

EXAM GH-300

GitHub Copilot Practice Question Bank

Every question, answer, and explanation from the exam simulator

117

QUESTIONS

37

MULTI-ANSWER

700

PASS SCORE / 1000

Fact-checked. Every question and answer has been checked against current GitHub and Microsoft documentation, and every explanation links to the page that supports it.

Take the timed, scored version online: www.mightybs.com/certifications/gh-300/exam/

Study hub: www.mightybs.com/certifications/gh-300/

Updated September 25, 2026. Independent study material, not affiliated with or endorsed by Microsoft or GitHub.

Part 1: Questions

Answers are not shown here, so you can test yourself first.

- Where a question says *Select two* or *Select three*, choose exactly that many. Multi-answer questions are all-or-nothing, as on the real exam.
- Check your answers in [Part 2](#), which gives the answer, the reasoning, and a documentation link for every question. Each question number links to its answer.
- The real exam is scored from 0 to 1000 with 700 to pass. Getting about 70% right here is a reasonable guide, not a prediction.

1 A social media manager wants to use AI to filter content. How can they promote transparency in the platform's AI operations?

- ☐ A By regularly updating the AI filtering algorithm.
- ☐ B By relying on a well-regarded AI development company.
- ☐ C By focusing on user satisfaction with the content filtering.
- ☐ D By providing clear explanations about the types of content the AI is designed to filter and how it arrives at its conclusion.

2 What GitHub Copilot configuration needs to be enabled to protect against IP infringements?

- ☐ A Blocking public code matches
- ☐ B Blocking license check configuration
- ☐ C Allowing public code matches
- ☐ D Allowing license check configuration

3 How long does it take content exclusion to add or be updated?

- ☐ A Up to 30 minutes
- ☐ B 45 - 60 minutes
- ☐ C 60 - 90 minutes
- ☐ D 24 hours

4 Which GitHub Copilot plan allows for prompt and suggestion collection?

- ☐ A GitHub Copilot Individuals
- ☐ B GitHub Copilot Business
- ☐ C GitHub Copilot Enterprise
- ☐ D GitHub Copilot Codespace

5 What is the primary purpose of organization audit logs in GitHub Copilot Business?

- ☐ A To track the number of lines of code suggested by Copilot
 - ☐ B To assign software licenses within the organization
 - ☐ C To monitor code conflicts across repositories
 - ☐ D To monitor administrator activities and actions within the organization
-

6 Which of the following is correct about GitHub Copilot Knowledge Bases?

- ☐ A All repos are indexed
 - ☐ B Indexing is static
 - ☐ C It is an Enterprise feature
 - ☐ D All file types are indexed
-

7 GitHub Copilot in the Command Line Interface (CLI) can be used to configure the following settings:

Each correct answer presents part of the solution. Choose two.

- ☐ A The default execution confirmation
 - ☐ B Usage analytics
 - ☐ C The default editor
 - ☐ D GitHub CLI subcommands
-

8 What is used by GitHub Copilot in the IDE to determine the prompt context?

- ☐ A Information from the IDE like open tabs, cursor location, selected code.
 - ☐ B All the code in the current repository and any git submodules.
 - ☐ C The open tabs in the IDE and the current folder of the terminal.
 - ☐ D All the code visible in the current IDE.
-

9 What is zero-shot prompting?

- ☐ A Only giving GitHub Copilot a question as a prompt and no examples
- ☐ B Giving GitHub Copilot examples of the problem you want to solve
- ☐ C Telling GitHub Copilot it needs to show only the correct answer
- ☐ D Giving GitHub Copilot examples of the algorithm and outcome you want to use
- ☐ E Giving as little context to GitHub Copilot as possible

-
- 10** How does GitHub Copilot Chat utilize its training data and external sources to generate responses when answering coding questions?
- ☐ A It primarily relies on the model's training data to generate responses.
 - ☐ B It primarily uses search results from Bing to generate responses.
 - ☐ C It combines its training data set, code in user repositories, and external sources like Bing to generate responses.
 - ☐ D It uses user-provided documentation exclusively to generate responses.
-

- 11** When using GitHub Copilot Chat to generate boilerplate code for various test types, how can you guide the AI to follow the testing standards of your company?
- ☐ A By using a specific setting in GitHub Copilot's configuration.
 - ☐ B By using a specific command in the terminal.
 - ☐ C By using specific prompt examples in your chat request.
 - ☐ D By using a specific slash command in the prompt.
-

- 12** What can be done during AI development to minimize bias?
- ☐ A Improve on the computational efficiency and speed.
 - ☐ B Focus on accuracy of the data.
 - ☐ C Collect massive amounts of data for training.
 - ☐ D Use diverse data, fairness metrics, and human oversight.
-

- 13** Why might a Generative AI (Gen AI) tool create inaccurate outputs?
- ☐ A The Gen AI tool is overloaded with too many requests at once.
 - ☐ B The Gen AI tool is experiencing downtime and is not fully recovered.
 - ☐ C The Gen AI tool is programmed with a focus on creativity over factual accuracy.
 - ☐ D The training data might contain biases or inconsistencies.
-

- 14** How does GitHub Copilot typically handle code suggestions that involve deprecated features or syntax of programming languages?
- ☐ A GitHub Copilot automatically updates deprecated features in its suggestions to the latest version.
 - ☐ B GitHub Copilot may suggest deprecated syntax or features if they are present in its training data.
 - ☐ C GitHub Copilot always filters out deprecated elements to promote the use of current standards.
 - ☐ D GitHub Copilot rejects all prompts involving deprecated features to avoid compilation errors.

15 How long does GitHub retain Copilot data for Business and Enterprise?

Each correct answer presents part of the solution. Choose two.

- ☐ A Prompts and Suggestions: Not retained
 - ☐ B Prompts and Suggestions: Retained for 28 days
 - ☐ C User Engagement Data: Kept for Two Years
 - ☐ D User Engagement Data: Kept for One Year
-

16 What is the best way to share feedback about GitHub Copilot Chat when using it on GitHub Mobile?

- ☐ A Use the emojis in the Copilot Chat interface.
 - ☐ B The feedback section on the GitHub website.
 - ☐ C By tweeting at GitHub's official X (Twitter) account.
 - ☐ D The Settings menu in the GitHub Mobile app.
-

17 How can GitHub Copilot assist in maintaining consistency across your tests?

- ☐ A By identifying a pattern in the way you write tests and suggesting similar patterns for future tests.
 - ☐ B By automatically fixing all tests in the code based on the context.
 - ☐ C By providing documentation references based on industry best practices.
 - ☐ D By writing the implementation code for the function based on context.
-

18 What are the potential risks associated with relying heavily on code generated from GitHub Copilot?

Each correct answer presents part of the solution. Choose two.

- ☐ A GitHub Copilot may introduce security vulnerabilities by suggesting code with known exploits.
 - ☐ B GitHub Copilot may decrease developer velocity by requiring too much time in prompt engineering.
 - ☐ C GitHub Copilot's suggestions may not always reflect best practices or the latest coding standards.
 - ☐ D GitHub Copilot may increase development lead time by providing irrelevant suggestions.
-

19 How does GitHub Copilot suggest code optimizations for improved performance?

- ☐ A By analyzing the codebase and suggesting more efficient algorithms or data structures.
- ☐ B By automatically rewriting the codebase to use more efficient code.
- ☐ C By enforcing strict coding standards that ensure optimal performance.
- ☐ D By providing detailed reports on the performance of the codebase.

20 Which of the following scenarios best describes the intended use of GitHub Copilot Chat as a tool?

- ☐ A A complete replacement for developers generating code.
- ☐ B A productivity tool that provides suggestions, but relying on human judgment.
- ☐ C A solution for software development, requiring no additional input or oversight.
- ☐ D A tool solely designed for debugging and error correction.

21 How can GitHub Copilot assist developers during the requirements analysis phase of the Software Development Life Cycle (SDLC)?

- ☐ A By automatically generating detailed requirements documents.
- ☐ B By providing templates and code snippets that help in documenting requirements.
- ☐ C By identifying and fixing potential requirement conflicts when using /help.
- ☐ D By managing stakeholder communication and meetings.

22 How does GitHub Copilot assist developers in reducing the amount of manual boilerplate code they write?

- ☐ A By engaging in real-time collaboration with multiple developers to write boilerplate code.
- ☐ B By predicting future coding requirements and pre-emptively generating boilerplate code.
- ☐ C By refactoring the entire codebase to eliminate boilerplate code without developer input.
- ☐ D By suggesting code snippets that can be reused across different parts of the project.

23 How can users provide feedback about GitHub Copilot Chat using their IDE?

- ☐ A By filling out a feedback form on the GitHub website
- ☐ B By emailing the support team directly
- ☐ C By posting on the GitHub forums
- ☐ D Through the "Share Feedback" button in the Copilot Chat panel

24 Which GitHub Copilot pricing plans include features that exclude your GitHub Copilot data like usage, prompts, and suggestions from default training GitHub Copilot?

Choose two correct answers.

- ☐ A GitHub Copilot Business
- ☐ B GitHub Copilot Codespace
- ☐ C GitHub Copilot Individual
- ☐ D GitHub Copilot Enterprise

25 Which statements are true about the GitHub Copilot usage metrics API?

Choose two.

- ☐ A The API can generate detailed reports on code quality improvements made by GitHub Copilot.
- ☐ B The API can track the acceptance rate of code suggestions accepted and used in the organization.
- ☐ C The API can refactor your code to improve productivity.
- ☐ D The API can provide feedback on coding style and standards compliance.
- ☐ E The API can provide Copilot Chat specific suggestions acceptance metrics.

26 If you are working on open source projects, GitHub Copilot Individual can be paid:

- ☐ A Based on the payment method in your user profile
- ☐ B N/A – Copilot Individual is a free service for all open source projects
- ☐ C Through an invoice or a credit card
- ☐ D Through an Azure Subscription

27 What is a likely effect of GitHub Copilot being trained on commonly used code patterns?

- ☐ A Suggest innovative coding solutions that are not yet popular.
- ☐ B Suggest completely novel projects, while reducing time on a project.
- ☐ C Suggest code snippets that reflect the most common practices in the training data.
- ☐ D Suggest homogeneous solutions if provided a diverse data set.

28 Which scenarios can GitHub Copilot Chat be used to increase productivity?

Each correct answer presents part of the solution. Choose two.

- ☐ A A project plan for the team needs to be generated using a project management software.
- ☐ B Create a documentation file for the newly created code base.
- ☐ C A developer is added to a new project and would like to understand the current software code.
- ☐ D Fast tracking of release management activities to move code to production main branch.

29 What method can a developer use to generate sample data with GitHub Copilot?

Each correct answer presents part of the solution. Choose two.

- ☐ A Utilizing GitHub Copilot's ability to create fictitious information from patterns in training data.
- ☐ B Leveraging GitHub Copilot's ability to independently initiate and manage data storage services.
- ☐ C Utilize GitHub Copilot's capability to directly access and use databases to create sample data.
- ☐ D Leveraging GitHub Copilot's suggestions to create data based on API documentation in the repository.

30 What are the effects of content exclusions?

Each correct answer presents part of the solution. Choose two.

- ☐ A The excluded content is not directly available to GitHub Copilot to use as context.
- ☐ B GitHub Copilot suggestions are no longer available in the excluded files.
- ☐ C The excluded content is no longer used while debugging the code.
- ☐ D The IDE will not count coding suggestions in the excluded content.

31 What types of prompts or code snippets might be flagged by the GitHub Copilot toxicity filter?

Each correct answer presents part of the solution. Choose two.

- ☐ A Hate speech or discriminatory language (e.g., racial slurs, offensive stereotypes)
- ☐ B Sexually suggestive or explicit content
- ☐ C Code that contains logical errors or produces unexpected results
- ☐ D Code comments containing strong opinions or criticisms

32 When using an IDE with a supported GitHub Copilot plug-in, which Chat features can be accessed from within the IDE?

Each correct answer presents part of the solution. Choose two.

