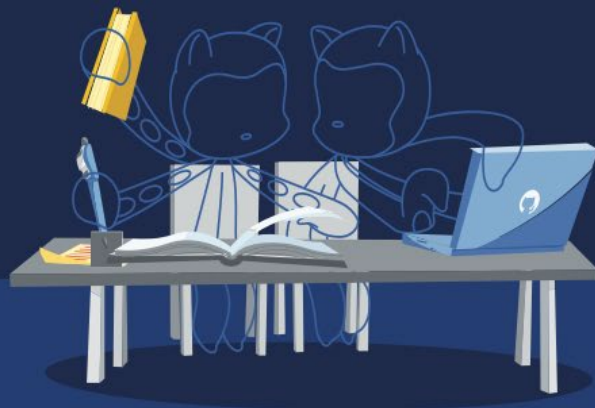


学習ガイド

GitHub Copilot



包括的な学習ガイドを使用して、GitHub Copilot 認定試験に備えましょう。試験を有利に進め、合格の可能性を高めるために不可欠なリソースと学習アクティビティを厳選しました。

対象読者のプロフィール

この試験は、ソフトウェア 開発者、管理者、プロジェクト マネージャーなど、GitHub の使用に熟練しているソフトウェア開発分野の個人向けに設計されています。この認定試験は、GitHub Copilot を使用したソフトウェア開発の最適化での実務経験に加え、製品およびその利用可能な機能としての GitHub Copilot についての基礎的な理解がある個人を対象としています。

試験範囲

認定試験の試験範囲は、「ドメイン」または「試験ドメイン」と呼ばれることが多く、認定試験で出題される具体的な知識、スキル、トピックを定義する構造化された概要またはフレームワークです。試験の出題傾向や、学習と準備が必要な部分についての明確なロードマップを受験者に提供します。

この学習ガイドで提供されるドメインは、各ドメイン内での学習目標に加え、GitHub Copilot の試験で扱われるトピックのカテゴリーを把握してもらうことを目的としています。

ドメインの内訳	ドメイン別の割合
ドメイン 1: 責任あるAI	7%
ドメイン 2: GitHub Copilot のプランおよび特徴	31%
ドメイン 3: GitHub Copilot の機能およびデータ処理の仕組み	15%
ドメイン 4: プロンプト クラフティングおよびプロンプト エンジニアリング	9%
ドメイン 5: 開発者による AI のユースケース	14%
ドメイン 6: GitHub Copilot でのテスト	9%
ドメイン 7: プライバシーの基礎およびコンテキスト除外	15%

合格のための推奨事項とベストプラクティス

GitHub Copilot の試験に合格する機会を増やすには、GitHub Copilot の機能を使用した実務経験および GitHub についての基礎的な理解が必要です。この認定試験の推奨されるラーニング パスでは、学習コンテンツを徹底的に学習した後に、実践的な演習やコンテンツに取り組み、知識を磨き、試験に備えることができます。

コンテンツ リソース

次のリソースは、GitHub Copilot の試験における各ドメインの学習目標をカバーする推奨コンテンツとして、GitHub と共同で作成されました。Microsoft Learn と LinkedIn ラーニングは、どちらも認定試験のための完全なラーニング パスを提供しますが、学習の進め方は異なります。

Microsoft Learn



Microsoft Learn (MS Learn) と共同で、モジュールの包括的なコレクションが提供される 2 つのラーニング パスを作成しました。このモジュールは、GitHub Copilot を効果的に使用する準備のために設計されています。開発ライフサイクルのあらゆる段階で AI による支援を使用してコーディング経験を向上させる方法について学習します。

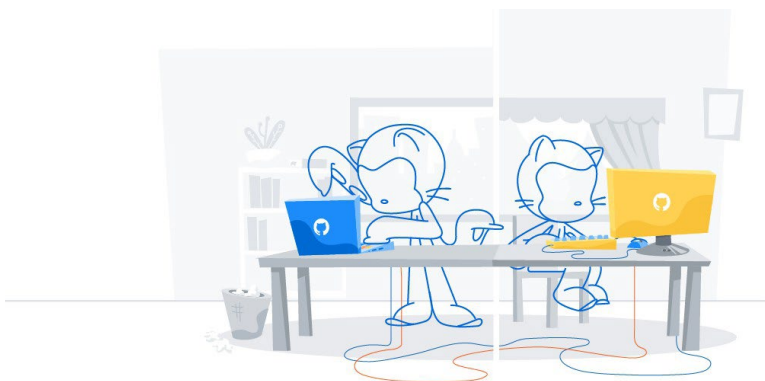
GitHub Copilot は、少ないエラーでコードをより迅速に記述するのに役立つ GitHub の統合ツールです。これを使用すると、コード スニペットの提案、行の完了、関数全体の生成を行うこともできます。以下のモジュールでは、GitHub Copilot の機能について紹介します。これにより、自身の GitHub 環境でこれらの機能を認識、適用、評価するのに必要なスキルを身に付けることができます。

