

The background is black with several abstract, colorful elements. On the left, there are overlapping semi-circular shapes in shades of blue and cyan. On the right, there is a stylized flower-like shape composed of several petals in shades of purple, pink, and red. At the bottom, there are more overlapping semi-circular shapes in shades of blue and purple.

**UNIVERSE 23 RECAP**

**東京**

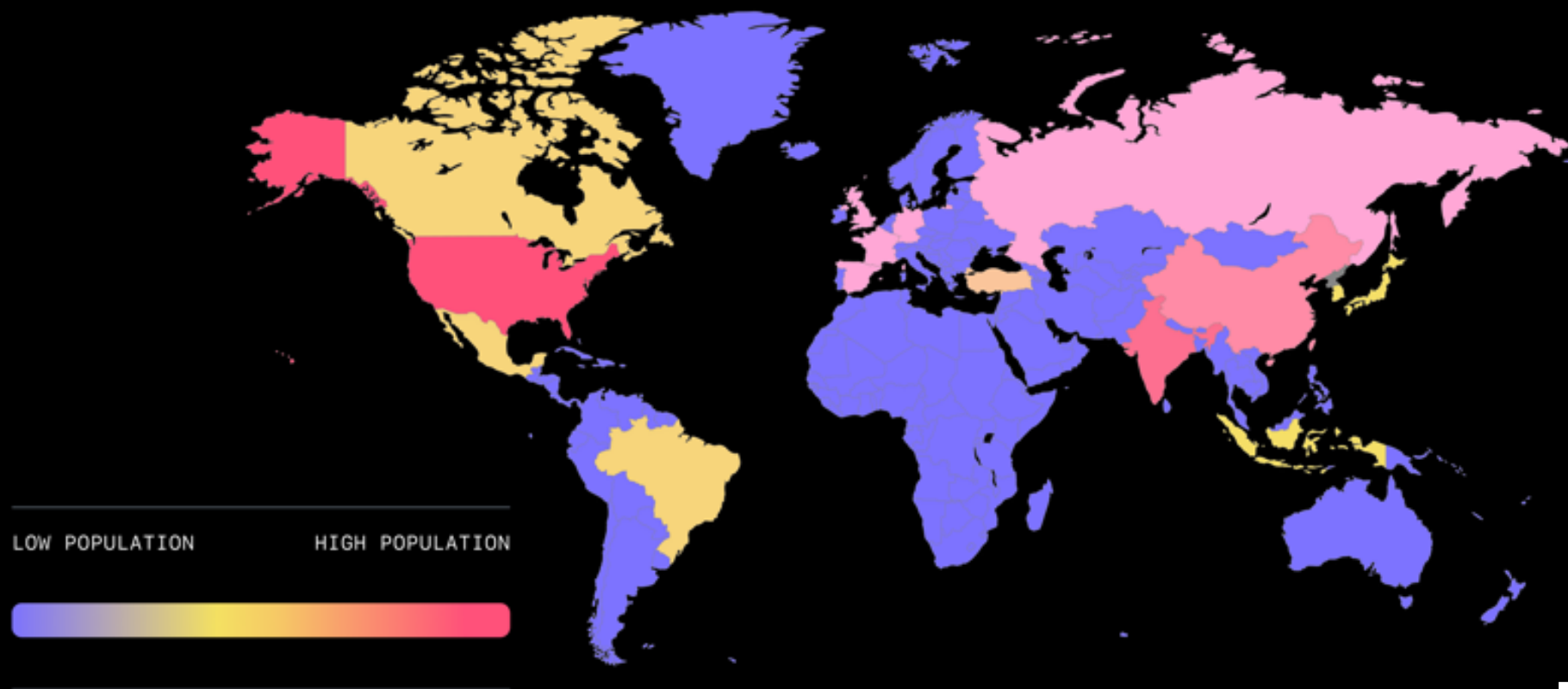


TOKYO

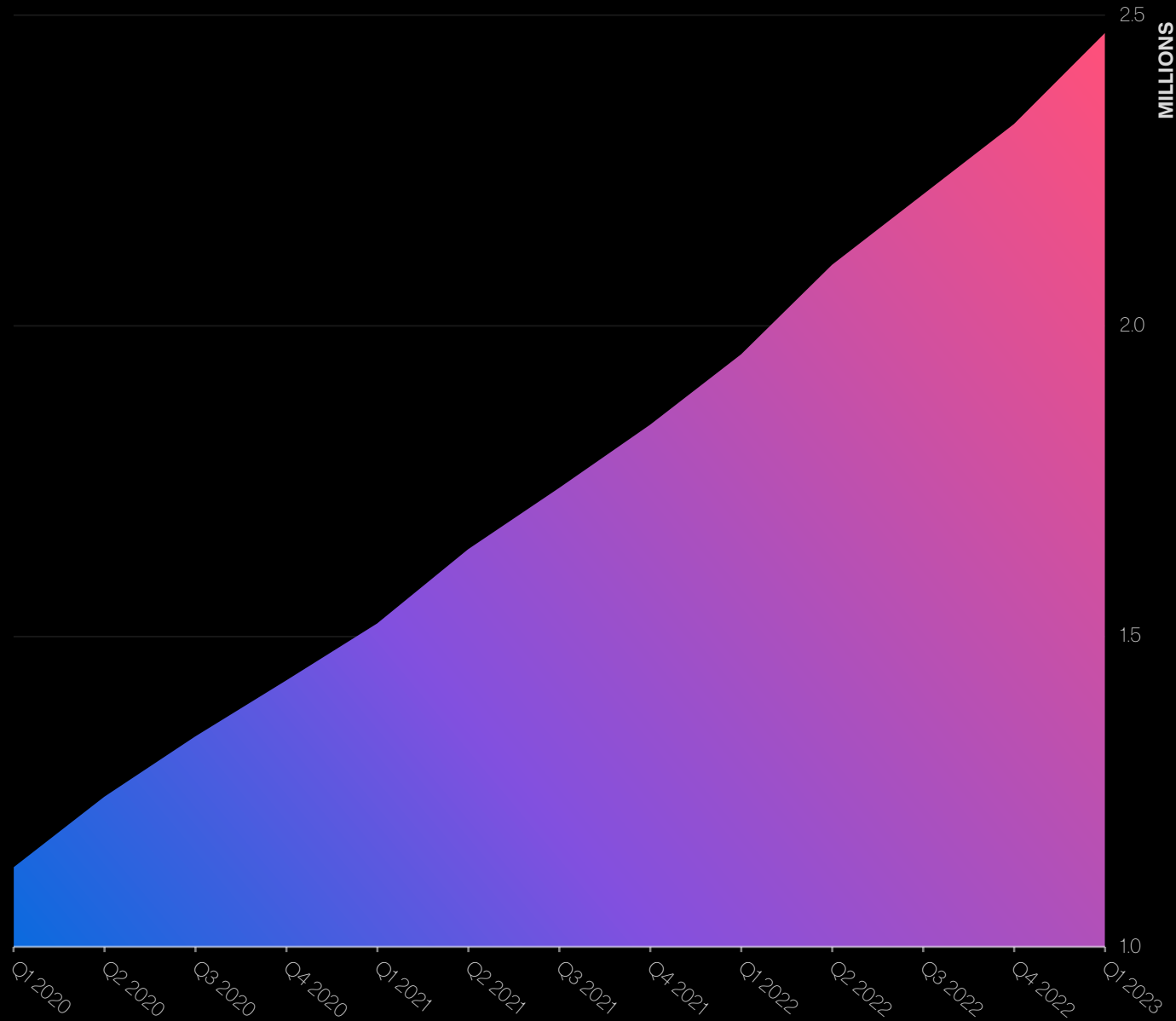
# Home for Developers



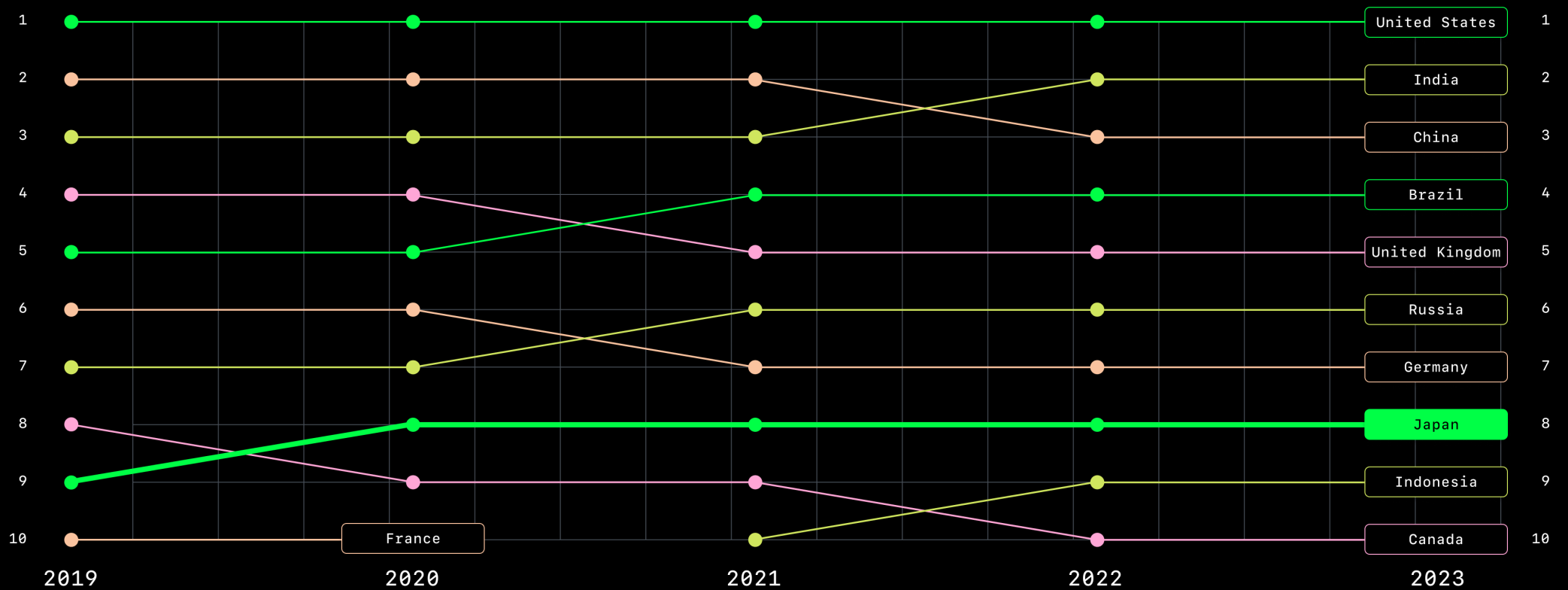
# Geographic heatmap of developer populations around the world



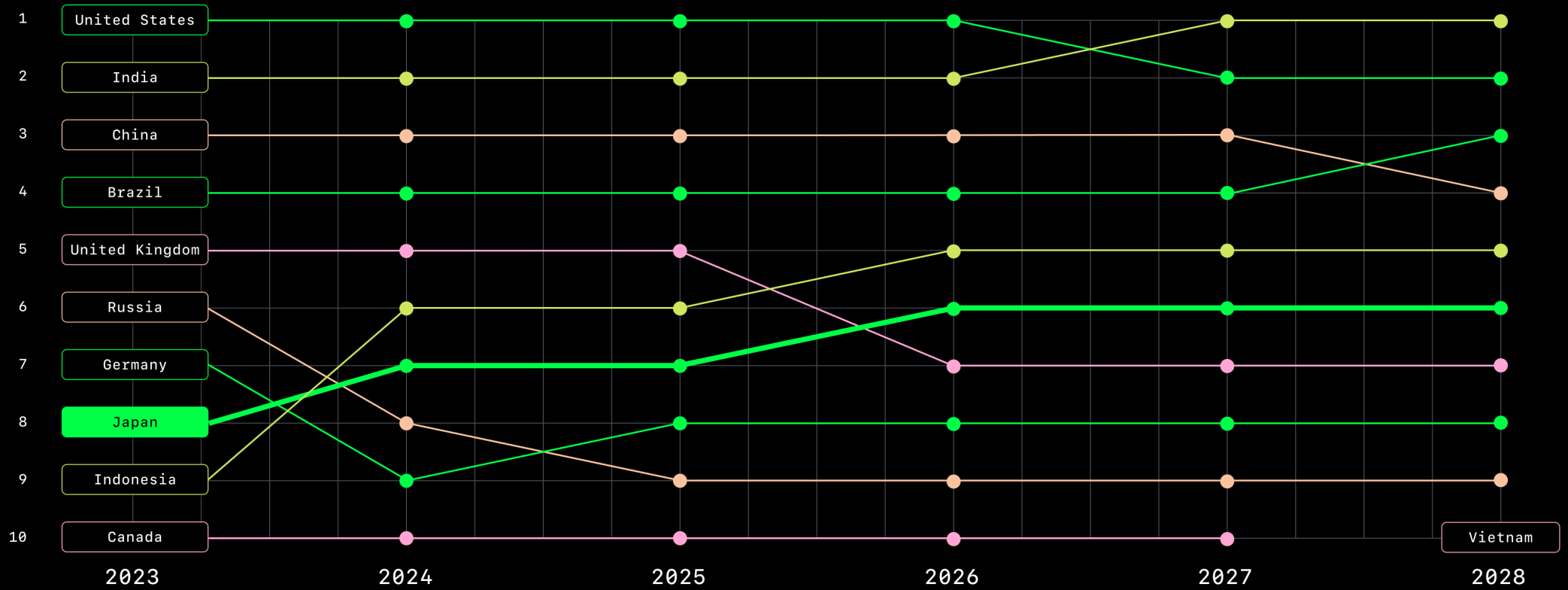
# 2.5 Million Japanese Developers on GitHub