- ☐ A Explain code and suggest improvements
- ☐ B Generate unit tests
- ☐ C Plan coding tasks
- ☐ D Find out about releases and commits

33 Which principle emphasizes that AI systems should be understandable and provide clear information on how they work?

- ☐ A Fairness
- ☐ B Transparency
- ☐ C Inclusiveness
- ☐ D Accountability

34 What is one of the recommended practices when using GitHub Copilot Chat to enhance code quality?

- ☐ A Avoid using Copilot for complex tasks.
- ☐ B Disable Copilot's inline suggestions.
- ☐ C Regularly review and refactor the code suggested by Copilot.
- ☐ D Rely solely on Copilot's suggestions without reviewing them.

35 How can the concept of fairness be integrated into the process of operating an AI tool?

- ☐ A Focusing on accessibility will ensure fairness.
- ☐ B Focusing on collecting large datasets for training will ensure fairness.
- ☐ C Regularly monitoring the AI tool's performance will ensure fairness in its outputs.
- ☐ D Training AI data and algorithms to be free from biases will ensure fairness.

36 How does GitHub Copilot Chat help to fix security issues in your codebase?

- ☐ A By enforcing strict coding standards that prevent the introduction of vulnerabilities.
- ☐ B By providing detailed reports on the security vulnerabilities present in the codebase.
- ☐ C By annotating the given suggestions with known vulnerability patterns.
- ☐ D By automatically refactoring the entire codebase to remove vulnerabilities.

37 Which Microsoft ethical AI principle is aimed at ensuring AI systems treat all people equally?

- ☐ A Inclusiveness
 - ☐ B Fairness
 - ☐ C Reliability and Safety
 - ☐ D Privacy and Security
-

38 How is GitHub Copilot Individual billed?

Each correct answer presents part of the solution. Choose two.

- ☐ A Monthly as a subscription
 - ☐ B Annually as a subscription
 - ☐ C Monthly, as a metered service based on actual consumption
 - ☐ D Free (not billed) for all open source projects
-

39 What is the impact of the "Fill-In-the-Middle" (FIM) technique on GitHub Copilot's code suggestions?

- ☐ A Restricts Copilot to use only external databases for generating code suggestions.
 - ☐ B Allows Copilot to generate suggestions based only on the prefix of the code.
 - ☐ C Ignores both the prefix and suffix of the code, focusing only on user comments for context.
 - ☐ D Improves suggestions by considering both the prefix and suffix of the code, filling in the middle part more accurately.
-

40 What two options navigate to configure duplicate detection?

Each correct answer presents part of the solution. Choose two.

- ☐ A Enterprise settings Copilot Policies
 - ☐ B Repository settings Copilot Policies
 - ☐ C Organization settings Copilot Policies
 - ☐ D User settings Copilot Policies
-

41 Why is code reviewing still necessary when using GitHub Copilot to write tests?

- ☐ A Because GitHub Copilot can cover all possible scenarios in your test cases.
- ☐ B Because GitHub Copilot generates the best code possible for the test scenario.
- ☐ C Because GitHub Copilot's generated test cases may not cover all possible scenarios.
- ☐ D Because GitHub Copilot replaces the need for manual testing.

42 What role does the pre-processing of user input play in the data flow of GitHub Copilot Chat?

- ☐ A It filters out irrelevant information from the user's input prompt.
 - ☐ B It enriches the input prompt with additional context before passing it to the language model.
 - ☐ C It directly generates a response based on the user's input prompt.
 - ☐ D It formats the output response before presenting it to the user.
-

43 How do you generate code suggestions with GitHub Copilot in the CLI?

- ☐ A Type out the code snippet Use the copilot refine command to enhance it Review the suggested command.
 - ☐ B Write code comments Press the suggestion shortcut Select the best suggestion from the list.
 - ☐ C Use gh copilot suggest Write the command you want Select the best suggestion from the list.
 - ☐ D Describe the project's architecture Use the copilot generate command Accept the generated suggestion.
-

44 How can GitHub Copilot assist with code refactoring tasks?

- ☐ A GitHub Copilot can fix syntax errors without user input.
 - ☐ B GitHub Copilot can automatically rewrite code to follow best practices.
 - ☐ C GitHub Copilot can suggest refactoring improvements for better code quality.
 - ☐ D GitHub Copilot can remove unnecessary files from the project directory.
-

45 What is the correct way to access the audit log events for GitHub Copilot Business?

- ☐ A Navigate to the Security tab in the organization's GitHub settings
 - ☐ B Navigate to the Insights tab in the repository settings
 - ☐ C Use the Audit log section in the organization's GitHub settings
 - ☐ D Use the Code tab in the GitHub repository
-

46 When can GitHub Copilot still use content that was excluded using content exclusion?

- ☐ A If the contents of an excluded file are referenced in code that is not excluded, for example function calls.
- ☐ B When the repository level settings allow overrides by the user.
- ☐ C If the content exclusion was configured at the enterprise level, and is overwritten at the organization level.
- ☐ D When the user prompts with @workspace.

47 How can the insights gained from the metrics API be used to improve the development process in conjunction with GitHub Copilot?

- ☐ A Real-time debugging and error resolution statistics.
- ☐ B Insights on the types of coding languages where GitHub Copilot is most helpful.
- ☐ C Automated generation of complete project documentation.
- ☐ D Detailed analysis of GitHub Copilot's suggestions vs. manual coding.

48 In what way can GitHub Copilot and GitHub Copilot Chat aid developers in modernizing applications?

- ☐ A GitHub Copilot can directly convert legacy applications into cloud-native architectures.
- ☐ B GitHub Copilot can suggest modern programming patterns based on your code.
- ☐ C GitHub Copilot can create and deploy full-stack applications based on a single query.
- ☐ D GitHub Copilot can refactor applications to align with upcoming standards.

49 How can GitHub Copilot facilitate a smoother learning experience when diving into a new programming language?

Each correct answer presents part of the solution. Choose two.

- ☐ A GitHub Copilot Chat can provide guidance and support for common coding tasks and challenges in the targeted programming language.
- ☐ B GitHub Copilot's /understand command will help GitHub Copilot to understand code written in a targeted programming language.
- ☐ C GitHub Copilot can provide contextualized code suggestions and answer sources from an organization's documentation.
- ☐ D GitHub Copilot can convert comments into code to grasp the syntax and nuances of a new programming language.

50 A company is currently storing code in Bitbucket and would like to use GitHub Copilot. Which GitHub Copilot plan will be most cost effective to allow them to manage users with their Identity Provider (e.g. Okta)?

- ☐ A GitHub Copilot Business for non-GHE customers
- ☐ B GitHub Copilot Individual
- ☐ C GitHub Copilot Enterprise
- ☐ D GitHub Copilot Teams

51 How does the `/tests slash` command assist developers?

- ☐ A Constructs detailed test documentation.
- ☐ B Creates unit tests for the selected code.
- ☐ C Integrates with external testing frameworks.
- ☐ D Executes test cases to find issues with the code.

52 What is the primary role of the `/optimize slash` command in Visual Studio?

- ☐ A Translates code into a more performant language.
- ☐ B Enhances the performance of the selected code by analyzing its runtime complexity.
- ☐ C Automatically formats the code according to the selected style guide.
- ☐ D Summarizes your documentation into more maintainable and readable formats.

53 What do you check when GitHub Copilot content exclusions are not working?

Each correct answer presents part of the solution. Choose two.

- ☐ A If GitHub Copilot can connect to the server selected in your user settings.
- ☐ B If the user is part of the content exclusion team that limits the use of content exclusions.
- ☐ C If the content exclusion settings changed in the last 30 minutes or before that.
- ☐ D If the user is in an organization that has content exclusions configured.

54 How can you get multiple suggestions from GitHub Copilot?

- ☐ A By asking for multiple suggestions using comments in your code
- ☐ B By opening the completions panel in your editor
- ☐ C By using the inline chat functionality with the command 'multiple'
- ☐ D By using `@workspace` in the chat window

55 Which of the following GitHub Copilot Business related activities can be tracked using the organization audit logs?

- ☐ A Accepted chat suggestions
- ☐ B Code suggestions made by GitHub Copilot
- ☐ C Changes to content exclusion settings
- ☐ D Suggestions blocked by duplication detection filtering

56 Which REST API endpoint is used to modify details about a GitHub Copilot Business subscription?

Each correct answer presents part of the solution. Choose two.

- ☐ A Add teams to the Copilot subscription for an organization
- ☐ B Upgrade or downgrade the subscription tier
- ☐ C Migrate Copilot seat assignments between GitHub organizations
- ☐ D Reassign Copilot seats based on GitHub repository size
- ☐ E Remove teams from the Copilot subscription for an organization

57 Which of the following prompts can be used to guide GitHub Copilot Chat in refactoring code for quality improvements?

Each correct answer presents part of the solution. Choose two.

- ☐ A "Show me how to improve the readability of this function."
- ☐ B "Suggest ways to enhance the maintainability of this code segment."
- ☐ C "Refactor my application to meet the latest coding standards."
- ☐ D "Predict future coding trends and update my codebase accordingly."

58 What is few-shot prompting?

- ☐ A Giving GitHub Copilot several examples of the input and output you want, so it can infer the pattern before it responds
- ☐ B Telling GitHub Copilot from which sources it should base the response on
- ☐ C Telling GitHub Copilot to try multiple times to answer the prompt
- ☐ D Asking GitHub Copilot to generate code from a description alone, without any examples

59 Where is the proxy service hosted?

- ☐ A Self hosted
- ☐ B Amazon Web Service
- ☐ C Microsoft Azure
- ☐ D Google Cloud Platform

60 A team is using GitHub Copilot Individual in their daily development activities. They need to exclude specific files from being used to inform code completion suggestions. How can they achieve this?

- ☐ A Have an organization owner configure content exclusions
- ☐ B Add a .gitignore file to the repo
- ☐ C Have a repo administrator configure content exclusions
- ☐ D Use the #file Chat variable to exclude the files
- ☐ E Upgrade to Copilot Business

61 What reasons could apply if code suggestions are not working in your editor?

Select three.

- ☐ A You are working in files included in your .gitignore
- ☐ B You do not have an active internet connection
- ☐ C Your programming language is not supported
- ☐ D Your content exclusion is active and blocks the use of GitHub Copilot
- ☐ E You do not have a valid GitHub Copilot license

62 How does GitHub Copilot identify matching code and ensure that public code is appropriately handled or blocked?

Each correct answer presents part of the solution. Choose two.

- ☐ A Using machine learning models trained only on private repositories
- ☐ B Reviewing and storing user-specific private repository data for future suggestions
- ☐ C Filtering out suggestions that match code from public repositories
- ☐ D Implementing safeguards to detect and avoid suggesting verbatim snippets from public code

63 Which of the following does GitHub Copilot's LLM derive context from when producing a response?

- ☐ A Frequency of commits to the repository
- ☐ B Syntax highlighting scheme of the code in the IDE
- ☐ C Neighboring or related files within a project
- ☐ D Version control system integrated with the IDE

64 Where can you validate if GitHub Copilot is not returning suggestions because of content exclusions?

- ☐ A The GitHub Copilot icon in the status bar of the editor will display a message
 - ☐ B The GitHub Copilot logs on GitHub.com under your user settings
 - ☐ C The code suggestions window will display a warning message
 - ☐ D The GitHub Copilot errors panel in your IDE
-

65 How does GitHub Copilot Chat ensure that a function works correctly?

- ☐ A By suggesting assertions based on the code's context and semantics.
 - ☐ B By automatically writing all the tests for the function.
 - ☐ C By writing the implementation code for the function.
 - ☐ D By executing the test cases to validate the correctness of the code.
-

66 How does GitHub Copilot utilize chat history to enhance its code completion capabilities?

- ☐ A By using chat history to offer personalized code snippets based on previous prompts.
 - ☐ B By logging chat history to monitor user activity and ensure compliance with coding standards.
 - ☐ C By analyzing past chat interactions to identify common programming patterns and errors.
 - ☐ D By sharing chat history with third-party services to improve integration and functionality.
-

67 Identify the steps involved in the life cycle of a GitHub Copilot code suggestion?

Each correct answer presents part of the solution. Choose two.

