

MS-GH900T00: GITHUB FOUNDATIONS



DURATION	LEVEL	TECHNOLOGY	DELIVERY METHOD	TRAINING CREDITS
2 Days	Beginner	Microsoft GitHub	Instructor-led	NA

INTRODUCTION

GitHub Foundations introduces you to the fundamental concepts, features, and products of GitHub. You'll discover the benefits of using GitHub as a collaborative platform and explore its core features, such as repository management, commits, branches, and merging. Through curated modules and hands-on exercises, you'll gain a solid understanding of GitHub's essential tools and be well-equipped to start contributing to projects and collaborating effectively within GitHub.

AUDIENCE PROFILE

This course is intended for students who want to understand and GitHub best practices. You will understand the fundamental features of GitHub, learn about repository management, gain an understanding of the GitHub flow, including branches, commits, and pull requests. Additionally, you will explore the collaborative features of GitHub by reviewing issues and discussions and be able to manage your GitHub notifications and subscriptions.

PREREQUISITES

Before attending this course, delegates must have:

- Basic computer skills - navigating files, using a browser, and general tech comfort.
- Understanding of software development concepts - you don't need to be a developer, but knowing what code is and how software is built will help.
- Experience with command-line tools - some exercises involve using the terminal or command prompt.
- Familiarity with version control concepts - like what it means to commit changes or use a repository.

COURSE OBJECTIVES

After attending this course, delegates will be able to:

- Administer GitHub organizations and manage user access and permissions.
- Understand Git and distributed version control to manage code changes effectively.
- Manage repositories using GitHub, including creating, cloning, and organizing them.
- Use GitHub Copilot for AI-powered coding assistance.
- Collaborate with GitHub Projects and Codespaces to streamline development workflows.
- Contribute to open-source projects and understand the GitHub flow (branches, commits, pull requests).
- Implement secure development practices using GitHub's built-in tools.

COURSE CONTENT

Module 1: Introduction to Git

Find out what source control is and get an introduction to Git—the world's most popular version control system.

Lessons

- Introduction
- What is version control?
- Exercise - Try out Git
- Basic Git commands
- Module assessment
- Summary

In this module, you'll:

- Learn what version control is
- Understand distributed version control systems, like Git
- Create a new Git project and configure it
- Make and track changes to code by using Git
- Use Git to recover from simple mistakes

Module 2: Introduction to GitHub

Learn to use key GitHub features, including issues, notifications,

branches, commits, and pull requests.

Lessons

- Introduction
- What is GitHub?
- Components of the GitHub flow
- GitHub is a collaborative platform
- GitHub platform management
- Exercise - A guided tour of GitHub
- Module assessment
- Summary

After completing this module, students will be able to:

- Identify the fundamental features of GitHub.
- Learn about repository management.
- Gain an understanding of the GitHub flow, which includes branches, commits, and pull requests.
- Explore the collaborative features of GitHub by reviewing issues and discussions.

- Recognize how to manage your GitHub notifications and subscriptions.

Module 3: Introduction to GitHub's products

This module provides an overview of GitHub products, including the account types, plan options, associated features, and billing. It also covers how to access GitHub on-the-go using GitHub Desktop and GitHub Mobile.

Lessons

- Introduction
- GitHub accounts and plans
- GitHub Mobile and GitHub Desktop
- GitHub billing
- License Usage Stats
- License Usage Stats in Machine and Peripheral Devices
- Metered Usage Reports
- Module assessment
- Summary

After completing this module, students will be able to:

- Define the difference between the different types of GitHub accounts: Personal, Organization, and Enterprise.
- Explain each GitHub plan: GitHub Free for personal accounts and organizations, GitHub Pro for personal accounts, GitHub Team, and GitHub Enterprise.
- Distinguish the features associated with accessing GitHub on GitHub Mobile and GitHub Desktop.
- Describe a brief overview of GitHub billing and payments.

Module 4: Configure code scanning on GitHub

This module introduces you to code scanning and its features. You'll learn how to implement code scanning using CodeQL, third party tools, and GitHub Actions.