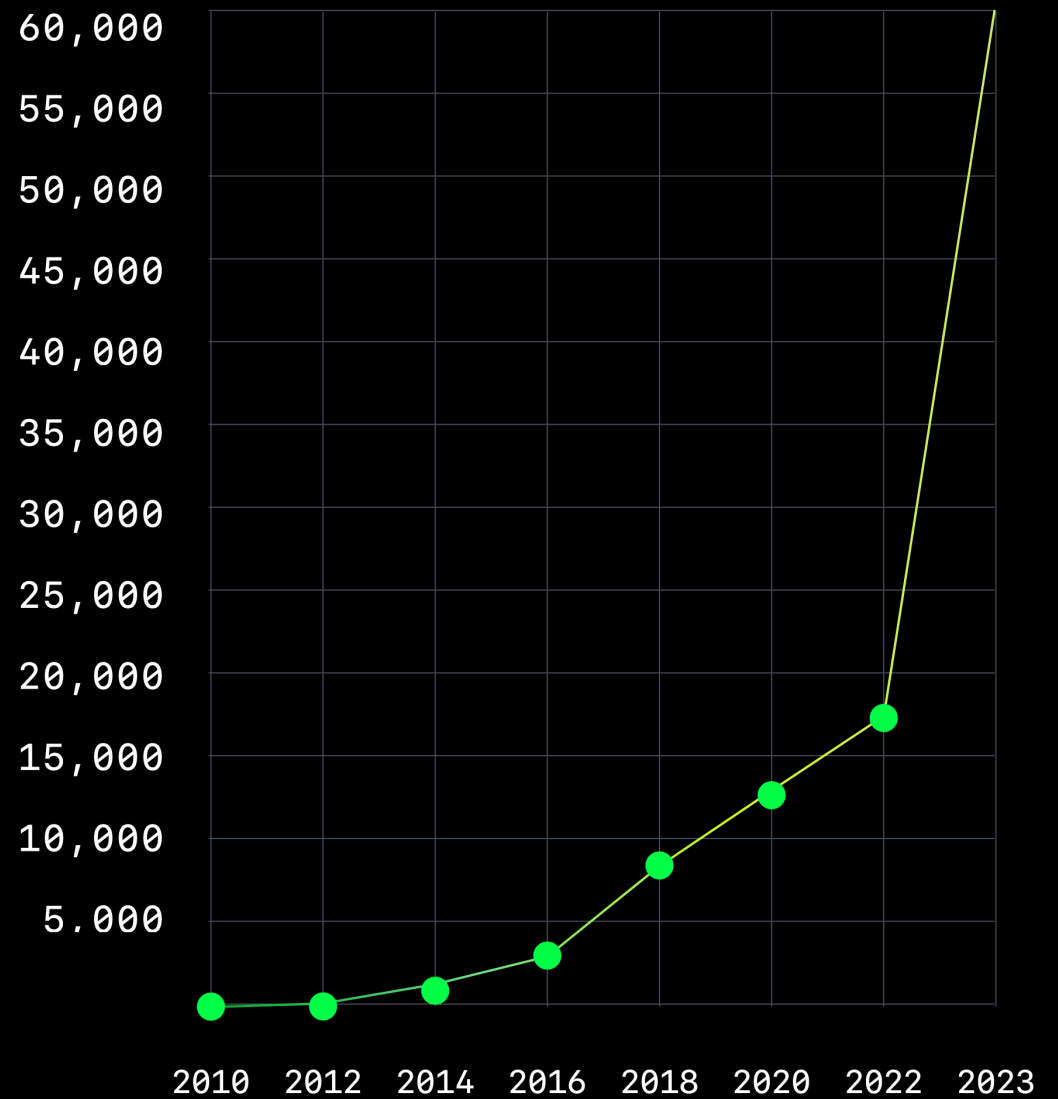
# The top 10 developer communities on GitHub from 2019-2023