- [GitHub Copilot の基礎 - AI ペア プログラマーの理解](#)
- [GitHub Copilot を使用してアプリケーション開発を加速させる](#)

LinkedIn ラーニング



LinkedIn ラーニングのラーニング パス「GitHub Copilot 認定試験に向けた準備」を開始して、ソフトウェア開発プロセスにおける AI による支援を使用したコーディング効率および品質技術についてマスターしましょう。GitHub Copilot では、コードをより迅速に記述し、改善を提案し、関数全体を生成するのに役立つ強力な AI 駆動のツールを利用できます。この動画ベースのラーニング パスは、GitHub Copilot の機能について案内するさまざまな魅力的なコースを備えています。このラーニング パスを終えるまでには、コーディング環境で GitHub Copilot の機能をシームレスに適用、評価、最大化するための知識と技術を身につけることができます。**(学習パスはまもなく公開されます)**



ドメイン 1: 責任あるAI

AI の責任ある使用について説明する
AI の使用に関するリスクについて説明する
生成 AI の使用制限について説明する (モデルにおけるソース データ部門、データのバイアスなど)
AI ツールの出力を検証する必要性について説明する
責任ある AI を実施する方法を特定する
生成 AI における潜在的な有害性を特定する (バイアス、セキュア コード、公平性、プライバシー、透明性)
潜在的な有害性が発生した場合の軽減方法について説明する
AI の倫理について説明する

ドメイン 2: GitHub Copilot のプランおよび特徴

GitHub Copilot のさまざまなプランを識別する
Copilot Individual、Copilot Business、Copilot Enterprise、および Copilot Business for non-GHE の違いについて理解する
GitHub の顧客以外のための Copilot について理解する
統合開発環境での GitHub Copilot について定義する
統合開発環境での GitHub Copilot Chat について定義する
GitHub Copilot を開始するためのさまざまな方法について説明する (チャット、インライン チャット、提案、例外処理、CLI)

GitHub Copilot Individual の主な機能について確認する
GitHub Copilot Individual および GitHub Copilot Business の違いについて説明する (データの除外、知的財産の損失補償、請求など)
GitHub Copilot Individual 用の統合開発環境で利用可能な機能について理解する

GitHub Copilot Business の主な機能について確認する

GitHub Copilot から特定のファイルを除外する方法を示す

組織全体のポリシーマネジメントを確立する方法を示す

GitHub Copilot Business の組織監査ログの目的を説明する

GitHub Copilot Business の監査ログ イベントを検索する方法について説明する

REST API を通して GitHub Copilot Business のサブスクリプションを管理する方法について説明する

GitHub Copilot Enterprise の主な機能について確認する

[GitHub.com](https://github.com) で GitHub Copilot Chat を使用する利点について説明する

GitHub Copilot のプルリクエストの概要について説明する

GitHub Copilot Enterprise 内のナレッジ ベースの構成および使用方法について説明する

ナレッジ ベースに保存可能なさまざまな種類のナレッジについて説明する (コード スニペット、ベストプラクティス、設計パターンなど)

コード補完およびレビューのためにナレッジ ベースを使用する利点について説明する (コード品質の向上、一貫性、効率など)

インデックス作成やその他の関連する構成手順を含む、GitHub Copilot Enterprise 内のナレッジ ベースを作成、管理、検索する方法について説明する