Lessons

- Introduction
- What is code scanning?
- Enable code scanning with third party tools
- Configure code scanning
- Configure code scanning exercise
- Module assessment
- Summary

After completing this module, you'll be able to:

- Describe code scanning.
- List the steps for enabling code scanning in a repository.
- List the steps for enabling code scanning with third-party analysis.
- Contrast how to implement CodeQL analysis in a GitHub Actions workflow versus a third-party continuous integration (CI) tool.
- Explain how to configure code scanning on a repository using triggering events.
- Contrast the frequency of code scanning workflows (scheduled vs triggered by events).

Module 5: Introduction to GitHub Copilot

GitHub Copilot uses OpenAI Codex to suggest code and entire functions in real time, right from your editor.

Lessons

- Introduction
- GitHub Copilot, your AI pair programmer
- Interact with Copilot
- Set up, configure, and troubleshoot GitHub Copilot
- Exercise - Develop with AI-powered code suggestions by using GitHub Copilot and VS Code
- Module assessment
- Summary

In this module, you will:

- Learn how GitHub Copilot can help you code by offering autocomplete-style suggestions.
- Learn about the various ways to trigger GitHub Copilot.
- Learn about the differences among GitHub Copilot Individual, Business, and Enterprise.
- Learn how to configure GitHub Copilot.
- Troubleshoot GitHub Copilot.

Module 6: Code with GitHub Codespaces

GitHub Codespaces is a fully configured development environment hosted in the cloud. By using GitHub Codespaces, your workspace, along with all of your configured development environments, is available from any computer with access to the internet.

Lessons

- Introduction
- The Codespace lifecycle
- Personalize your Codespace
- Codespaces versus GitHub.dev editor
- Exercise - Code with Codespaces and Visual Studio Code
- Module assessment
- Summary

By the end of this module, you're able to:

- Describe GitHub Codespaces.
- Explain the GitHub Codespace lifecycle and how to perform each step.
- Define the different customizations you can personalize with GitHub Codespaces.
- Discern the differences between GitHub.dev and GitHub Codespaces.

Module 7: Manage your work with GitHub Projects

Learn to use GitHub Projects to create issues, break them into tasks, track relationships, add custom fields, and have conversations.

Lessons

- Introduction
- Projects versus Projects Classic
- How to create a project
- How to organize your project
- How to organize and automate your project
- Insight and automation with projects
- Module assessment
- Summary

In this module, you learn to:

- Discern the differences between Projects and Projects (Classic).

- Build an organization level Project.
- Organize your Project.
- Edit the visibility, access, and management of your Project.
- Develop automation and insights from your Project.

Module 8: Communicate effectively on GitHub using Markdown

Learn to use Markdown to communicate with brevity, clarity, and expression.

Lessons

- Introduction
- What is Markdown?
- Exercise - Communicate using Markdown
- Module assessment
- Summary

In this module, you'll:

- Use Markdown to add lists, images, and links in a comment or text file.
- Determine where and how to use Markdown in a GitHub repository.
- Learn about syntax extensions available in GitHub (GitHub-flavored Markdown).

Module 9: Contribute to an open-source project on GitHub

Learn how to use GitHub to find open-source projects and tasks to contribute to. Discover how to create pull requests and communicate with project maintainers effectively to get your changes accepted. Learn about the benefits of getting involved with open-source communities.

Lessons

- Introduction
- Identify where you can help
- Contribute to an open-source repository
- Exercise - Create your first pull request
- Next steps
- Module assessment
- Summary

In this module, you will learn how to:

- Find open-source projects and tasks to contribute to in GitHub.
- Create pull requests to open-source projects.
- Implement best practices to communicate with open-source maintainers and perform code reviews.
- Find and engage with open-source communities.

Module 10: Manage an InnerSource program by using GitHub

Learn to manage a successful InnerSource program on GitHub through effective discoverability, guidance, and maintenance.