# Projecting future growth of developer communities (2023-2028)

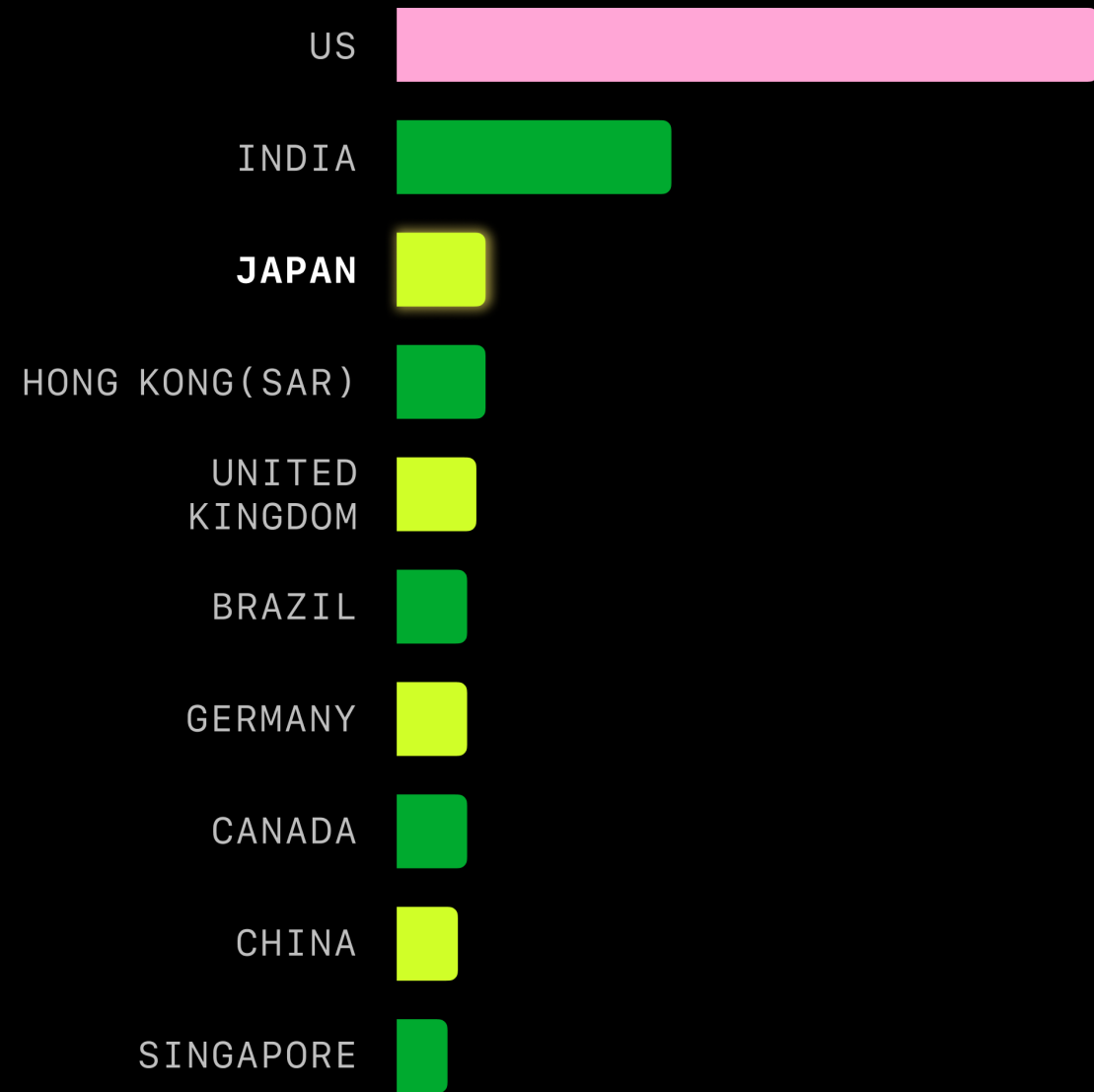


# The global growth in generative AI projects on GitHub





# Top 10 global communities creating the most generative AI projects on GitHub



# Home for Developers





# AI-powered developer platform



COPILOT



COLLABORATION



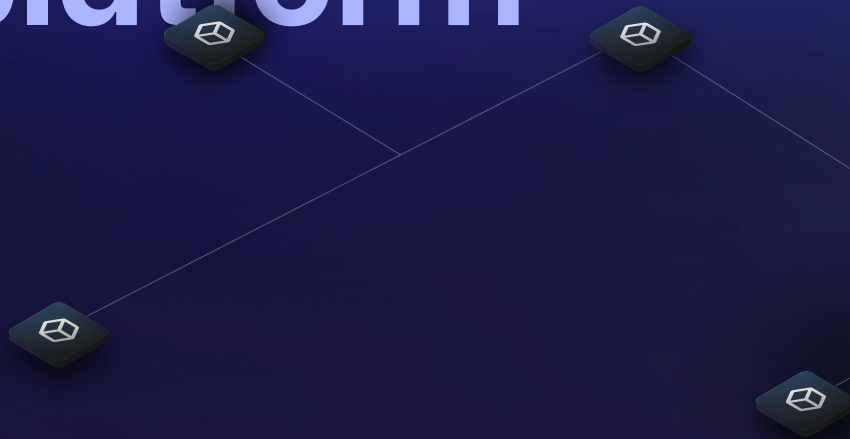
PRODUCTIVITY



SECURITY



SCALE



 GitHub Copilot

# Most widely adopted AI developer tool

 1M paid users  190 countries

**55% faster**

The image features a dark blue background with a bright green curved shape on the right side. The text "55% faster" is prominently displayed in the center-left area.



stripe



LINEヤフー



indra

Panasonic  
CONNECT

SAS



duolingo



Infosys



accenture



Gall&Gall  
SINDS 1884



37,000+

Organizations using GitHub Copilot

```
expenses_string
expected_output = []
self.assertEqual(parse_expenses(expenses_string),

def test_parse_expenses_with_comments(self):
    expenses_string = '# This is a comment\n2023-01-0
    expected_output = [(datetime.datetime(2023, 1, 2)
    self.assertEqual(parse_expenses(expenses_string),

if __name__ == '__main__':
    unittest.main()
```

The code assumes that the datetime module is imported. The test cases cover different scenarios such as valid input, empty input, input with comments, invalid date format, and invalid value format.