カスタム モデルを使用する利点について説明する

GitHub Copilot Chat の主な機能について確認する

GitHub Copilot Chat が最も効果的なユース ケースについて確認する

GitHub Copilot Chat 向けにパフォーマンスを改善する方法について説明する

GitHub Copilot Chat の使用制限について確認する

GitHub Copilot Chat からのコード提案を使用する際に利用可能なオプションについて確認する

GitHub Copilot Chat についてのフィードバックを共有する方法について説明する

GitHub Copilot Chat を使用する際の一般的なベストプラクティスについて確認する

GitHub Copilot Chat を使用する際に利用可能なスラッシュコマンドについて確認する

CLI で GitHub Copilot を使用する

CLI で GitHub Copilot をインストールする手順について検討する

CLI で GitHub Copilot を使用する際の一般的なコマンドについて確認する

CLI で GitHub Copilot 内に構成できる複数の設定について確認する

ドメイン 3: GitHub Copilot の機能およびデータ処理の仕組み**GitHub Copilot がデータを処理する仕組みについて説明する**

GitHub Copilot Individual でのデータの使用および共有方法について説明する

GitHub Copilot のコード補完用のデータ フローについて説明する

GitHub Copilot Chat 用のデータ フローについて説明する

GitHub Copilot Chat 用のさまざまな種類の入力処理 (設計されたプロンプトの種類) について説明する

統合開発環境での GitHub Copilot コード提案におけるデータ パイプラインのライフサイクルについて説明する

GitHub Copilot コード提案のライフサイクルを可視化する

GitHub Copilot でのコンテキストの収集方法について説明する

GitHub Copilot でのプロンプトの構築方法について説明する

プロキシ サービスと各プロンプトが通過するフィルターについて説明する

大規模言語モデルで応答が生成される仕組みについて説明する

プロキシ サーバーを使用した GitHub Copilot の応答のポスト処理について説明する

GitHub Copilot で一致するコードを特定する仕組みについて確認する

GitHub Copilot (および一般的な LLM) での制限について説明する

ソース データで最もよく見られている例の効果について説明する

コード提案のデータがどのくらい前のもので、どのくらい関連があるのかについて説明する

プロンプトと計算から推論とコンテキストを提供する GitHub Copilot の性質について説明する

限定されたコンテキスト ウィンドウについて説明する

ドメイン 4: プロンプト クラフティングおよびプロンプト エンジニアリング

プロンプト クラフティングの基礎について説明する

プロンプトのコンテキストが決定される仕組みについて説明する

GitHub Copilot のプロンプトを行うための言語オプションについて説明する

プロンプトの異なる部分について説明する

プロンプトの役割について説明する

Zero-shot プロンプティングおよび Few-Shot プロンプティングの違いについて説明する

GitHub Copilot でチャット履歴が使用される方法について説明する

GitHub Copilot を使用する際のプロンプト クラフティングのベストプラクティスについて確認する

プロンプト エンジニアリングの基礎について説明する

エンジニアリング プリンシパル、トレーニング方法、ベストプラクティスについて説明する

プロンプトの処理フローについて説明する

ドメイン 5: 開発者による AI のユースケース

開発生産性の向上

開発生産性の向上に向けた AI による一般的なユース ケースの改善方法について説明する

- 新しいプログラミング言語およびフレームワークについて学習する
- 言語翻訳
- コンテキスト切り替え
- ドキュメントの作成
- パーソナライズされたコンテストアウェアな応答
- サンプルデータの作成
- レガシー アプリケーションの最新化
- コードのデバッグ
- データサイエンス
- コードのリファクタリング

SDLC (ソフトウェア開発ライフ サイクル) の管理に GitHub Copilot がどのように役立つのかについて説明する

GitHub Copilot での制限について説明する

GitHub Copilot のコーディングへの影響を確認するための生産性 API の使用方法について説明する

ドメイン 6: GitHub Copilot でのテスト

コード用のテストを生成するためのオプションについて説明する

GitHub Copilot がコードへのユニットテスト、インテグレーションテスト、その他の種類のテストにどのように使用されるかについて説明する

エッジケースを特定する際、およびそれらに対処するためテストを提案する際の GitHub Copilot による支援の仕組みについて確認する

テストによるコード品質の向上

GitHub Copilot の提案を使用して、既存のテストの効果を向上させる方法について説明する

GitHub Copilot を使用してさまざまな種類のテスト用に定型句コードを生成する方法について説明する

さまざまなテストのシナリオ用のアサーションを記述するのに GitHub Copilot がどのように役立つのかについて説明する

セキュリティおよびパフォーマンスのために GitHub Copilot を活用する

GitHub Copilot が既存のテストを学習し、改善を提案して、コード内の潜在的な問題を特定する仕組みについて確認する

共同のコードレビュー、セキュリティのベストプラクティスの活用、パフォーマンスに関する考察における GitHub Copilot Enterprise の使用方法について説明する

GitHub Copilot で、コード内の潜在的なセキュリティの脆弱性を特定できる仕組みについて確認する

パフォーマンスの改善のため、GitHub Copilot により提案コードの最適化が行われる仕組みについて確認する

ドメイン 7: プライバシーの基礎およびコンテキスト除外

GitHub Copilot 向けのさまざまな SKU について説明する

GitHub Copilot 向けのさまざまな SKU およびプライバシーに関する考察について説明する

組織レベルのさまざまなコード提案の構成オプションについて説明する

GitHub Copilot のエディタ構成ファイルについて説明する

コンテンツ除外について確認する

リポジトリおよび組織にコンテンツ除外を構成する方法について説明する

コンテンツ除外による効果について説明する

コンテンツ除外における制限について説明する

GitHub Copilot の入力の所有権について説明する

セーフガード

重複データ検出フィルターについて説明する

契約上の保護について説明する

[GitHub.com](#) で GitHub Copilot の設定を構成する方法について説明する

- 重複データ検出の有効化/無効化
- プロンプトおよび提案コレクションの有効化/無効化

セキュリティ チェックおよび警告について説明する

トラブルシューティング

複数のファイルのエディタでコード提案が表示されない場合の問題の解決方法について説明する

コンテキスト除外が適用されない場合がある理由を説明する

提案が存在しない、または適していない場合に GitHub Copilot を開始する方法について説明する

コード エディタでのコンテキスト除外の手順を説明する