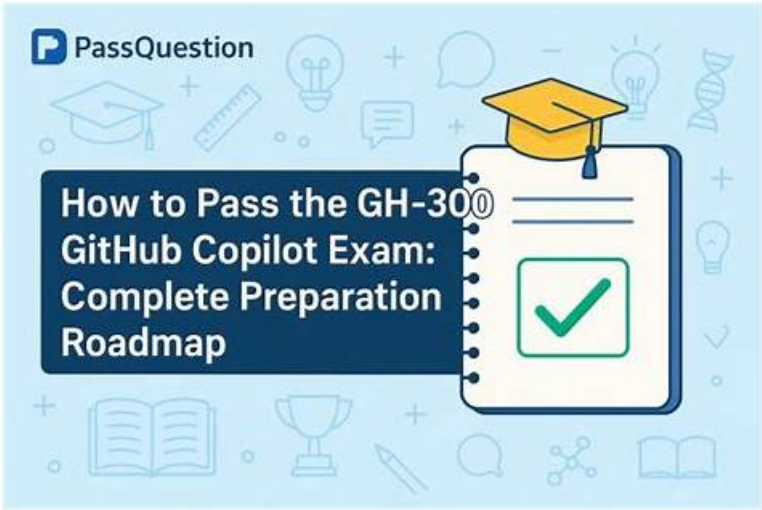


唯一無二GH-300受験練習参考書 |素晴らしい合格率の GH-300 Exam |素敵なGH-300: GitHub Copilot



P.S.JPNTTestがGoogle Driveで共有している無料の2026 Microsoft GH-300ダンプ: <https://drive.google.com/open?id=1FqhdKZymab3A-XP2HFL5zB938APosYQ>

JPNTTestのMicrosoftのGH-300試験トレーニング資料を手に入れたら、我々は一年間の無料更新サービスを提供します。それはあなたがいつでも最新の試験資料を持てるということです。試験の目標が変わる限り、あるいは我々の勉強資料が変わる限り、すぐに更新して差し上げます。あなたのニーズをよく知っていますから、あなたに試験に合格する自信を与えます。

人間はそれぞれ夢を持っています。適当な方法を採用する限り、夢を現実にすることができます。JPNTTestのMicrosoftのGH-300試験トレーニング資料を利用したら、MicrosoftのGH-300認定試験に合格することができますようになります。どうしてですかと質問したら、JPNTTestのMicrosoftのGH-300試験トレーニング資料はIT認証に対する最高のトレーニング資料ですから。その資料は最完全かつ最新で、合格率が非常に高いということで人々に知られています。それを持っていたら、あなたは時間とエネルギーを節約することができます。JPNTTestを利用したら、あなたは楽に試験に受かることができます。

>> GH-300受験練習参考書 <<

GH-300試験の準備方法 | ユニークなGH-300受験練習参考書試験 | 実用的なGitHub Copilotテスト問題集

あなたに相応しいJPNTTest問題集を探していますか。GH-300試験備考資料の整理を悩んでいますか。専門化のIT認定試験資料提供者JPNTTestとして、かねてより全面的の資料を準備します。あなたの資料を探す時間を節約し、Microsoft GH-300試験の復習をやっています。

Microsoft GH-300 認定試験の出題範囲:

トピック	出題範囲
トピック 1	• Privacy Fundamentals and Context Exclusions: This domain focuses on Security Engineers and Compliance Officers and addresses improving code quality with Copilot’s test suggestions and security optimizations. It covers identification of security vulnerabilities, performance enhancements, and privacy features like content exclusions at repository and organization levels with explanation of their limitations. Candidates learn about safeguarding mechanisms such as duplication detection, contractual protections, security checks, and troubleshooting guide for common Copilot issues including context exclusions and suggestion gaps.