Analyze and debug

Ask a question or type '/' for topics

```
18         float(value),
19         currency))
20     return expenses
21
22 expenses_data = '''2023-01-02 -34.01 USD
23                 2023-01-03 2.59 DKK
24                 2023-01-03 -2.72 EUR'''
25
26
27
28
29
30
31
```

Ln 17, Col 3 Spaces: 2 UTF-8 LF {} TypeScript

 GitHub Copilot Chat

# Makes natural language a new, universal programming language





# myFirstCommit

@ Enter a GitHub Handle



# Demo

UNIVERSE23



# Copilot Chat in VS & VS Code

GITHUB COPILOT: CHAT

GitHub Copilot

```
import unittest
import datetime

class TestParseExpenses(unittest.TestCase):
    def test_parse_expenses_with_valid_input(self):
        expenses_string = '2023-01-02 -34.01 USD\n2023-01-03 2.59 DKK\n2023-01-03 -2.72 EUR'
        expected_output = [(datetime.datetime(2023, 1, 2), -34.01, 'USD'),
                            (datetime.datetime(2023, 1, 3), 2.59, 'DKK'),
                            (datetime.datetime(2023, 1, 3), -2.72, 'EUR')]
        self.assertEqual(parse_expenses(expenses_string), expected_output)

    def test_parse_expenses_with_empty_input(self):
        expenses_string = ''
        expected_output = []
        self.assertEqual(parse_expenses(expenses_string), expected_output)

    def test_parse_expenses_with_comments(self):
        expenses_string = '# This is a comment\n2023-01-02 -34.01 USD\n2023-01-03 2.59 DKK\n2023-01-03 -2.72 EUR'
        expected_output = [(datetime.datetime(2023, 1, 2), -34.01, 'USD'),
                            (datetime.datetime(2023, 1, 3), 2.59, 'DKK'),
                            (datetime.datetime(2023, 1, 3), -2.72, 'EUR')]
        self.assertEqual(parse_expenses(expenses_string), expected_output)

if __name__ == '__main__':
    unittest.main()
```

The code assumes that the datetime module is imported. The test cases cover different scenarios such as valid input, empty input, input with comments, invalid date format, and invalid value format.

Analyze and debug

Ask a question or type '/' for topics

```
expenses_string):
    expenses and return the list of
    g with #.
    datetime.
    ing:
    1 USD
    DKK
    EUR

_string.splitlines():
    th("#"):

    rency = line.split(" ")
    (datetime.datetime.strptime(date
    float(value),
    currency))

01-02 -34.01 USD
03 2.59 DKK
03 -2.72 EUR'''
```



# Copilot Chat available in JetBrains IDEs

GitHub Copilot Chat

Compilation problems in current project

monalisa

Why is my code not compiling?

GitHub Copilot

The compilation errors in your `ChatMessageController.java` file are due to the unrecognized symbols `RequestMapping` and `RestController`. These are annotations provided by the Spring Framework, but they are not being recognized in your current setup.

To resolve this issue, you need to ensure that the Spring Web dependency is included in your Maven configuration and import the necessary classes in your `ChatMessageController.java` file.

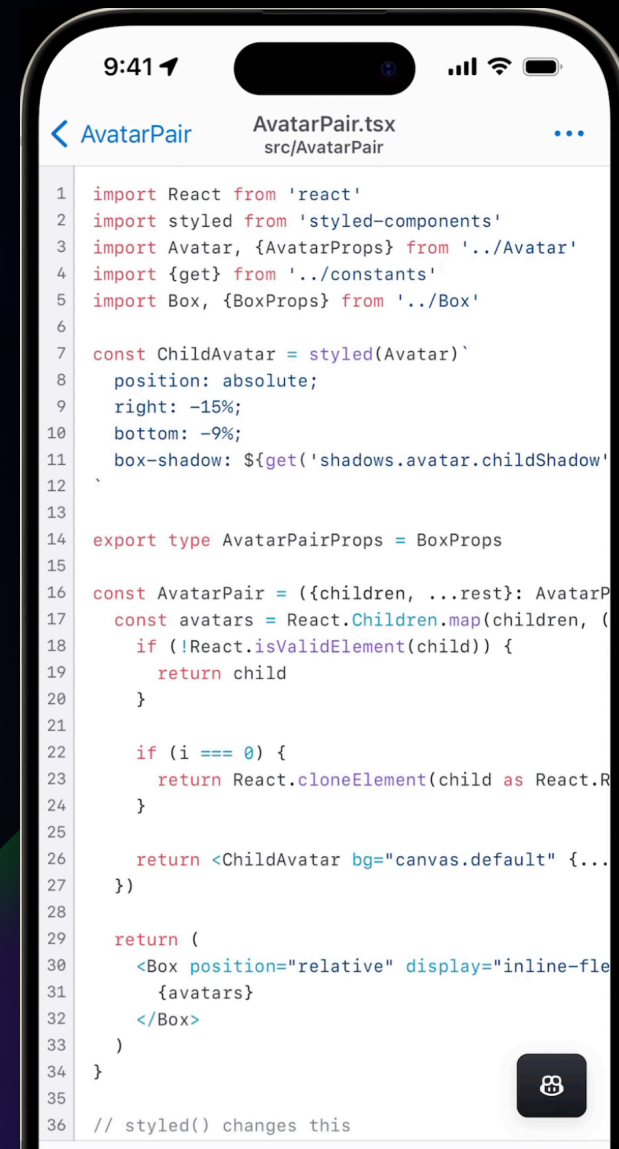
Here's how you can do it:

1. Add the Spring Web dependency to your `pom.xml`:

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-web</artifactId>  
</dependency>
```

Ask Copilot a question or type '/' for commands

# Copilot Chat in Mobile




```
9:41 📶 🔋  
< AvatarPair AvatarPair.tsx  
src/AvatarPair  
1 import React from 'react'  
2 import styled from 'styled-components'  
3 import Avatar, {AvatarProps} from '../Avatar'  
4 import {get} from '../constants'  
5 import Box, {BoxProps} from '../Box'  
6  
7 const ChildAvatar = styled(Avatar)`  
8   position: absolute;  
9   right: -15%;  
10  bottom: -9%;  
11  box-shadow: ${get('shadows.avatar.childShadow')};  
12 `;  
13  
14 export type AvatarPairProps = BoxProps  
15  
16 const AvatarPair = ({children, ...rest}: AvatarPairProps) => {  
17   const avatars = React.Children.map(children, (child, i) => {  
18     if (!React.isValidElement(child)) {  
19       return child  
20     }  
21  
22     if (i === 0) {  
23       return React.cloneElement(child as React.ReactElement, {  
24         style: {position: 'relative', display: 'inline-block'}  
25       });  
26     }  
27     return <ChildAvatar bg="canvas.default" {...child.props} />;  
28   });  
29   return (  
30     <Box position="relative" display="inline-block" style={rest} />  
31     {avatars}  
32   </Box>  
33 );  
34 }  
35  
36 // styled() changes this
```

 GitHub Copilot Chat

# Even more powerful

 Context aware

 Inline chat

 Upgraded to GPT-4

/ Slash commands

 Smart actions