- ☐ A Generate suggestions
 - ☐ B Capturing the user's context
 - ☐ C Processing telemetry data
 - ☐ D Retraining the model
 - ☐ E Storing user data
-

68 Which of the following steps correctly demonstrates how to establish an organization-wide policy for GitHub Copilot Business to restrict its use to certain repositories?

- ☐ A Apply policies through the GitHub Actions configuration
- ☐ B Create a copilot.policy file in each repository
- ☐ C Configure the policies in the organization settings
- ☐ D Create a copilot.policy in the .github repository

69 Which of the following statements best describes the impact of GitHub Copilot on the software development process?

- ☐ A It decreases software vulnerabilities from third party dependencies.
 - ☐ B It reduces overhead by automating testing workflows.
 - ☐ C It increases productivity by automating repetitive coding tasks.
 - ☐ D It replaces the need for developers in the software development process.
-

70 What method can be used to interact with GitHub Copilot?

- ☐ A By using a properly configured GitHub CLI
 - ☐ B By using chat capabilities in NeoVim
 - ☐ C From a watch window in an IDE debug session
 - ☐ D From a web browser at <https://github.copilot.com>
-

71 Which Copilot Individual features are available when using a supported extension for Visual Studio, VS Code, or JetBrains IDEs?

Each correct answer presents part of the solution. Choose two.

- ☐ A Code suggestions
 - ☐ B Chat
 - ☐ C Knowledge Base
 - ☐ D Pull Request Diff Analysis
-

72 What types of content can GitHub Copilot Knowledge Base answer questions about?

Each correct answer presents part of the solution. Choose three.

- ☐ A Code snippets
 - ☐ B Compiled binaries
 - ☐ C Documentation
 - ☐ D Design patterns
 - ☐ E Screenshots
-

73 What configuration needs to be set to get help from Microsoft and GitHub protecting against IP infringement while using GitHub Copilot?

- ☐ A Suggestions matching public code to 'blocked'
- ☐ B Enforce blocking of MIT or GPL licensed code
- ☐ C You need to check code suggestions yourself before accepting
- ☐ D Enable GitHub Copilot license checking

74 Which of the following are true about code suggestions?

Each correct answer presents part of the solution. Choose two.

- ☐ A Code suggestions are guaranteed to not expose known security vulnerabilities
- ☐ B You can use keyboard shortcuts to accept the next word in a suggestion
- ☐ C Code suggestions are limited to single-line suggestions
- ☐ D Code suggestions will always compile or run without modifications
- ☐ E Alternative code suggestions can be shown in a new tab

75 What should developers consider when relying on GitHub Copilot for generating code that involves statistical analysis?

- ☐ A GitHub Copilot can independently verify the statistical significance of results.
- ☐ B GitHub Copilot's suggestions are based on statistical trends and may not always apply accurately to specific datasets.
- ☐ C GitHub Copilot will automatically correct any statistical errors found in the user's initial code.
- ☐ D GitHub Copilot can design new statistical methods that have not been previously documented.

76 What specific function does the /fix slash command perform?

- ☐ A Initiates a code review with static analysis tools for security and logic errors.
- ☐ B Converts pseudocode into executable code, optimizing for readability and maintainability.
- ☐ C Generates new code snippets based on language syntax and best practices.
- ☐ D Proposes changes for detected issues, suggesting corrections for syntax errors and programming mistakes.

77 Which of the following describes role prompting?

- ☐ A Describing in your prompt what your role is to get a better suggestion
- ☐ B Tell GitHub Copilot in what tone of voice it should respond
- ☐ C Prompt GitHub Copilot to explain what was the role of a suggestion
- ☐ D Giving GitHub Copilot multiple examples of the form of the data you want to use

78 What kind of insights can the GitHub Copilot usage metrics API provide to help evaluate the effectiveness of GitHub Copilot?

Each correct answer presents part of the solution. Choose two.

- ☐ A The API can generate detailed reports on code quality improvements made by GitHub Copilot.
- ☐ B The API can track the number of code suggestions accepted and used in the organization.
- ☐ C The API can provide Copilot Chat specific suggestions acceptance metrics.
- ☐ D The API can refactor your code to improve productivity.
- ☐ E The API can provide feedback on coding style and standards compliance.

79 Which of the following statements correctly describes how GitHub Copilot Individual uses prompt data?

Each correct answer presents part of the solution. Choose two.

- ☐ A Prompt data is stored unencrypted for faster processing.
- ☐ B Prompt data is used internally by GitHub for improving the search engine.
- ☐ C Prompt data is used to train machine learning models for better code suggestions.
- ☐ D Real-time user input helps generate context-aware code suggestions.

80 What is the main purpose of the duplication detection filter in GitHub Copilot?

- ☐ A To compare user-generated code against a private repository for potential matches.
- ☐ B To allow administrators to control which suggestions are visible to developers based on custom criteria.
- ☐ C To encourage the user to follow coding best practices preventing code duplication.
- ☐ D To detect and block suggestions that match public code snippets on GitHub if they contain about 150 characters.

81 An Azure DevOps organization wants to buy GitHub Copilot for its developers and manage their licenses centrally, without paying for GitHub Enterprise licenses. Which plan should it use?

- ☐ A GitHub Copilot Pro
- ☐ B GitHub Copilot Enterprise
- ☐ C GitHub Copilot Business
- ☐ D GitHub Copilot for Azure DevOps

82 What type of information can you retrieve through GitHub Copilot Business Subscriptions via REST API?

Each correct answer presents part of the solution. Choose two.

- ☐ A Get a summary of GitHub Copilot usage for organization members
 - ☐ B List all GitHub Copilot seat assignments for an organization
 - ☐ C View code suggestions for a specific user
 - ☐ D List of all unsubscribed GitHub Copilot members within an organization
-

83 Select a strategy to increase the performance of GitHub Copilot Chat.

- ☐ A Optimize the usage of memory-intensive operations within generated code
 - ☐ B Apply prompt engineering techniques to be more specific
 - ☐ C Use a single GitHub Copilot Chat query to find resolutions for the collection of technical requirements
 - ☐ D Limit the number of concurrent users accessing GitHub Copilot Chat
-

84 What is a limitation of content exclusions?

- ☐ A Repository administrators and organization owners cannot manage content exclusion settings.
 - ☐ B Content exclusions can be worked around as it is only available for Git repositories.
 - ☐ C Content exclusions can only be configured by an enterprise administrator.
 - ☐ D Content exclusions are only available in the GitHub Copilot Individual plan.
-

85 How can GitHub Copilot aid developers in writing documentation for their code?

- ☐ A GitHub Copilot cannot assist in writing documentation or comments.
 - ☐ B GitHub Copilot can automatically generate complete and detailed documentation.
 - ☐ C GitHub Copilot can suggest summaries or descriptions based on the code's functionality.
 - ☐ D GitHub Copilot can only generate content in markdown format.
-

86 What role does chat history play in GitHub Copilot's code suggestions?

- ☐ A Chat history is used to train the GitHub Copilot model in real-time.
- ☐ B Chat history provides context to GitHub Copilot, improving the relevance and accuracy of its code suggestions.
- ☐ C Chat history is stored and shared with other users to enhance collaboration.
- ☐ D Chat history is irrelevant to GitHub Copilot and does not affect its functionality.

87 When using GitHub Copilot to identify missing tests in your codebase, which of the following is the most important factor to consider?

- ☐ A Using well-known coding practices in your repository.
- ☐ B Ensuring that the correct context is available to GitHub Copilot.
- ☐ C Close all the tabs in your IDE that do not have tests in them.
- ☐ D Having a high test coverage percentage in the codebase.

88 What GitHub Copilot feature can be configured at the organization level to prevent GitHub Copilot suggesting publicly available code snippets?

- ☐ A GitHub Copilot Chat in the IDE
- ☐ B GitHub Copilot Chat in GitHub Mobile
- ☐ C GitHub Copilot duplication detection filter
- ☐ D GitHub Copilot access to Bing

89 What is a benefit of using custom models in GitHub Copilot?

- ☐ A Responses are faster to produce and appear sooner
- ☐ B Responses use practices and patterns in your repositories
- ☐ C Responses use the organization's LLM engine
- ☐ D Responses are guaranteed to be correct

90 When using GitHub Copilot Chat to generate unit tests, which slash command would you use?

- ☐ A `/init-tests`
- ☐ B `/create-tests`
- ☐ C `/generate-tests`
- ☐ D `/tests`

91 How can you use GitHub Copilot to get inline suggestions for refactoring your code?

Select two.

- ☐ A By adding comments to your code and triggering a suggestion.
- ☐ B By highlighting the code you want to fix, right-clicking, and selecting "Fix using GitHub Copilot."
- ☐ C By running the `gh copilot fix` command.
- ☐ D By using the `/fix` command in GitHub Copilot in-line chat.
- ☐ E By highlighting the code you want to fix, right-clicking, and selecting "Refactor using GitHub Copilot."

92 How does GitHub Copilot assist developers in minimizing context switching?

- ☐ A GitHub Copilot can automatically handle project management tasks.
- ☐ B GitHub Copilot can completely replace the need for human collaboration.
- ☐ C GitHub Copilot can predict and prevent bugs before they occur.
- ☐ D GitHub Copilot allows developers to stay in their IDE.

93 When crafting prompts for GitHub Copilot, what is a recommended strategy to enhance the relevance of the generated code?

- ☐ A Keep the prompt as short as possible, using single words or brief phrases.
- ☐ B Write the prompt in natural language without any programming language.
- ☐ C Avoid mentioning the programming language to allow for more flexible suggestions.
- ☐ D Provide examples of expected input and output within the prompt.

94 Which two techniques does GitHub recommend for improving prompts to GitHub Copilot?

Select two.

- ☐ A Be specific about your requirements and success criteria
- ☐ B Keep every file in the project open so Copilot has all available context
- ☐ C Combine several related tasks into one long prompt so Copilot sees the whole picture
- ☐ D Provide examples of the input, output, or implementation you want

95 An independent contractor develops applications for a variety of different customers. Assuming no concerns from their customers, which GitHub Copilot plan is best suited?

- ☐ A GitHub Copilot Individual
- ☐ B GitHub Copilot Business
- ☐ C GitHub Copilot Business for non-GHE Customers
- ☐ D GitHub Copilot Enterprise
- ☐ E GitHub Copilot Teams

96 How does GitHub Copilot Enterprise assist in code reviews during the pull request process?

Select two.

- ☐ A It automatically merges pull requests after an automated review.
- ☐ B It generates a prose summary and a bulleted list of key changes for pull requests.
- ☐ C It can validate the accuracy of the changes in the pull request.
- ☐ D It can answer questions about the changeset of the pull request.

97 Why is it important to ensure the security of the code used in Generative AI (Gen AI) tools?

- ☐ A Ensuring code security prevents unauthorized access and potential data breaches.
 - ☐ B Ensuring code security supports the development of more advanced AI features.
 - ☐ C Ensuring code security enables the AI system to handle larger datasets effectively.
 - ☐ D Ensuring code security maintains the integrity of the AI system.
-

98 What are the additional checks that need to pass before the GitHub Copilot responses are submitted to the user?

Each correct answer presents part of the solution. Choose two.

- ☐ A Code quality
 - ☐ B Compatibility with user-specific settings
 - ☐ C Suggestions matching public code (optional based on settings)
 - ☐ D Performance benchmarking
-

99 Are there any limitations to consider when using GitHub Copilot for code refactoring?

- ☐ A GitHub Copilot may not always produce optimized or best-practice code for refactoring.
 - ☐ B GitHub Copilot can only be used with a limited set of programming languages.
 - ☐ C GitHub Copilot always produces bug-free code during refactoring.
 - ☐ D GitHub Copilot understands the context of your entire project and refactors code accordingly.
-

100 Which Copilot Enterprise features are available in all commercially supported IDEs?

Choose two.

- ☐ A Inline suggestions
 - ☐ B Pull request summaries
 - ☐ C Knowledge bases
 - ☐ D Chat
-

101 What content can be configured to be excluded with content exclusions?

Each correct answer presents part of the solution. Choose three.