トピック 2	• GitHub Copilot Plans and Feature: This domain targets Product Managers and DevOps Engineers and focuses on understanding the various GitHub Copilot subscription plans like Individual, Business, and Enterprise, including distinctions and management features. It covers how Copilot is integrated into IDEs, different triggering methods for code suggestions, organizational policy management, subscription administration via API, and effective use of Copilot Chat and Knowledge Bases. Candidates also learn about CLI usage and configuration.
トピック 3	• How GitHub Copilot Works and Handles Data: Designed for Machine Learning Engineers and Data Privacy Specialists, this section covers the data lifecycle and processing behind Copilot's code suggestions. It explains how context is gathered, prompts constructed, responses generated, and post-processed through proxy services. Candidates understand Copilot's data policies, handling of inputs, and limitations such as context window size and data age influencing suggestion relevance.
トピック 4	• Developer Use Cases for AI: Targeting Software Engineers and Technical Leads, this domain elaborates on how AI improves developer productivity across common tasks like learning new languages, translation, documentation, debugging, data science, and refactoring. It discusses Copilot's support in software development lifecycle management and highlights its limitations. Use of the productivity API to track Copilot's impact is also included.
トピック 5	• Responsible :This section of the exam measures skills of AI Ethics Officers and Risk Managers and covers the responsible and ethical usage of AI technologies. It explains the risks and limitations associated with generative AI tools, including biases in training data and the need to validate AI outputs. Candidates learn how to operate AI responsibly by identifying potential harms such as bias, fairness, privacy concerns, and mitigating these harms by applying ethical AI principles.
トピック 6	• Domain 6: Testing with GitHub Copilot: This section measures abilities of QA Engineers and Test Automation Specialists to use Copilot for test generation, including unit and integration tests. It explains how Copilot can identify edge cases and assist in writing assertions. The domain also covers different Copilot subscription SKUs, privacy considerations, organizational code suggestion settings, and configuration files related to Copilot.

Microsoft GitHub Copilot 認定 GH-300 試験問題 (Q51-Q56):

質問 # 51

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

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

正解: D

解説:

GitHub Copilot Chat is designed to be a productivity enhancer, not a replacement for human developers. It provides suggestions and assists with coding tasks, but the final decision and validation always rest with the developer. Copilot Chat is meant to augment the developer's workflow, making it faster and more efficient, but it does not remove the need for human oversight and judgment.

Reference: GitHub Copilot official documentation on the tool's purpose and usage.

質問 # 52

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

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

正解: C

解説:

In the CLI, GitHub Copilot generates code suggestions by analyzing code comments. You write comments describing what you want, and Copilot provides relevant code suggestions. You then select the best suggestion from the list.

Reference: GitHub Copilot CLI documentation.

質問 # 53

Identify the right use cases where GitHub Copilot Chat is most effective. (Each correct answer presents part of the solution. Choose two.)

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

正解: A、D

解説:

GitHub Copilot Chat is effective for explaining and translating legacy code and generating unit test scenarios for new code.

質問 # 54

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. Design patterns
- C. Compiled binaries
- D. Screenshots
- E. Documentation

正解: A、B、E

解説:

"GitHub Copilot Knowledge Bases allow you to ask questions about your organization's documentation, design patterns, and code snippets." This proves that Copilot Knowledge Base can work with code snippets, documentation, and design patterns, but not binaries or screenshots.

References: GitHub Copilot Enterprise Knowledge Base documentation.

質問 # 55

How does the /tests slash command assist developers?

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

正解: B

解説:

The /tests slash command in GitHub Copilot Chat creates unit tests for the selected code, helping developers ensure the functionality and reliability of their code.

質問 # 56

.....

MicrosoftのGH-300認定試験を受けることを決めたら、JPNTestがそばにいて差し上げますよ。JPNTestはあなたが自分の目標を達成することにヘルプを差し上げられます。あなたがMicrosoftのGH-300「GitHub Copilot」認定試

- ちなみに、JPNTest GH-300の一部をクラウドストレージからダウンロードできます: <https://drive.google.com/open?id=1FqhdKZymuab3A-XP2HFL5zB938APosYQ>