```
def test_parse_expenses_with_empty_input(self):
    expenses_string = ''
    expected_output = []
    self.assertEqual(parse_expenses(expenses_string),
                      expected_output)

def test_parse_expenses_with_comments(self):
    expenses_string = '# This is a comment\n2023-01-02 -34.01 USD'
    expected_output = [(datetime.datetime(2023, 1, 2),
                                -34.01, 'USD')]
    self.assertEqual(parse_expenses(expenses_string),
                      expected_output)

if __name__ == '__main__':
    unittest.main()
```

The code assumes that the datetime module is imported. The test cases cover different scenarios such as valid input, empty input, input with comments, invalid date format, and invalid value format.

Analyze and debug

Ask a question or type '/' for topics

```
15         continue
16     date, value, currency = line.split(" ")
17     expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),
18                    float(value),
19                    currency))
20     return expenses
21
22 expenses_data = '''2023-01-02 -34.01 USD
23                 2023-01-03 2.59 DKK
24                 2023-01-03 -2.72 EUR'''
25
26
27
28
29
30
31
```

Ln 17, Col 3 Spaces: 2 UTF-8 LF {} TypeScript

 GitHub Copilot Chat

# Generally available

DEC 2023

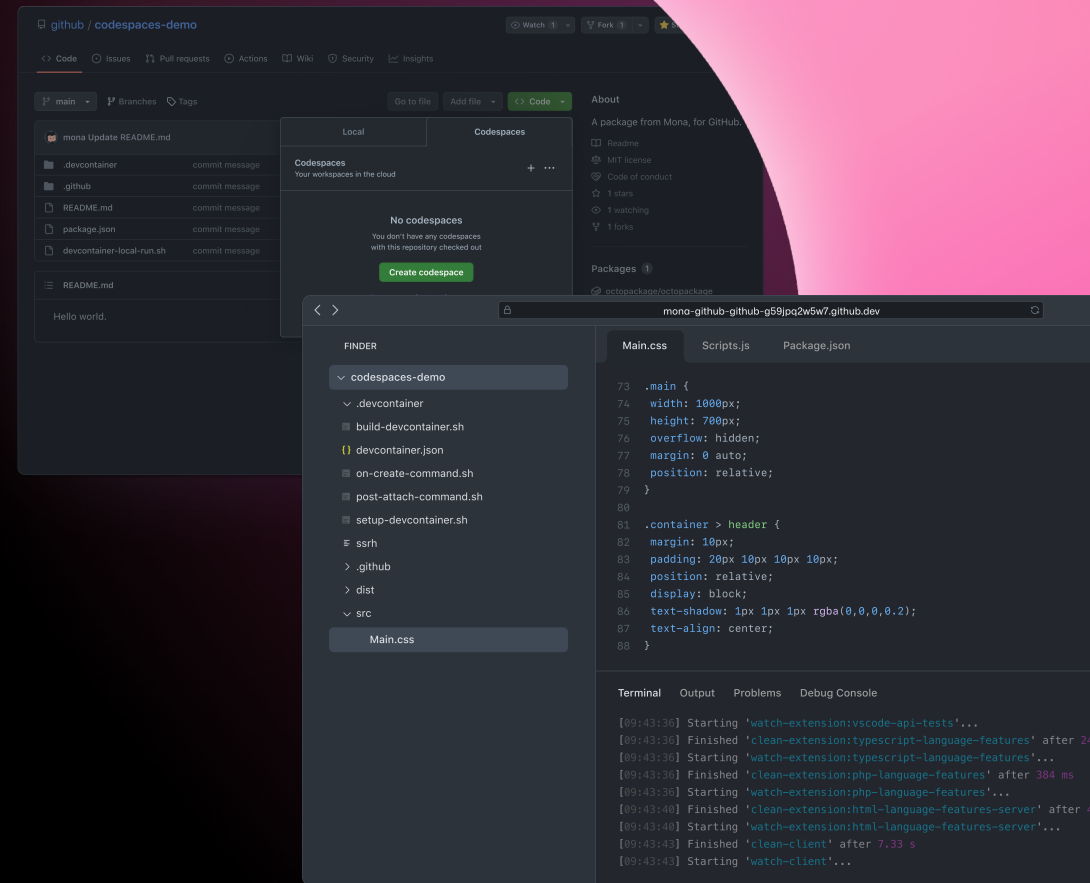
Accelerating idea to production

**Welcome to**  
**your flow state**



GitHub Codespaces

# Start coding instantly



# Collaborate from anywhere

❤️ 10

Answered

🚀 19

🎉 12

The screenshot shows the GitHub Discussions interface for the repository 'acme / octoinvaders'. At the top, there are navigation tabs for Code, Issues, Pull requests, Discussions (selected), Actions, Security, and Insights. Below the navigation, there are three featured discussion cards: 'Game dev log' by Stamat, 'Power-up contest' by johncreek, and 'Fan Art showcase' by tonyjaramillo. A search bar is located below the cards. The main content area is divided into two columns: 'Categories' and 'Discussions'. The 'Categories' column lists: General, Show & tell, Getting started, Feedback, Q&A, Game engine, and Art & Design. The 'Discussions' column lists: 'Power-up contest' (started 2 days ago in Feedback), 'Solo & coop modes' (Long term, started 3 days ago in General), 'Mars level design' (X Large, started 2 days ago in General), 'Implement sustainability themes into the game' (started 2 days ago in Art & Design), and 'Need help setup the game engine on my local' (engine, started 2 days ago in Game engine).

# Hit your milestones with ease



GitHub Actions

# Automate everything



M1 arm