- ☐ A Files
- ☐ B Folders
- ☐ C Lines in files
- ☐ D Gists
- ☐ E Repositories

102 What is a key consideration when relying on GitHub Copilot Chat's explanations of code functionality and proposed improvements?

- ☐ A The explanations are dynamically updated based on user feedback.
- ☐ B Reviewing and validating the generated output for accuracy and completeness.
- ☐ C GitHub Copilot Chat uses a static database for generating explanations.
- ☐ D The explanations are primarily derived from user-provided documentation.

103 What GitHub Copilot pricing plan gives you access to your company's knowledge bases?

- ☐ A GitHub Copilot Individual
- ☐ B GitHub Copilot Business
- ☐ C GitHub Copilot Enterprise
- ☐ D GitHub Copilot Professional

104 Which of the following is a risk associated with using AI?

- ☐ A AI algorithms are incapable of perpetuating existing biases.
- ☐ B AI systems can sometimes make decisions that are difficult to interpret.
- ☐ C AI eliminates the need for data privacy regulations.
- ☐ D AI replaces the need for developer opportunities in most fields.

105 What practices enhance the quality of suggestions provided by GitHub Copilot?

Select three.

- ☐ A Clearly defining the problem or task
- ☐ B Including personal information in the code comments
- ☐ C Using meaningful variable names
- ☐ D Providing examples of desired output
- ☐ E Use a .gitignore file to exclude irrelevant files

106 How does GitHub Copilot Chat help in understanding the existing codebase?

- ☐ A By running code linters and formatters.
- ☐ B By providing visual diagrams of the code structure.
- ☐ C By answering questions about the code and generating explanations.
- ☐ D By automatically refactoring code to improve readability.

107 What is the correct way to exclude specific files from being used by GitHub Copilot Business during code suggestions?

- ☐ A Modify the .gitignore file to include the specific files.
- ☐ B Add the specific files to a copilot.ignore file.
- ☐ C Use the GitHub Copilot settings in the user interface to exclude files.
- ☐ D Rename the files to include the suffix _no_copilot.

108 In what ways can GitHub Copilot support a developer during the code refactoring process?

Each correct answer presents part of the solution. Choose two.

- ☐ A By offering code transformation examples that enhance performance and reduce complexity.
- ☐ B By independently ensuring compliance with regulatory standards across industries.
- ☐ C By providing suggestions for improving code readability and maintainability based on best practices.
- ☐ D By autonomously refactoring entire codebases to the latest programming language.

109 Identify the right use cases where GitHub Copilot Chat is most effective.

Each correct answer presents part of the solution. Choose two.

- ☐ A Create a technical requirement specification from the business requirement documentation
- ☐ B Explain a legacy COBOL code and translate the code to another language like Python.
- ☐ C Creation of a unit test scenario for newly developed Python code
- ☐ D Creation of end-to-end performance testing scenarios for a web application

110 What is the process behind identifying public code matches when using a public code filter enabled in GitHub Copilot?

- ☐ A Running code suggestions through filters designed to detect public code
- ☐ B Comparing suggestions against public code using machine learning.
- ☐ C Analyzing the context and structure of the code being written
- ☐ D Reviewing the user's browsing history to identify public repositories

111 What are the different ways to give context to GitHub Copilot to get more precise responses?

Each correct answer presents part of the solution. Choose two.

- ☐ A Utilize to interpret developer's thoughts and intentions without any code or comments.
- ☐ B Engage with chat participants such as @workspace to incorporate collaborative context into the responses.
- ☐ C Access developer's previous projects and code repositories to understand their coding style without explicit permission.
- ☐ D Utilize chat variables like *file to anchor the conversation within the specific context of the files or editors in use.

112 How can GitHub Copilot be limited when it comes to suggesting unit tests?

- ☐ A GitHub Copilot can generate all types of unit tests, including those for edge cases and complex integration scenarios.
- ☐ B GitHub Copilot primarily suggests basic unit tests that focus on core functionalities, often requiring additional input from developers for comprehensive coverage.
- ☐ C GitHub Copilot can handle any complexity in code and automatically generate appropriate unit tests.
- ☐ D GitHub Copilot's limitations in generating unit tests can vary based on the IDE version you are using.

113 What caution should developers exercise when using GitHub Copilot for assistance with mathematical computations?

- ☐ A GitHub Copilot's capability to optimize complex mathematical algorithms beyond manual coding.
- ☐ B GitHub Copilot's ability to execute and verify mathematical results in real-time.
- ☐ C GitHub Copilot's reliance on pattern-based responses without verifying computation accuracy.
- ☐ D GitHub Copilot's automatic update of outdated mathematical formulas to modern standards.

114 What are the potential limitations of GitHub Copilot in maintaining existing codebases?

- ☐ A GitHub Copilot can independently manage and resolve all merge conflicts in version control.
- ☐ B GitHub Copilot might not fully understand the context and dependencies within a large codebase.
- ☐ C GitHub Copilot's suggestions are always aware of the entire codebase.
- ☐ D GitHub Copilot can refactor and optimize the entire codebase up to 10,000 lines of code.

115 What are the potential limitations of GitHub Copilot Chat?

Each correct answer presents part of the solution. Choose two.

- ☐ A Limited training data
 - ☐ B No biases in code suggestions
 - ☐ C Ability to handle complex code structures
 - ☐ D Extensive support for all programming languages
-

116 In what ways can GitHub Copilot contribute to the design phase of the Software Development Life Cycle (SDLC)?

- ☐ A GitHub Copilot can independently create a complete software design.
 - ☐ B GitHub Copilot can suggest design patterns and best practices relevant to the project.
 - ☐ C GitHub Copilot can manage design team collaboration and version control.
 - ☐ D GitHub Copilot can generate user interface (UI) prototypes without prompting.
-

117 How can you improve the context used by GitHub Copilot?

Each correct answer presents part of the solution. Choose two.

- ☐ A By opening the relevant tabs in your IDE
- ☐ B By adding relevant code snippets to your prompt
- ☐ C By adding the important files to your .gitconfig
- ☐ D By adding the full file paths to your prompt of important files

Part 2: Answers and explanations

A quick answer key first, then the full reasoning for each question.

1	D	2	A	3	A	4	A	5	D	6	C	7	AB	8	A	9	A
10	C	11	C	12	D	13	D	14	B	15	BC	16	A	17	A	18	AC
19	A	20	B	21	B	22	D	23	D	24	AD	25	BE	26	A	27	C
28	BC	29	AD	30	AB	31	AB	32	AB	33	B	34	C	35	D	36	C
37	B	38	AB	39	D	40	AC	41	C	42	B	43	C	44	C	45	C
46	A	47	B	48	B	49	AD	50	A	51	B	52	B	53	CD	54	B
55	C	56	AE	57	AB	58	A	59	C	60	E	61	BCE	62	CD	63	C
64	A	65	A	66	C	67	AB	68	C	69	C	70	A	71	AB	72	ACD
73	A	74	BE	75	B	76	D	77	A	78	BC	79	CD	80	D	81	C
82	AB	83	B	84	B	85	C	86	B	87	B	88	C	89	B	90	D
91	AE	92	D	93	D	94	AD	95	A	96	BD	97	A	98	AC	99	A
100	AD	101	ABE	102	B	103	C	104	B	105	ACD	106	C	107	C	108	AC
109	BC	110	A	111	BD	112	B	113	C	114	B	115	AC	116	B	117	AB

- 1 A social media manager wants to use AI to filter content. How can they promote transparency in the platform's AI operations?

ANSWER: D

- D** By providing clear explanations about the types of content the AI is designed to filter and how it arrives at its conclusion.

Transparency, one of Microsoft and GitHub's six responsible AI principles, means being open about how an AI system works. Providing clear explanations of what the AI filters and how it reaches conclusions is the textbook example of transparency. Options A, B, and C relate to performance or reputation, not to making the system understandable to users.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

2 What GitHub Copilot configuration needs to be enabled to protect against IP infringements?

ANSWER: A

A Blocking public code matches

Setting the 'Suggestions matching public code' policy to Block causes Copilot to compare a suggestion plus about 150 characters of surrounding context against public code on GitHub and withhold any match, which mitigates intellectual property and licensing concerns. Allowing matches (C) does the opposite. There is no separate 'license check' toggle, so B and D describe settings that do not exist.

Source: [GitHub Docs - Managing GitHub Copilot policies as an individual subscriber](#)

3 How long does it take content exclusion to add or be updated?

ANSWER: A

A Up to 30 minutes

The documentation states that after you add or change content exclusions it can take up to 30 minutes to take effect in IDEs where the settings are already loaded, though you can force an immediate reload by restarting the IDE or reloading the window. The other time ranges are not values GitHub publishes.

Source: [GitHub Docs - Excluding content from GitHub Copilot](#)

4 Which GitHub Copilot plan allows for prompt and suggestion collection?

ANSWER: A

A GitHub Copilot Individuals

GitHub only collects prompts, suggestions, code snippets, and context for model training under the individual plans (Copilot Free, Pro, Pro+, and Max), and even then users can opt out. Copilot Business and Copilot Enterprise data is protected under GitHub's Data Protection Agreement and is explicitly not used or retained for this purpose, so the collection setting is not even shown for those seats.

Source: [GitHub Docs - Managing GitHub Copilot policies as an individual subscriber](#)

5 What is the primary purpose of organization audit logs in GitHub Copilot Business?

ANSWER: D

D To monitor administrator activities and actions within the organization

The Copilot audit log records administrative actions such as changes to Copilot settings and policies, seat and license assignments and removals, and agent activity, so its purpose is monitoring actions and administrative activity in the organization or enterprise. It does not count lines of code (A), it is not itself the license-assignment tool (B), and it does not detect code conflicts (C).

Source: [GitHub Docs - Reviewing audit logs for GitHub Copilot](#)

6 Which of the following is correct about GitHub Copilot Knowledge Bases?

ANSWER: C

- C** It is an Enterprise feature

Knowledge bases are a Copilot Enterprise feature, requiring a Copilot Enterprise subscription and managed at the organization or enterprise level in GitHub Enterprise Cloud. Indexing is not limited to all repos or all file types, and it is refreshed rather than purely static, which rules out A, B, and D.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

7 GitHub Copilot in the Command Line Interface (CLI) can be used to configure the following settings:

ANSWERS: A, B

- A** The default execution confirmation
B Usage analytics

The GitHub Copilot in the CLI configuration menu let you set the default execution confirmation behavior and toggle optional usage analytics. It did not configure the default editor or GitHub CLI subcommands, so C and D are wrong. Note the original gh copilot extension has since been replaced by the newer GitHub Copilot CLI, but these two config options remain the intended answer.

Source: [GitHub Docs - Using GitHub Copilot in the command line](#)

8 What is used by GitHub Copilot in the IDE to determine the prompt context?

ANSWER: A

- A** Information from the IDE like open tabs, cursor location, selected code.

In the IDE Copilot builds prompt context from signals available in the editor, such as open tabs and files, the cursor location, and selected code, which is why GitHub advises opening relevant files and closing irrelevant ones. It does not ingest the entire repository or all submodules (B), nor everything visible or the terminal's folder (C, D).

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

9 What is zero-shot prompting?

ANSWER: A

- A** Only giving GitHub Copilot a question as a prompt and no examples

Zero-shot prompting means giving Copilot only the instruction or question with no worked examples. Providing examples would be one-shot or few-shot prompting. The option about supplying as little context as possible is a distractor: zero-shot refers specifically to the absence of examples, not to minimizing all context.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

10 How does GitHub Copilot Chat utilize its training data and external sources to generate responses when answering coding questions?

ANSWER: C

- C** It combines its training data set, code in user repositories, and external sources like Bing to generate responses.

Copilot Chat generates answers by drawing on the model's training data, the code and context in your repositories and workspace, and, when web search is enabled, external sources. Because it uses more than one of these, the single-source options are incomplete.