Lessons

- Introduction
- How to manage a successful InnerSource program
- Exercise - InnerSource fundamentals
- Module assessment
- Summary

In this module, you learn how to:

- Contrast user- versus organization-owned projects.
- Make recommendations about the number of GitHub organizations you should have.
- Create discoverable repositories.
- Create robust repository READMEs.
- Use issue and pull-request templates.
- Build transparency into repositories.
- Measure the success of InnerSource within your organization.
- Distribute your InnerSource toolkit.

Module 11: Maintain a secure repository by using GitHub best practices

In this module, you'll learn best practices for building, hosting, and maintaining a secure repository on GitHub.

Lessons

- Introduction
- How to maintain a secure GitHub repository
- Automated security
- Exercise - Secure your repository's supply chain
- Module assessment
- Summary

In this module, you will:

- Identify the tools and GitHub features to establish a secure development strategy.
- Enable vulnerable dependency detection for private repositories.
- Detect and fix outdated dependencies with security vulnerabilities.
- Automate the detection of vulnerable dependencies with Dependabot.
- Add a security policy with a SECURITY.md file.
- Remove a commit exposing sensitive data in a pull request.
- Remove historical commits exposing sensitive data deep in your repository.

Module 12: Introduction to GitHub administration

GitHub Copilot is an AI pair programmer that offers autocomplete-style suggestions as you code in Python.

Lessons

Understand the security and control measures available to GitHub administrators within an organization or enterprise.

Lessons

- Introduction
- What is GitHub administration?
- How does GitHub authentication work?
- How does GitHub organization and permissions work?
- Managing enterprise access, permissions, and governance
- Module assessment
- Summary

After completing this module, students will be able to:

- Summarize the organizational structures and permission levels that GitHub administrators can use to organize members in order to control access and security.
- Identify the various technologies that enable a secure authentication strategy allowing administrators to centrally manage repository access.
- Describe the technologies required to centrally manage teams and members using existing directory information services.
- Describe how you can use GitHub itself as an identity provider for authentication and authorization.

Module 13: Authenticate and authorize user identities on GitHub

This module provides an overview of the authentication and authorization options available to you in your GitHub organization or GitHub Enterprise.

Lessons

- Introduction
- User identity and access management
- User authentication
- User authorization
- Team synchronization
- Module assessment
- Summary

By the end of this module, you will:

- Be able to describe the Authentication and Authorization Model.
- Understand how to manage user access to your GitHub organization through Authorization and Authentication tools.
- Identify the supported identity providers and technologies
- Introduction
- What is GitHub Copilot?
- Exercise - Set up GitHub Copilot to work with Visual Studio Code

that support secure repository access.

- Understand the implications of enabling SAML SSO.
- Identify the authorization and authentication options available and understand the administrator's role in enforcement of a secure access strategy for a GitHub enterprise.
- Describe how users access private information in a GitHub organization.
- Evaluate the benefits of enabling Team Synchronization to manage team membership.

Module 14: Manage repository changes by using pull requests on GitHub

Learn how to manage changes to your repository source by using pull requests.

Lessons

- Introduction
- What are pull requests?
- Exercise - Reviewing pull requests
- Module assessment
- Summary

In this module, you will:

- Review branches and their importance to pull requests.
- Define what a pull request is.
- Learn how to create a pull request.
- Understand the different pull request statuses.
- Walk through how to merge a pull request to a base branch.

Module 15: Search and organize repository history by using GitHub

Learn to search and organize repository history by using filters, blame, and cross-linking on GitHub.

Lessons

- Introduction
- How to search and organize repository history by using GitHub
- Exercise - Connect the dots in a GitHub repository
- Module assessment
- Summary

In this module, you will:

- Find relevant issues and pull requests.
- Search history to find context.
- Make connections within GitHub to help others find things.

Module 16: Using GitHub Copilot with Python

- Use GitHub Copilot with Python
- Exercise - Update a Python web API with GitHub Copilot
- Module assessment

- Summary
- By the end of this module, you're able to:
- Enable the GitHub Copilot extension in Visual Studio Code.
 - Craft prompts that can generate useful suggestions from GitHub Copilot.
 - Use GitHub Copilot to improve a Python project.

ASSOCIATED CERTIFICATIONS & EXAM

This course will prepare delegates to write the GitHub Foundations exam.