```
Mona/.github/workflows/matrix-build-deploy.yml
```

```
1 name: matrix-build-deploy
2 on: repository_vulnerability_alert
3 jobs:
4   test
5   s
6
7   [ubuntu-latest, window-latest]
8
9
10
11
12
13
14
15   - uses: actions/checkout@mast
16   - uses: actions/setup-node@ma
17   build:
18     strategy:
19       matrix:
20         platform: [ubuntu-latest,
21         runs-on: ${ matrix.platform
22   publish:
23     needs: [build]
```

**build-deploy**

- >  lint 3s
- >  Check Pre-compiled 3s
- ▼  Test unit 3s
  - 1 ▶ Current runner version: '2.277.1'
  - 7 Operating System
  - 8 Virtual Environment
  - 9 GITHUB\_TOKEN Permissions
  - 10 Prepare workflow directory
  - 11 Prepare all required actions
  - 12 Getting action download info
  - 13 Download action repository 'actions/checkout@v2'
  - 14 Download action repository 'actions/setup-node@v1'
  - 15 Prepare all required actions
  - 16 Prepare workflow directory
- >  testYarnPnP 3s



# AI-powered developer platform



COPILOT



COLLABORATION



PRODUCTIVITY



SECURITY



SCALE



# Demo

UNIVERSE23



Embedded security

# GitHub Advanced Security

**7X+ faster**

Embedded security



# GitHub Advanced Security



Code  
scanning



Secret  
scanning



Supply chain  
security

AI-POWERED

# Code scanning autofix

The screenshot shows a code editor with a file named `src/DexcaliburEngine.js`. The code includes a `try` block with `await DeviceManager.getInstance().scan();` and `project = DexcaliburProject.load(this, pUID);`. A security alert from CodeQL is displayed, titled "Uncontrolled data used in path expression (High)", with the message "This path depends on a user-provided value." Below the alert, a "github-advanced-security" bot comment provides a detailed explanation of the vulnerability and suggests using `sanitize-filename` for validation. The "AI suggested fix" section shows the corrected code: `const DexcaliburWorkspace = require("../DexcaliburWorkspace");`, `const sanitize = require('sanitize-filename');`, and `let project = null, success = false;`. The `package.json` file is also updated to include `"sanitize-filename": "^1.6.3"`. A "Commit fix" button is visible at the bottom right of the code editor.

The "Commit fix" dialog box contains a text input field with the text "Fixes vulnerabilities with the power of AI ✨", a text area with the placeholder "Add an optional extended description...", and a "Commit changes" button at the bottom right.



## API Key #413

[Open](#) GitHub detected a secret 1 minute ago

Possibly active secret

```
api-4d3e-fc47-d778-cd38b-006e
```

AI-POWERED

# Secret scanning

Blame 6337 lines (6337 sloc) · 1.05 MB