Source: [GitHub Docs - Managing GitHub Copilot policies](#)

11 When using GitHub Copilot Chat to generate boilerplate code for various test types, how can you guide the AI to follow the testing standards of your company?

ANSWER: C

- C** By using specific prompt examples in your chat request.

To steer Copilot toward your company's testing standards you include concrete examples in your chat prompt, such as sample test structures or existing tests, since Copilot uses examples to infer the desired pattern. There is no configuration setting, terminal command, or single slash command that encodes bespoke testing standards.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

12 What can be done during AI development to minimize bias?

ANSWER: D

- D** Use diverse data, fairness metrics, and human oversight.

Bias is reduced by using diverse and representative training data, applying fairness metrics, and keeping humans in the loop for oversight. Chasing computational speed, narrow accuracy, or simply collecting massive data volumes does not by itself address bias and can even amplify it.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

13 Why might a Generative AI (Gen AI) tool create inaccurate outputs?

ANSWER: D

- D** The training data might contain biases or inconsistencies.

Generative AI can produce inaccurate outputs because its training data may contain biases, gaps, or inconsistencies that the model reproduces. Server load, downtime, or a deliberate preference for creativity over accuracy are not the documented root causes of factual inaccuracy or hallucination.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

14 How does GitHub Copilot typically handle code suggestions that involve deprecated features or syntax of programming languages?

ANSWER: B

- B** GitHub Copilot may suggest deprecated syntax or features if they are present in its training data.

Copilot generates suggestions based on patterns in its training data, so if deprecated syntax or features appear in that data it may suggest them. It does not automatically modernize code, guarantee filtering out deprecated elements, or refuse prompts involving deprecated features. This is why suggestions must be reviewed for currency.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

15 How long does GitHub retain Copilot data for Business and Enterprise?

ANSWERS: B, C

- B** Prompts and Suggestions: Retained for 28 days
C User Engagement Data: Kept for Two Years

For Copilot Business and Enterprise, GitHub retains prompts and suggestions for Copilot Chat for 28 days and retains user engagement data for two years. Raw code-completion prompts are handled transiently, but for the two-answer form of this question the documented figures are 28 days and two years.

Source: [GitHub Docs - Managing GitHub Copilot policies](#)

16 What is the best way to share feedback about GitHub Copilot Chat when using it on GitHub Mobile?

ANSWER: A

- A** Use the emojis in the Copilot Chat interface.

In GitHub Mobile you give feedback on a specific Copilot Chat response using the thumbs up and thumbs down reaction icons attached to that response. Tweeting, the general GitHub website feedback area, and the app's Settings menu are not the intended per-response feedback mechanism.

Source: [GitHub Docs - Asking GitHub Copilot questions in GitHub Mobile](#)

17 How can GitHub Copilot assist in maintaining consistency across your tests?

ANSWER: A

- A** By identifying a pattern in the way you write tests and suggesting similar patterns for future tests.

Copilot promotes test consistency by recognizing the patterns in how you already write tests and suggesting similar patterns for new tests. It does not automatically fix every test, its value here is not documentation references, and writing implementation code is a different task from maintaining test consistency.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

18 What are the potential risks associated with relying heavily on code generated from GitHub Copilot?

ANSWERS: A, C

- A** GitHub Copilot may introduce security vulnerabilities by suggesting code with known exploits.
- C** GitHub Copilot's suggestions may not always reflect best practices or the latest coding standards.

Documented risks of over-relying on Copilot output include introducing security vulnerabilities such as insecure or known-exploitable patterns, and suggestions that may not reflect current best practices or the latest standards. The other options invert the usual benefit of Copilot, which is to speed work up rather than slow it down.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

19 How does GitHub Copilot suggest code optimizations for improved performance?

ANSWER: A

- A** By analyzing the codebase and suggesting more efficient algorithms or data structures.

Copilot helps with performance by analyzing your code and suggesting more efficient algorithms or data structures, which you then choose to accept. It does not autonomously rewrite your whole codebase, enforce coding standards, or generate performance reports; the human stays in control of applying any optimization.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

20 Which of the following scenarios best describes the intended use of GitHub Copilot Chat as a tool?

ANSWER: B

B A productivity tool that provides suggestions, but relying on human judgment.

Copilot Chat is intended as a productivity aid that offers suggestions while the developer applies human judgment and remains responsible for reviewing and approving output. It is explicitly not a replacement for developers, not an autonomous solution without oversight, and not limited to debugging.

Source: [GitHub Docs - About GitHub Copilot](#)

21 How can GitHub Copilot assist developers during the requirements analysis phase of the Software Development Life Cycle (SDLC)?

ANSWER: B

B By providing templates and code snippets that help in documenting requirements.

In the requirements analysis phase Copilot acts as an assistant, offering templates, boilerplate structures, and code snippets that help developers document and scaffold requirements. It does not autonomously author complete requirements documents, resolve requirement conflicts, or manage stakeholder meetings, which are human coordination tasks outside its scope.

Source: [Microsoft Learn - Study guide for Exam GH-300](#)

22 How does GitHub Copilot assist developers in reducing the amount of manual boilerplate code they write?

ANSWER: D

D By suggesting code snippets that can be reused across different parts of the project.

Copilot reduces boilerplate by suggesting reusable code snippets and completing repetitive patterns inline as you type. It does not do real-time multi-developer collaboration, predict and pre-generate future requirements, or refactor an entire codebase without developer input; the developer always reviews each suggestion.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

23 How can users provide feedback about GitHub Copilot Chat using their IDE?

ANSWER: D

D Through the "Share Feedback" button in the Copilot Chat panel

Copilot Chat exposes feedback controls directly inside the IDE panel, including thumbs up and down and a Share Feedback option, so users can rate responses in context. The other channels such as a web form, email, or community forums are not the in-IDE mechanism the question asks about.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

24 Which GitHub Copilot pricing plans include features that exclude your GitHub Copilot data like usage, prompts, and suggestions from default training GitHub Copilot?

ANSWERS: A, D

- A** GitHub Copilot Business
- D** GitHub Copilot Enterprise

For Copilot Business and Copilot Enterprise, prompts, suggestions, and usage data are not used to train the foundation models by default, which is the data protection the question describes. Individual plan users must opt out manually, and Copilot Codespace is not a real plan.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

25 Which statements are true about the GitHub Copilot usage metrics API?

ANSWERS: B, E

- B** The API can track the acceptance rate of code suggestions accepted and used in the organization.
- E** The API can provide Copilot Chat specific suggestions acceptance metrics.

The Copilot usage metrics API reports aggregate engagement data including code suggestion acceptance counts and rates and Copilot Chat specific metrics such as chat acceptances. It cannot refactor code, assess code quality, or judge coding style compliance; it only reports usage telemetry.

Source: [GitHub Docs - REST API endpoints for Copilot usage metrics](#)

26 If you are working on open source projects, GitHub Copilot Individual can be paid:

ANSWER: A

- A** Based on the payment method in your user profile

An individual Copilot subscription is charged to the payment method stored on your personal GitHub account. It is not free for all open source work, since only verified maintainers of popular projects and students get it free, it is not paid by invoice, which is an organization option, and it is not billed through an Azure subscription.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

27 What is a likely effect of GitHub Copilot being trained on commonly used code patterns?

ANSWER: C

- C** Suggest code snippets that reflect the most common practices in the training data.

Because Copilot is trained on large volumes of publicly available code, its suggestions tend to reflect the most common patterns and practices present in that training data. It does not reliably invent novel unpopular solutions or entirely new projects, and a diverse dataset would not force homogeneous output.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

28 Which scenarios can GitHub Copilot Chat be used to increase productivity?

ANSWERS: B, C

- B** Create a documentation file for the newly created code base.
- C** A developer is added to a new project and would like to understand the current software code.

Copilot Chat boosts productivity by generating documentation for an existing codebase and by explaining unfamiliar code so a newly onboarded developer can understand it quickly. It is not a project management application and does not perform release management or push code to production.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

29 What method can a developer use to generate sample data with GitHub Copilot?

ANSWERS: A, D

- A** Utilizing GitHub Copilot's ability to create fictitious information from patterns in training data.
- D** Leveraging GitHub Copilot's suggestions to create data based on API documentation in the repository.

Copilot can generate sample or fictitious data by drawing on patterns learned from its training data and by inferring realistic structures from API documentation or schemas present in the repository. It cannot independently spin up and manage storage services or connect to and query live databases; it only suggests code and content.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

30 What are the effects of content exclusions?

ANSWERS: A, B

- A** The excluded content is not directly available to GitHub Copilot to use as context.
- B** GitHub Copilot suggestions are no longer available in the excluded files.

The content exclusion documentation states that inline suggestions are not available in affected files and that excluded content will not inform Copilot's suggestions or responses, meaning it is not available as context. The exclusion is about access and suggestions, not about debugging behavior or suggestion counting.

Source: [GitHub Docs - Content exclusion for GitHub Copilot](#)

31 What types of prompts or code snippets might be flagged by the GitHub Copilot toxicity filter?

ANSWERS: A, B

- A** Hate speech or discriminatory language (e.g., racial slurs, offensive stereotypes)
- B** Sexually suggestive or explicit content

Copilot's content filtering safety system scans prompts and responses to block harmful content such as hate speech and discriminatory language and sexually explicit content. Logical errors and opinionated code comments are not toxic content and are not what the content filter targets.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

32 When using an IDE with a supported GitHub Copilot plug-in, which Chat features can be accessed from within the IDE?

ANSWERS: A, B

- A** Explain code and suggest improvements
- B** Generate unit tests

Within a supported IDE, Copilot Chat can explain code and suggest improvements and generate unit tests for the code you are working on. Finding out about releases and commits is a GitHub.com and Copilot Enterprise knowledge feature rather than a core in-IDE chat capability.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

33 Which principle emphasizes that AI systems should be understandable and provide clear information on how they work?

ANSWER: B

- B** Transparency

Transparency is the responsible AI principle that requires AI systems to be understandable, with clear information about how they work and their limitations. Fairness concerns equitable treatment, inclusiveness concerns empowering everyone, and accountability concerns human responsibility for the system.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

34 What is one of the recommended practices when using GitHub Copilot Chat to enhance code quality?

ANSWER: C

- C** Regularly review and refactor the code suggested by Copilot.

A recommended practice is to regularly review and refactor Copilot's suggestions, since AI output must be validated and improved by the developer. Avoiding Copilot for complex tasks and disabling inline suggestions reduce its usefulness, and relying blindly on suggestions without review directly contradicts responsible use guidance.

Source: [Microsoft Learn - Study guide for Exam GH-300](#)

35 How can the concept of fairness be integrated into the process of operating an AI tool?

ANSWER: D

D Training AI data and algorithms to be free from biases will ensure fairness.

Fairness is integrated by training the AI's data and algorithms to be free from bias so outputs do not disadvantage particular groups. Accessibility, large datasets alone, and performance monitoring may support quality or inclusiveness but do not by themselves ensure the bias-free treatment that defines fairness.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

36 How does GitHub Copilot Chat help to fix security issues in your codebase?

ANSWER: C

C By annotating the given suggestions with known vulnerability patterns.

Copilot Chat helps with security by flagging or annotating suggestions against known vulnerability patterns and pointing out insecure code so the developer can fix it. It does not enforce coding standards to guarantee no vulnerabilities, produce full vulnerability scan reports like code scanning, or automatically refactor an entire codebase.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

37 Which Microsoft ethical AI principle is aimed at ensuring AI systems treat all people equally?

ANSWER: B

B Fairness

Fairness is the responsible AI principle focused on treating all people equally and avoiding bias. Inclusiveness is about empowering and engaging everyone, reliability and safety is about consistent safe operation, and privacy and security is about protecting data, so none of those match the equal-treatment description.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

38 How is GitHub Copilot Individual billed?

ANSWERS: A, B

A Monthly as a subscription

B Annually as a subscription

The individual Copilot plan is offered as a subscription billed either monthly or annually. It is not a metered pay-per-use service, and it is not free for all open source projects; free access is limited to verified maintainers of popular projects and to students.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

39 What is the impact of the "Fill-In-the-Middle" (FIM) technique on GitHub Copilot's code suggestions?

ANSWER: D

- D** Improves suggestions by considering both the prefix and suffix of the code, filling in the middle part more accurately.

