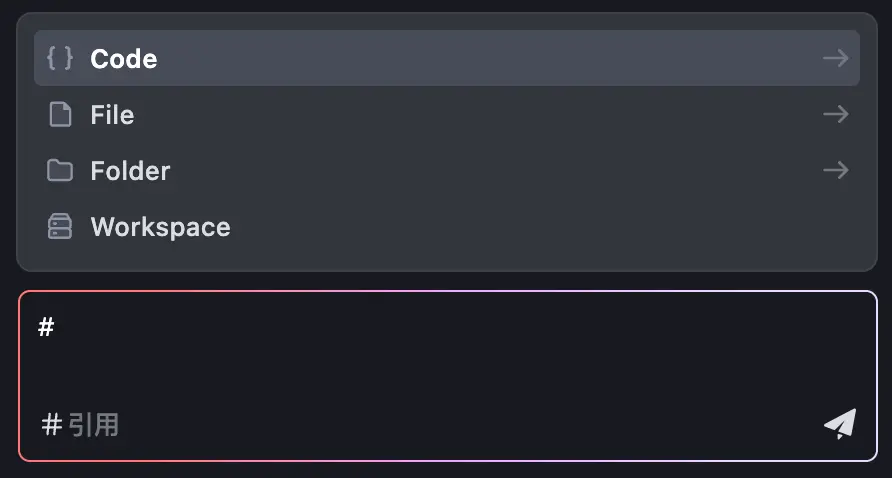
1. 从列表中选择需指定为上下文的文件夹。
2. 在输入框的文件夹标识后，输入你的问题并发送。



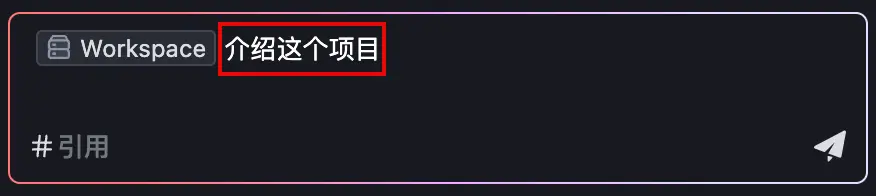
**#Workspace**

你可以使用 #Workspace 来向 AI 助手提出有关整个工作空间的问题，AI 助手将自动在工作空间中查找与你的问题最相关的内容作为上下文进行回答。遇到不熟悉的项目时，该功能可以帮助你全局了解该项目所实现的功能和代码，从而使你快速上手该项目的开发。

1. 在输入框中输入 #，或直接点击输入框左下角的 # 引用 按钮。 输入框上方显示上下文类型选择列表。



1. 在列表中选择 Workspace（或在 # 符号后手动输入 Workspace）。
2. 在输入框的 Workspace 标识后，输入你的问题并发送。

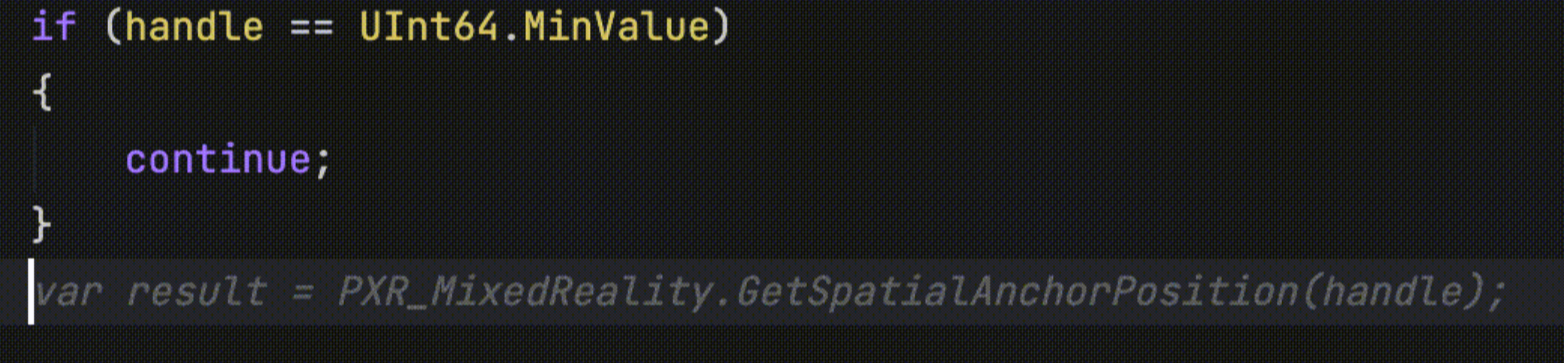


## 代码自动补全

Trae 内置原生的 AI 代码补全功能，无需手动开启。在编辑器内编写代码时，AI 助手会阅读并理解已编写的代码，然后自动续写代码。

**自动补全代码**

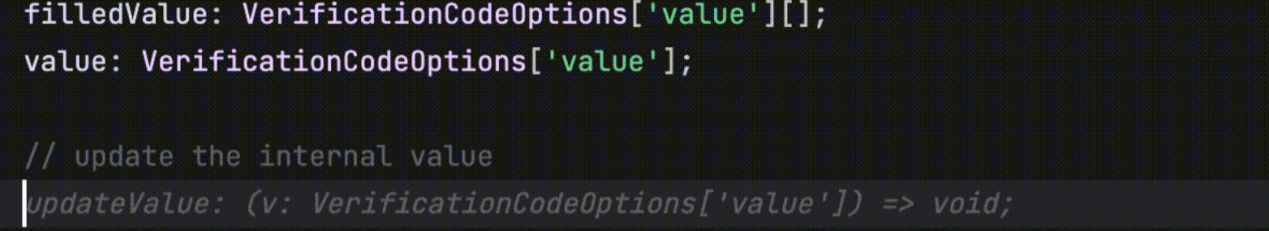
在光标所在位置，敲击回车键换行，AI 助手会阅读并理解当前代码，然后自动补全后续代码。



当鼠标悬停在自动补全的代码区域时，你可以按下 Tab 键一次性插入所有自动补全的代码，或者使用 Ctrl + → 快捷键逐字添加代码。

**基于注释的代码补全**

在代码中需要的位置添加注释，然后按下 Enter 键创建新行。AI 助手会读取并理解注释内容，然后生成相关的代码。



当鼠标悬停在自动补全的代码区域时，你可以按下 Tab 键一次性插入所有自动补全的代码，或者使用 Ctrl + → 快捷键逐字添加代码。

# 场景演示：一句话生成贪吃蛇游戏

切换到“Builder”模式。

用中文描述需求就能自动生成代码，可以几分钟搞定贪吃蛇、扫雷等小游戏，分析数据等重复性工作。

提出需求，如：使用 web 技术栈生成一个贪吃蛇游戏/帮我生成一个贪吃蛇游戏。