```
export const useSWRHandler = <Data = any, Error = any>({
  73
  74
  75
  76
  77
  78
  79
  80
  81
  82
  83
  84
  85
  86
  87
  88
  89
  90
  91
  92
  93
  94
  95
  96
  97
  98
  99
  100
  101
  102
  103
  104
  105
})
```

Including credentials with Fetch API

monalisa 1s ago

@api.ts:182-195 how can I update this code to include credentials @MDN Web Docs

Copilot 1s ago

To include credentials in your fetch call, you need to set the credentials option to 'include' in the init object. This will send cookies and other authentication information on both same-origin and cross-origin requests. For example:

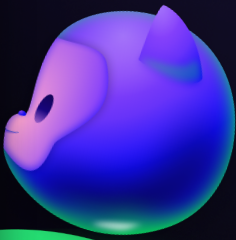
```
async function postData(url = "", data = 13) {
  // Default options are marked with *
  const response = await fetch(url, {
    method: "POST", // *GET, POST, PUT, DELETE, etc.
    cache: "no-cache", // *default, no-cache, reload,
    headers: {
      "Content-Type": "application/json",
      // 'Content-Type': 'application/x-www-form-urlencoded'
      redirect: "follow", // manual, *follow, error refer
    };
    return response.json(); // parses JSON response in
  }
}
```

To include credentials in your fetch call, you need to set the credentials option to 'include' in the init object. This will send cookies and other authentication information on both same-origin and cross-origin requests. For example:

References

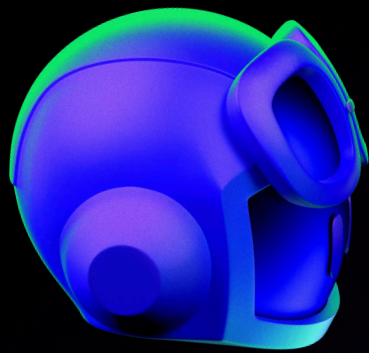
- Guides / Using Fetch API
- Sending a request with credentials included
- api.ts:182-195

Ask Copilot



AI-POWERED

# Copilot Chat in github.com



**Personalized to  
your organization**

# Demo

UNIVERSE23



# octoacademy

Organization account [Switch to another account](#)

[Go to your organization profile](#)

- General
- Billing and plans
- Repository roles
- Member privileges
- Team discussions
- Moderation
- Code, planning, and automation
  - Repository
  - Codespaces
  - Actions
- Copilot
  - General
  - Policies and features
  - Custom models** Beta
- Webhooks
- Packages
- Pages
- Projects

## Create a new custom model

Beta [Share feedback](#)

Grant access to repositories to train your model. By selecting repositories, you are granting permission to collect your repository data for fine-tuning the model. [Learn more about model fine-tuning.](#)

### Repository access

Grant permission to repositories and their data to train your model.

All repositories

### Allow access to telemetry data

Grant permission to collect and use [developer's telemetry data](#) for fine-tuning the model. **Not selecting this will impact the model performance.**

Off

*i* The selected repositories will take an estimated time of **36h 21m** to train.

[Create new custom model](#)

**Code**

main + 🔍

🔍 Jump to file

- models
  - `_init_.py`
  - `book.py`
  - `user.py`
- templates
- static
- database
- test
  - `app.py`
  - `config.py`
  - `requirements.txt`
  - `README.md`

bookstore\_app / models / book.py ⋮

monalisa initial setup and book crud bc07c5d · 23 days ago

Code Blame 🛡 6339 lines (6339 sloc) · 1.05 MB Raw

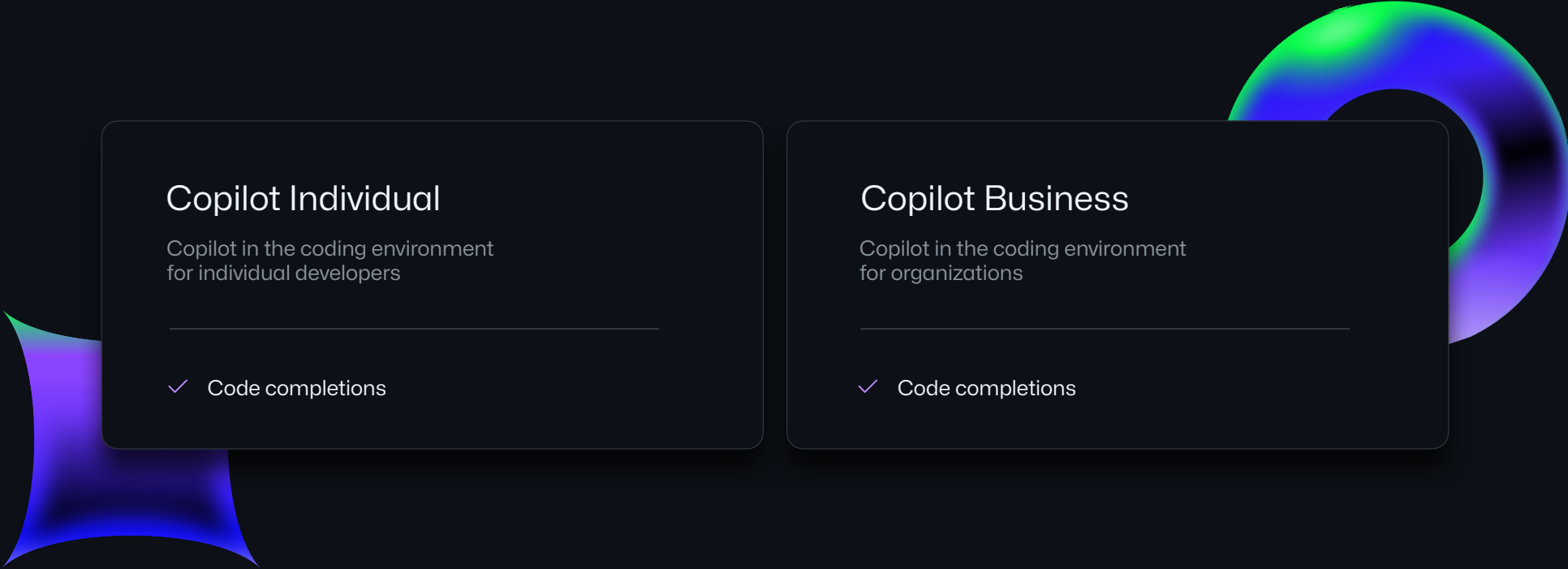
```
1 class Book:
2     def __init__(self, isbn, title, author):
3         self.isbn = isbn
4         self.title = title
5         self.author = author
6
7     def __repr__(self):
8         return f"<Book(isbn='{self.isbn}', title='{self.title}', author='{self.author}')>"
9
10    @classmethod
11    def find_books_by_author(cls, session, author_name):
12        """
13        List all books by a given author, ordered by title. Return a list of Book objects.
14        """
15        rows = session.execute('SELECT * FROM books WHERE author_name = %s ORDER BY title', [author_name])
16        books = []
17        for row in rows:
18            books.append(cls(row.isbn, row.title, row.author_name))
19        return books
20
21    @classmethod
22    def find_by_isbn(cls, session, isbn):
23        """
24        Retrieve a book by its ISBN. Return a Book object or None if not found.
25        """
26        row = session.execute('SELECT * FROM books WHERE isbn = %s', [isbn]).one()
27        if row:
28            return cls(row.isbn, row.title, row.author_name)
```



GitHub Copilot Partner Program

# Extend Copilot beyond your code





## Copilot Individual

Copilot in the coding environment  
for individual developers

---

✓ Code completions

## Copilot Business

Copilot in the coding environment  
for organizations

---

✓ Code completions

## Copilot Individual

Copilot in the coding environment  
for individual developers

---

- ✓ Code completions
- ✓ Chat in IDE and Mobile
- ✓ CLI assistance
- ✓ Security vulnerability filter
- ✓ Code referencing
- ✓ Public code filter

## Copilot Business

Copilot in the coding environment  
for organizations

---

- ✓ Code completions
- ✓ Chat in IDE and Mobile
- ✓ CLI assistance
- ✓ Security vulnerability filter
- ✓ Code referencing
- ✓ Public code filter
- ✓ IP indemnity
- ✓ Enterprise-grade security, safety, and privacy

## Copilot Individual

Copilot in the coding environment  
for individual developers

**\$10** / month

- ✓ Code completions
- ✓ Chat in IDE and Mobile
- ✓ CLI assistance
- ✓ Security vulnerability filter
- ✓ Code referencing
- ✓ Public code filter

## Copilot Business

Copilot in the coding environment  
for organizations

**\$19** per user / month

- ✓ Code completions
- ✓ Chat in IDE and Mobile
- ✓ CLI assistance
- ✓ Security vulnerability filter
- ✓ Code referencing
- ✓ Public code filter
- ✓ IP indemnity
- ✓ Enterprise-grade security, safety, and privacy

# Copilot Enterprise

Copilot personalized to your organization

## Copilot Enterprise

Copilot personalized for your organization throughout the software development lifecycle.

- \* Everything in Copilot Business, plus...
- ✓ Chat personalized to your codebase
- ✓ Documentation search and summaries
- ✓ Pull request summaries
- ✓ Code review skills
- ✓ Fine-tuned models

\*Requires GitHub Enterprise Cloud

# Copilot Enterprise

Copilot personalized to your organization

## Copilot Enterprise

Available Feb 2024

Copilot personalized for your organization throughout the software development lifecycle.

\$39 per user / month

- \* Everything in Copilot Business, plus...
- ✓ Chat personalized to your codebase
- ✓ Documentation search and summaries
- ✓ Pull request summaries
- ✓ Code review skills
- ✓ Fine-tuned models

\*Requires GitHub Enterprise Cloud



# AI-powered developer platform



COPILOT



COLLABORATION



PRODUCTIVITY



SECURITY



SCALE





[gh.io/features](https://gh.io/features)

UNIVERSE23





**Thank you!**