Fill-In-the-Middle improves suggestion quality by giving the model both the prefix, meaning the code before the cursor, and the suffix, meaning the code after it, so it can fill the gap in a way that fits the surrounding context. Options restricting it to external databases, the prefix only, or comments only misdescribe the technique.

Source: [Microsoft Learn - Study guide for Exam GH-300](#)

40 What two options navigate to configure duplicate detection?

ANSWERS: A, C

- A** Enterprise settings Copilot Policies
C Organization settings Copilot Policies

Duplicate detection, formally the Suggestions matching public code policy, is configured through Copilot policies at the enterprise level and the organization level. There is no repository-level or user-level Copilot policy for this setting.

Source: [GitHub Docs - Feature availability when GitHub Copilot policies conflict](#)

41 Why is code reviewing still necessary when using GitHub Copilot to write tests?

ANSWER: C

- C** Because GitHub Copilot's generated test cases may not cover all possible scenarios.

GitHub's own Copilot Chat application card states that generated test cases may not cover all possible scenarios and that manual testing and code review are still necessary. Copilot cannot guarantee complete coverage or replace human review, which rules out the other options as overstatements.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

42 What role does the pre-processing of user input play in the data flow of GitHub Copilot Chat?

ANSWER: B

- B** It enriches the input prompt with additional context before passing it to the language model.

The Copilot Chat data flow describes an input processing step where the user's prompt is pre-processed and combined with contextual information such as the current repository, open files, and chat history before being sent to the language model. Formatting the output happens at the end, and the model rather than pre-processing generates the response.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

43 How do you generate code suggestions with GitHub Copilot in the CLI?

ANSWER: C

- C** Use `gh copilot suggest` Write the command you want Select the best suggestion from the list.

In the CLI you invoke Copilot with a `suggest` command, describe the command you want in natural language, and pick from the returned suggestions. The other options reference invented commands such as `copilot refine` and `copilot generate`. Note that `gh copilot suggest` is the classic documented workflow, and GitHub has since also shipped a newer native Copilot CLI.

Source: [GitHub Docs - Installing GitHub Copilot CLI](#)

44 How can GitHub Copilot assist with code refactoring tasks?

ANSWER: C

- C** GitHub Copilot can suggest refactoring improvements for better code quality.

Copilot Chat can suggest improvements such as better error handling or more readable control flow, but it suggests rather than acting autonomously. Options claiming automatic unattended behavior are wrong, and file management is not something Copilot does for refactoring.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

45 What is the correct way to access the audit log events for GitHub Copilot Business?

ANSWER: C

- C** Use the Audit log section in the organization's GitHub settings

Copilot audit events are reviewed through the Audit log section in the organization's settings, not the Security or Insights tabs and not the repository Code tab. The audit log is the dedicated location where Copilot Business administrative events are recorded.

Source: [GitHub Docs - Audit log events for your enterprise](#)

46 When can GitHub Copilot still use content that was excluded using content exclusion?

ANSWER: A

- A** If the contents of an excluded file are referenced in code that is not excluded, for example function calls.

The documented limitation is that Copilot may still use semantic information from an excluded file when the IDE surfaces it indirectly, for example type information and hover definitions for symbols used in code that is not excluded. Content exclusions are not user-overridable, organization settings do not loosely override enterprise exclusions, and workspace prompts do not bypass exclusions.

Source: [GitHub Docs - Content exclusion for GitHub Copilot](#)

47 How can the insights gained from the metrics API be used to improve the development process in conjunction with GitHub Copilot?

ANSWER: B

B Insights on the types of coding languages where GitHub Copilot is most helpful.

The Copilot metrics API returns breakdowns of acceptances and active users by language and editor, so the concrete documented insight is understanding which coding languages Copilot is most helpful with. The API does not compare Copilot suggestions against manual coding productivity; it only reports suggestion and acceptance counts.

Source: [GitHub Docs - REST API endpoints for Copilot usage metrics](#)

48 In what way can GitHub Copilot and GitHub Copilot Chat aid developers in modernizing applications?

ANSWER: B

B GitHub Copilot can suggest modern programming patterns based on your code.

Copilot suggests modern programming patterns based on the code you already have, which is the accurate and measured capability. The other options overstate it by claiming direct legacy-to-cloud-native conversion, a full-stack build and deploy from one query, or alignment with standards that do not yet exist.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

49 How can GitHub Copilot facilitate a smoother learning experience when diving into a new programming language?

ANSWERS: A, D

A GitHub Copilot Chat can provide guidance and support for common coding tasks and challenges in the targeted programming language.

D GitHub Copilot can convert comments into code to grasp the syntax and nuances of a new programming language.

Copilot Chat can guide you through common coding tasks in a target language, and Copilot can turn comments into code so you learn the syntax and idioms. There is no understand slash command, and the organization documentation option describes the Copilot Enterprise knowledge base feature, which is about internal context rather than learning a new language.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

-
- 50** A company is currently storing code in Bitbucket and would like to use GitHub Copilot. Which GitHub Copilot plan will be most cost effective to allow them to manage users with their Identity Provider (e.g. Okta)?

ANSWER: A

- A** GitHub Copilot Business for non-GHE customers

Copilot Business provides centralized license and policy management and is available to organizations that are not on GitHub Enterprise, at a lower per-seat price than Enterprise. Since the company only needs to manage users through their identity provider, Business is the most cost-effective plan. Copilot Enterprise is designed for GitHub Enterprise Cloud customers and adds features they did not ask for.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

- 51** How does the `/tests slash` command assist developers?

ANSWER: B

- B** Creates unit tests for the selected code.

The chat cheat sheet defines the tests slash command as generating unit tests for the selected code. It does not build test documentation, integrate external frameworks, or execute tests.

Source: [GitHub Docs - GitHub Copilot Chat cheat sheet](#)

- 52** What is the primary role of the `/optimize slash` command in Visual Studio?

ANSWER: B

- B** Enhances the performance of the selected code by analyzing its runtime complexity.

In Visual Studio the optimize command is documented as analyzing and improving the running time of the selected code, matching the description of enhancing performance by analyzing runtime complexity. It does not translate to another language, reformat to a style guide, or summarize documentation.

Source: [GitHub Docs - GitHub Copilot Chat cheat sheet](#)

- 53** What do you check when GitHub Copilot content exclusions are not working?

ANSWERS: C, D

- C** If the content exclusion settings changed in the last 30 minutes or before that.
D If the user is in an organization that has content exclusions configured.

When exclusions do not seem to work you check whether the user actually belongs to an organization that has content exclusions configured, and whether the settings were changed recently, since exclusion changes can take up to about 30 minutes to propagate. The other options describe concepts that do not exist.

Source: [GitHub Docs - Content exclusion for GitHub Copilot](#)

54 How can you get multiple suggestions from GitHub Copilot?

ANSWER: B

B By opening the completions panel in your editor

You can open the Copilot completions panel in your editor to see multiple alternative suggestions at once. There is no comment-driven multiple suggestions request, no inline chat multiple command, and workspace context supplies information rather than producing multiple completions.

Source: [GitHub Docs - GitHub Copilot Chat cheat sheet](#)

55 Which of the following GitHub Copilot Business related activities can be tracked using the organization audit logs?

ANSWER: C

C Changes to content exclusion settings

Administrative actions such as changes to content exclusion settings are recorded as Copilot audit log events. Individual code suggestions, accepted chat suggestions, and suggestions blocked by duplication detection are not audit log events.

Source: [GitHub Docs - Audit log events for your enterprise](#)

56 Which REST API endpoint is used to modify details about a GitHub Copilot Business subscription?

ANSWERS: A, E

A Add teams to the Copilot subscription for an organization

E Remove teams from the Copilot subscription for an organization

The Copilot user management REST API includes endpoints to add teams to and remove teams from an organization's Copilot subscription, which are the two real ways to modify the subscription's membership. Upgrading or downgrading a tier, migrating seats between organizations, and reassigning seats by repository size are not Copilot REST endpoints.

Source: [GitHub Docs - REST API endpoints for Copilot user management](#)

57 Which of the following prompts can be used to guide GitHub Copilot Chat in refactoring code for quality improvements?

ANSWERS: A, B

A "Show me how to improve the readability of this function."

B "Suggest ways to enhance the maintainability of this code segment."

Prompts that ask Copilot to improve the readability of a specific function and enhance the maintainability of a code segment are focused, actionable refactoring requests. Asking it to meet the latest coding standards across an entire application is too broad, and asking it to predict future trends is not a code-quality refactoring task.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

58 What is few-shot prompting?

ANSWER: A

- A** Giving GitHub Copilot several examples of the input and output you want, so it can infer the pattern before it responds

Few-shot prompting means presenting Copilot with several examples so it can recognize the pattern you want and apply it, which Microsoft Learn describes as striking a balance between zero-shot unpredictability and the precision of fine-tuning. Generating code from a description with no examples is zero-shot prompting, and a single example is one-shot. Naming sources and retrying the prompt are not prompting techniques at all.

Source: [Microsoft Learn - Prompt engineering foundations and best practices](#)

59 Where is the proxy service hosted?

ANSWER: C

- C** Microsoft Azure

GitHub Copilot's core serving infrastructure, including its proxy service, runs on GitHub's Microsoft Azure infrastructure. It is not self-hosted, and it is not AWS or GCP for the proxy layer, though some third-party bring-your-own-key models may run elsewhere.

Source: [GitHub Docs - Hosting of models for GitHub Copilot](#)

60 A team is using GitHub Copilot Individual in their daily development activities. They need to exclude specific files from being used to inform code completion suggestions. How can they achieve this?

ANSWER: E

- E** Upgrade to Copilot Business

Content exclusions are a feature of Copilot Business and Copilot Enterprise only, so an individual subscriber must upgrade to Copilot Business. A .gitignore file does not control Copilot, a file chat variable adds rather than excludes context, and configuring exclusions through an organization owner or repository admin is not available because the feature is not part of the individual plan.

Source: [GitHub Docs - Content exclusion for GitHub Copilot](#)

61 What reasons could apply if code suggestions are not working in your editor?

ANSWERS: B, C, E

- B** You do not have an active internet connection
- C** Your programming language is not supported
- E** You do not have a valid GitHub Copilot license

Copilot needs an active internet connection, a supported programming language, and a valid license or subscription to produce inline suggestions, so those three are genuine causes. Working in a file listed in .gitignore does not disable suggestions. Content exclusion can block suggestions in specific files, but it is a Business and Enterprise scoped feature and is framed here as blocking Copilot entirely, which overstates its effect.

Source: [GitHub Docs - Troubleshooting common issues with GitHub Copilot](#)

62 How does GitHub Copilot identify matching code and ensure that public code is appropriately handled or blocked?

ANSWERS: C, D

- C** Filtering out suggestions that match code from public repositories
- D** Implementing safeguards to detect and avoid suggesting verbatim snippets from public code

Copilot's duplication detection filter can be configured to filter out suggestions that match public code, and Copilot includes safeguards to detect and avoid producing verbatim snippets from public repositories. It was not trained only on private repositories, and it does not store user private repository data to build future suggestions.

Source: [GitHub Docs - Managing policies and features for GitHub Copilot in your organization](#)

63 Which of the following does GitHub Copilot's LLM derive context from when producing a response?

ANSWER: C

- C** Neighboring or related files within a project

Copilot derives context from the code around your cursor and from neighboring or related open files in the project. Commit frequency, syntax highlighting themes, and the integrated version control system are not signals the model uses to generate a response.

Source: [GitHub Docs - About GitHub Copilot](#)

64 Where can you validate if GitHub Copilot is not returning suggestions because of content exclusions?

ANSWER: A

A The GitHub Copilot icon in the status bar of the editor will display a message

When a file is affected by content exclusion, the Copilot icon in the editor status bar shows a diagonal line through it, and hovering over the icon displays a tooltip explaining which settings applied the restriction. There is no separate errors panel or dedicated GitHub.com Copilot logs page for validating this.

Source: [GitHub Docs - Troubleshooting common issues with GitHub Copilot](#)

65 How does GitHub Copilot Chat ensure that a function works correctly?

ANSWER: A

A By suggesting assertions based on the code's context and semantics.

Copilot Chat can help confirm a function behaves correctly by suggesting assertions and tests based on the code's context and semantics, but the developer still runs and reviews them. Copilot does not execute test cases itself, and writing all tests or the implementation is not how it verifies correctness.

Source: [GitHub Docs - Asking GitHub Copilot questions in your IDE](#)

66 How does GitHub Copilot utilize chat history to enhance its code completion capabilities?

ANSWER: C

C By analyzing past chat interactions to identify common programming patterns and errors.

Copilot uses prior chat interactions as context to recognize recurring programming patterns and mistakes, which improves the relevance of its completions. Logging to monitor compliance and sharing history with third parties are not how history is used, and claims of persistent personalization overstate the behavior.

Source: [GitHub Docs - Asking GitHub Copilot questions in your IDE](#)

67 Identify the steps involved in the life cycle of a GitHub Copilot code suggestion?

ANSWERS: A, B

A Generate suggestions

B Capturing the user's context

The lifecycle of a Copilot suggestion involves capturing the user's context, meaning surrounding code, open files, and the prompt, and then generating candidate suggestions that are filtered before being shown. Processing telemetry, retraining the model, and storing user data are not part of the per-suggestion generation lifecycle.

Source: [GitHub Docs - About GitHub Copilot](#)

68 Which of the following steps correctly demonstrates how to establish an organization-wide policy for GitHub Copilot Business to restrict its use to certain repositories?

ANSWER: C

C Configure the policies in the organization settings

Copilot policies, including restricting features and access, are configured centrally in the organization and enterprise settings, not through per-repository policy files or GitHub Actions. There is no copilot.policy file mechanism.

Source: [GitHub Docs - Managing policies and features for GitHub Copilot in your organization](#)

69 Which of the following statements best describes the impact of GitHub Copilot on the software development process?

ANSWER: C

C It increases productivity by automating repetitive coding tasks.

Copilot's primary documented impact is increasing developer productivity by automating repetitive and boilerplate coding tasks so developers can focus on higher value work. It does not guarantee fewer third party vulnerabilities, it is not primarily a testing automation tool, and it explicitly does not replace developers, who must review all output.

Source: [GitHub Docs - About GitHub Copilot](#)

70 What method can be used to interact with GitHub Copilot?

ANSWER: A

A By using a properly configured GitHub CLI

A properly configured command line is a documented way to interact with Copilot, first through the gh copilot extension and now through the GitHub Copilot CLI that runs from your terminal. Copilot Chat is officially supported in VS Code, Visual Studio, JetBrains, Eclipse, and Xcode, not Neovim, which has only community completion support.

Source: [GitHub Docs - About GitHub Copilot CLI](#)

71 Which Copilot Individual features are available when using a supported extension for Visual Studio, VS Code, or JetBrains IDEs?

ANSWERS: A, B

A Code suggestions

B Chat

In supported IDE extensions the individual Copilot plan provides code completion suggestions and Copilot Chat. Knowledge bases and pull request or diff features are Copilot Enterprise capabilities tied to GitHub.com, not individual IDE features.

Source: [GitHub Docs - About GitHub Copilot](#)

72 What types of content can GitHub Copilot Knowledge Base answer questions about?

ANSWERS: A, C, D

- A** Code snippets
- C** Documentation
- D** Design patterns

A knowledge base is built from Markdown documentation across selected repositories, so Copilot can answer questions grounded in that documentation, including the code snippets and design patterns described in the text. It cannot interpret compiled binaries or screenshots as knowledge base sources.

Source: [GitHub Docs - About GitHub Copilot Spaces](#)

73 What configuration needs to be set to get help from Microsoft and GitHub protecting against IP infringement while using GitHub Copilot?

ANSWER: A

- A** Suggestions matching public code to 'blocked'

To receive IP indemnity under the Copilot Copyright Commitment the duplication detection filter must be enabled, meaning the Suggestions matching public code policy is set to Blocked. Manually reviewing suggestions or license checking are not the configuration that triggers the protection, and there is no license-specific blocking option.

Source: [GitHub Docs - Managing policies and features for GitHub Copilot in your organization](#)

74 Which of the following are true about code suggestions?

ANSWERS: B, E

- B** You can use keyboard shortcuts to accept the next word in a suggestion
- E** Alternative code suggestions can be shown in a new tab

You can accept a suggestion word by word using a keyboard shortcut, and you can open multiple alternative suggestions in a separate tab. Copilot does not guarantee vulnerability-free or compilable code, and suggestions are not limited to single lines.

Source: [GitHub Docs - Troubleshooting common issues with GitHub Copilot](#)

75 What should developers consider when relying on GitHub Copilot for generating code that involves statistical analysis?

ANSWER: B

- B** GitHub Copilot's suggestions are based on statistical trends and may not always apply accurately to specific datasets.

Copilot generates output from statistical patterns learned across large amounts of code, so its suggestions for statistical analysis may not correctly fit a specific dataset and must be validated by the developer. It cannot independently verify statistical significance, automatically correct statistical errors, or invent novel undocumented statistical methods.

Source: [GitHub Docs - About GitHub Copilot](#)

76 What specific function does the `/fix slash` command perform?

ANSWER: D

- D** Proposes changes for detected issues, suggesting corrections for syntax errors and programming mistakes.

The fix slash command proposes changes for problems Copilot detects in your code, suggesting corrections for syntax errors and programming mistakes. It is not a static analysis security review, a pseudocode-to-code converter, or a general new-code generator.

Source: [GitHub Docs - Asking GitHub Copilot questions in your IDE](#)

77 Which of the following describes role prompting?

ANSWER: A

- A** Describing in your prompt what your role is to get a better suggestion

Role prompting means telling Copilot what role or persona to assume, for example acting as a senior security engineer, so it tailors its response. Specifying tone and asking Copilot to explain a suggestion's role are different techniques, and giving multiple examples of the data form describes few-shot prompting.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

78 What kind of insights can the GitHub Copilot usage metrics API provide to help evaluate the effectiveness of GitHub Copilot?

ANSWERS: B, C

- B** The API can track the number of code suggestions accepted and used in the organization.
- C** The API can provide Copilot Chat specific suggestions acceptance metrics.

The Copilot usage metrics API surfaces adoption and engagement data such as the number of code completions suggested and accepted, and Copilot Chat activity and acceptance metrics. The API does not generate code quality reports, refactor code, or grade coding style compliance.

Source: [GitHub Docs - REST API endpoints for Copilot usage metrics](#)

79 Which of the following statements correctly describes how GitHub Copilot Individual uses prompt data?

ANSWERS: C, D

- C** Prompt data is used to train machine learning models for better code suggestions.
- D** Real-time user input helps generate context-aware code suggestions.

For the individual plan, real-time user input such as surrounding code and prompts is used to generate context-aware suggestions, and unless a user opts out code snippet data can be used to help improve the models. Prompt data is not stored unencrypted for speed, and it is not used to improve a search engine.

Source: [GitHub Docs - About GitHub Copilot](#)

80 What is the main purpose of the duplication detection filter in GitHub Copilot?

ANSWER: D

- D** To detect and block suggestions that match public code snippets on GitHub if they contain about 150 characters.

The duplication detection filter detects and blocks suggestions that match public code on GitHub when the match is roughly 150 characters or longer. It does not compare against a private repository, give admins custom visibility controls, or simply coach the user about best practices.

Source: [GitHub Docs - Managing policies and features for GitHub Copilot in your organization](#)

81 An Azure DevOps organization wants to buy GitHub Copilot for its developers and manage their licenses centrally, without paying for GitHub Enterprise licenses. Which plan should it use?

ANSWER: C

- C** GitHub Copilot Business

Copilot Business is the organization plan that does not depend on GitHub Enterprise Cloud: GitHub documents creating an enterprise account solely to allocate Copilot Business licenses, which gives identity-provider integration without paying for GitHub Enterprise licenses. Copilot Enterprise is available only with GitHub Enterprise Cloud, Copilot Pro is an individual plan with no central license management, and there is no plan called GitHub Copilot for Azure DevOps.

Source: [GitHub Docs - About enterprise accounts for Copilot Business](#)

82 What type of information can you retrieve through GitHub Copilot Business Subscriptions via REST API?

ANSWERS: A, B

- A** Get a summary of GitHub Copilot usage for organization members
- B** List all GitHub Copilot seat assignments for an organization

The Copilot REST API for organizations with a Business or Enterprise subscription exposes an endpoint to list all seat assignments and usage metrics endpoints that summarize Copilot usage for organization members. There is no endpoint that returns an individual user's code suggestions or a list of unsubscribed members.

Source: [GitHub Docs - REST API endpoints for Copilot user management](#)

83 Select a strategy to increase the performance of GitHub Copilot Chat.

ANSWER: B

- B** Apply prompt engineering techniques to be more specific

Response quality from Copilot Chat is improved by prompt engineering, being specific and giving context, not by tuning infrastructure or limiting users. Cramming many requirements into one query contradicts the guidance to break complex tasks into smaller, specific prompts.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

84 What is a limitation of content exclusions?

ANSWER: B

- B** Content exclusions can be worked around as it is only available for Git repositories.

Content exclusion can be configured by repository administrators, organization owners, and enterprise owners, so the claim that only an enterprise administrator can configure it is false. The documented limitation is that exclusions can effectively be worked around: Copilot may still use semantic information from excluded files provided indirectly by the IDE, and exclusions do not apply to symlinks.

Source: [GitHub Docs - Content exclusion for GitHub Copilot](#)

85 How can GitHub Copilot aid developers in writing documentation for their code?

ANSWER: C

- C** GitHub Copilot can suggest summaries or descriptions based on the code's functionality.

Copilot assists documentation by suggesting summaries, comments, and descriptions based on what the code does, which you then review. Claiming it automatically generates complete and detailed documentation overstates this, since output must be reviewed.

Source: [GitHub Docs - About GitHub Copilot](#)

86 What role does chat history play in GitHub Copilot's code suggestions?

ANSWER: B

- B** Chat history provides context to GitHub Copilot, improving the relevance and accuracy of its code suggestions.

The prompt engineering guidance states Copilot Chat uses chat history to get context about your request, which improves relevance. It does not train the model in real time, is not shared with other users for collaboration, and is far from irrelevant.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

87 When using GitHub Copilot to identify missing tests in your codebase, which of the following is the most important factor to consider?

ANSWER: B

- B** Ensuring that the correct context is available to GitHub Copilot.

Copilot's ability to find missing tests depends on having the right context available, meaning relevant open files, the code under test, and existing test patterns. High coverage or closing non-test tabs are not the key factor, and good practices help generally but context is what most directly drives accurate identification of gaps.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

88 What GitHub Copilot feature can be configured at the organization level to prevent GitHub Copilot suggesting publicly available code snippets?

ANSWER: C

- C** GitHub Copilot duplication detection filter

The duplication detection filter, the Suggestions matching public code policy, can be configured at the organization level to block suggestions that match publicly available code. Chat in the IDE or Mobile and web search access are unrelated to blocking public code matches.

Source: [GitHub Docs - GitHub Copilot code referencing](#)

89 What is a benefit of using custom models in GitHub Copilot?

ANSWER: B

- B** Responses use practices and patterns in your repositories

The benefit of a custom or fine-tuned model is that responses reflect the practices, patterns, and domain knowledge in your organization's repositories. Custom models do not guarantee correctness, do not inherently make responses faster, and do not run on a customer-provided LLM engine.

Source: [GitHub Docs - About GitHub Copilot](#)

90 When using GitHub Copilot Chat to generate unit tests, which slash command would you use?

ANSWER: D

D /tests

The Copilot Chat cheat sheet lists the tests slash command as the one that generates unit tests for the selected code. The other options are not real Copilot slash commands.

Source: [GitHub Docs - GitHub Copilot Chat cheat sheet](#)

91 How can you use GitHub Copilot to get inline suggestions for refactoring your code?

ANSWERS: A, E

A By adding comments to your code and triggering a suggestion.

E By highlighting the code you want to fix, right-clicking, and selecting "Refactor using GitHub Copilot."

You can trigger inline refactoring suggestions by describing the change in a comment and letting Copilot complete it, or by selecting code and using the Copilot context-menu refactor option. There is no gh copilot fix command, since the CLI supports suggest and explain, and the fix slash command targets fixing problems rather than refactoring.

Source: [GitHub Docs - GitHub Copilot Chat cheat sheet](#)

92 How does GitHub Copilot assist developers in minimizing context switching?

ANSWER: D

D GitHub Copilot allows developers to stay in their IDE.

Copilot reduces context switching by letting developers get suggestions, answers, and explanations without leaving their IDE. It does not replace human collaboration, guarantee bug prevention, or take over project management.

Source: [GitHub Docs - About GitHub Copilot](#)

93 When crafting prompts for GitHub Copilot, what is a recommended strategy to enhance the relevance of the generated code?

ANSWER: D

D Provide examples of expected input and output within the prompt.

Official prompt guidance explicitly recommends giving examples, including example input data and expected outputs, to help Copilot understand what you want, which is a form of few-shot prompting. Overly short prompts and omitting the programming language reduce relevance rather than improve it.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

94 Which two techniques does GitHub recommend for improving prompts to GitHub Copilot?

ANSWERS: A, D

- A** Be specific about your requirements and success criteria
- D** Provide examples of the input, output, or implementation you want

GitHub's prompt engineering guidance tells you to start general and then list specific requirements, and to give examples, including example input, example output, and example implementations. The same guidance says the opposite of the other two options: open relevant files and close irrelevant ones, and break complex tasks into simpler ones rather than combining them.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

95 An independent contractor develops applications for a variety of different customers. Assuming no concerns from their customers, which GitHub Copilot plan is best suited?

ANSWER: A

- A** GitHub Copilot Individual

A solo independent contractor with no customer data or intellectual property concerns is best served by the individual plan, which requires no organization or enterprise management. Business and Enterprise add centralized policy, seat management, and GitHub Enterprise tied features that a lone contractor does not need.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

96 How does GitHub Copilot Enterprise assist in code reviews during the pull request process?

ANSWERS: B, D

- B** It generates a prose summary and a bulleted list of key changes for pull requests.
- D** It can answer questions about the changeset of the pull request.

Copilot Enterprise helps with pull requests by generating a prose summary plus a bulleted list of key changes, and by letting you ask Copilot Chat questions about the changeset. It does not automatically merge pull requests and cannot validate or guarantee the accuracy of the changes, which remains the reviewer's responsibility.

Source: [GitHub Docs - About GitHub Copilot](#)

97 Why is it important to ensure the security of the code used in Generative AI (Gen AI) tools?

ANSWER: A

- A** Ensuring code security prevents unauthorized access and potential data breaches.

Securing code used with generative AI tools matters primarily because it prevents unauthorized access and potential data breaches, protecting sensitive source and secrets. The other options describe secondary or unrelated benefits rather than the core security rationale.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

98 What are the additional checks that need to pass before the GitHub Copilot responses are submitted to the user?

ANSWERS: A, C

- A** Code quality
- C** Suggestions matching public code (optional based on settings)

Before a suggestion is returned Copilot applies additional filters, including checks related to code quality and security such as filtering out insecure patterns and, when the policy is enabled, a duplication check against public code. Compatibility with user settings and performance benchmarking are not part of these response filters.

Source: [Microsoft Learn - Responsible AI with GitHub Copilot](#)

99 Are there any limitations to consider when using GitHub Copilot for code refactoring?

ANSWER: A

- A** GitHub Copilot may not always produce optimized or best-practice code for refactoring.

A real limitation is that Copilot may not always produce optimized or best-practice refactorings, so its output must be reviewed. It is not limited to a few languages, does not always produce bug-free code, and does not fully understand the context of your entire project.

Source: [GitHub Docs - Refactor code with GitHub Copilot](#)

100 Which Copilot Enterprise features are available in all commercially supported IDEs?

ANSWERS: A, D

- A** Inline suggestions
- D** Chat

Inline suggestions and Copilot Chat are available across the commercially supported IDEs including Visual Studio, VS Code, and JetBrains. Pull request summaries and knowledge bases are Copilot Enterprise features surfaced on GitHub.com, not delivered inside the IDEs.

Source: [GitHub Docs - About GitHub Copilot](#)

101 What content can be configured to be excluded with content exclusions?

ANSWERS: A, B, E

- A** Files
- B** Folders
- E** Repositories

Content exclusion lets you specify paths to individual files, folders and directories, and at the organization level you scope exclusions to specific repositories by referencing the repository followed by paths. Copilot cannot exclude at the granularity of individual lines within a file, and gists are not a supported exclusion target.

Source: [GitHub Docs - Excluding content from GitHub Copilot](#)

102 What is a key consideration when relying on GitHub Copilot Chat's explanations of code functionality and proposed improvements?

ANSWER: B

- B** Reviewing and validating the generated output for accuracy and completeness.

GitHub's guidance stresses that Copilot Chat explanations and suggested improvements may not always be accurate or complete and should be reviewed and validated by the developer. Copilot uses live model generation rather than a static database, does not primarily rely on user-provided documentation, and does not dynamically self-update from feedback.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

103 What GitHub Copilot pricing plan gives you access to your company's knowledge bases?

ANSWER: C

- C** GitHub Copilot Enterprise

Access to an organization's curated knowledge bases, now surfaced as Copilot Spaces, is a Copilot Enterprise capability layered on top of Copilot Business. Copilot Business provides policy and management controls but not knowledge bases. Note the option labels are dated, since Individual was renamed Pro and Professional is not a real plan.

Source: [GitHub Docs - Plans for GitHub Copilot](#)

104 Which of the following is a risk associated with using AI?

ANSWER: B

- B** AI systems can sometimes make decisions that are difficult to interpret.

A well documented risk of AI is that model decisions can be opaque and hard to interpret, which is why transparency and human review are emphasized. The other options are false claims: AI can perpetuate bias rather than being incapable of it, it does not remove the need for privacy regulation, and it does not replace developers.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

105 What practices enhance the quality of suggestions provided by GitHub Copilot?

ANSWERS: A, C, D

- A** Clearly defining the problem or task
- C** Using meaningful variable names
- D** Providing examples of desired output

GitHub's prompt engineering guidance recommends clearly describing the goal or task, using descriptive names for variables and functions, and giving examples of desired input and output. Including personal information in comments is a security and privacy risk rather than a quality booster, and a .gitignore file does not shape suggestion quality.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

106 How does GitHub Copilot Chat help in understanding the existing codebase?

ANSWER: C

- C** By answering questions about the code and generating explanations.

Copilot Chat helps you understand an existing codebase mainly by answering questions about the code and generating natural language explanations of functions, inputs, outputs, and dependencies. It is not fundamentally a linter, a diagram generator, or an automatic refactoring engine.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

107 What is the correct way to exclude specific files from being used by GitHub Copilot Business during code suggestions?

ANSWER: C

- C** Use the GitHub Copilot settings in the user interface to exclude files.

Content exclusion for Copilot Business is configured through repository or organization settings in the GitHub user interface under Copilot content exclusion. A .gitignore file only affects Git tracking and does not stop Copilot reading a file, there is no copilot.ignore file, and renaming files with a suffix has no effect.

Source: [GitHub Docs - Excluding content from GitHub Copilot](#)

108 In what ways can GitHub Copilot support a developer during the code refactoring process?

ANSWERS: A, C

- A** By offering code transformation examples that enhance performance and reduce complexity.
- C** By providing suggestions for improving code readability and maintainability based on best practices.

During refactoring Copilot can offer code transformation examples aimed at better performance and reduced complexity, and can suggest improvements to readability and maintainability based on best practices. It does not autonomously refactor entire codebases or independently guarantee regulatory compliance.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

109 Identify the right use cases where GitHub Copilot Chat is most effective.

ANSWERS: B, C

- B** Explain a legacy COBOL code and translate the code to another language like Python.
- C** Creation of a unit test scenario for newly developed Python code

Copilot Chat is strong at explaining unfamiliar or legacy code and translating it to another language, and at generating unit test scenarios for newly written code. Authoring full technical requirement specifications from business documents and designing end-to-end performance test scenarios are broader tasks where Copilot needs substantial human direction.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

110 What is the process behind identifying public code matches when using a public code filter enabled in GitHub Copilot?

ANSWER: A

- A** Running code suggestions through filters designed to detect public code

With the duplication filter enabled, Copilot runs its generated suggestions through a filter that detects matches of roughly 150 characters against publicly available code on GitHub and then blocks or annotates them. The close alternative is wrong because the match check is a direct comparison against indexed public code rather than a machine learning process, and browsing history plays no part.

Source: [GitHub Docs - GitHub Copilot code referencing](#)

111 What are the different ways to give context to GitHub Copilot to get more precise responses?

ANSWERS: B, D

- B** Engage with chat participants such as @workspace to incorporate collaborative context into the responses.
- D** Utilize chat variables like *file to anchor the conversation within the specific context of the files or editors in use.

You can steer Copilot with chat participants such as workspace or project scopes and with chat variables that anchor the conversation to specific files or editors. Copilot cannot read a developer's intentions with no input and does not access private prior projects without permission.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)

112 How can GitHub Copilot be limited when it comes to suggesting unit tests?

ANSWER: B

- B** GitHub Copilot primarily suggests basic unit tests that focus on core functionalities, often requiring additional input from developers for comprehensive coverage.

GitHub notes that generated test cases may not cover all possible scenarios and that manual testing and review remain necessary, so Copilot mainly produces baseline tests for core functionality and needs developer input for comprehensive coverage. Options claiming it handles any complexity overstate its abilities.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

113 What caution should developers exercise when using GitHub Copilot for assistance with mathematical computations?

ANSWER: C

- C** GitHub Copilot's reliance on pattern-based responses without verifying computation accuracy.

Copilot is a pattern-based language model that predicts plausible text and code but does not actually execute or verify computations, so mathematical results can be wrong and must be checked. Options claiming real-time execution, verification, or guaranteed optimization are misleading.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

114 What are the potential limitations of GitHub Copilot in maintaining existing codebases?

ANSWER: B

- B** GitHub Copilot might not fully understand the context and dependencies within a large codebase.

A key limitation is that Copilot may not fully grasp the context and dependencies across a large codebase, because it works from limited surrounding context rather than global awareness. It cannot reliably resolve all merge conflicts, its suggestions are not always aware of the whole codebase, and it does not safely refactor an entire codebase up to a fixed line count.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

115 What are the potential limitations of GitHub Copilot Chat?

ANSWERS: A, C

- A** Limited training data
C Ability to handle complex code structures

Real limitations include being bounded by its training data, which affects coverage and can be dated, and having limited ability to reliably handle very complex code structures. The other options are not limitations but false positive claims, since Copilot can reflect biases and its language support varies rather than being extensive for every language.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

116 In what ways can GitHub Copilot contribute to the design phase of the Software Development Life Cycle (SDLC)?

ANSWER: B

- B** GitHub Copilot can suggest design patterns and best practices relevant to the project.

In the design phase Copilot can suggest relevant design patterns and best practices to inform the architecture, acting as an assistant to human designers. It cannot independently produce a complete software design, manage team collaboration and version control, or generate UI prototypes without prompting.

Source: [GitHub Docs - Responsible use of GitHub Copilot Chat](#)

117 How can you improve the context used by GitHub Copilot?

ANSWERS: A, B

- A** By opening the relevant tabs in your IDE
- B** By adding relevant code snippets to your prompt

GitHub's guidance says to open relevant files and tabs so Copilot uses them as context, and to include relevant code snippets directly in your prompt. Adding files to a .gitconfig has nothing to do with Copilot context, and pasting full file paths into a prompt is not the recommended way to supply context.

Source: [GitHub Docs - Prompt engineering for GitHub Copilot Chat](#)