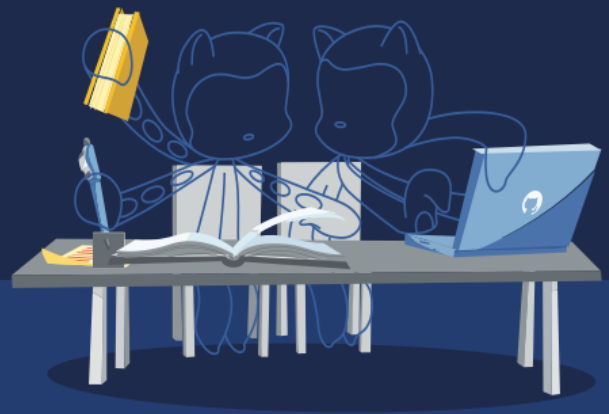# GitHub

Study Guide
GitHub Foundations

Get exam-ready for your GitHub Foundations Certification with our comprehensive study guide. We've curated the essential resources and insights you need to navigate the foundations of GitHub and boost your success with the exam.

Audience Profile

This exam is targeted towards GitHub users who want to validate their understanding of the foundational topics, products, and concepts of collaborating, contributing, and working on GitHub.

Objective Domains

An objective domain for a certification exam, often referred to as a "domain" or "exam domain," is a structured outline or framework that defines the specific knowledge, skills, and topics that the certification exam will cover. It provides a clear roadmap for what candidates should expect to encounter on the exam and what they need to study and prepare for.

The domains provided in this study guide are intended to provide insight into the topic categories covered in the GitHub Foundations exam, along with the learning objective within each domain.

Domain Breakdown
Domain 1: Introduction to Git and GitHub
Domain 2: Working with GitHub Repositories
Domain 3: Collaboration Features
Domain 4: Modern Development
Domain 5: Project Management
Domain 6: Privacy, Security, and Administration
Domain 7: Benefits of the GitHub Community

Recommendations and Best Practices for Success

To increase your chances of success in passing the GitHub Foundations exam, it's essential to start with a solid foundation of basic experience, exposure, and proficiency on GitHub. The recommended learning paths for this exam provide you with an in-depth study of the learning content, followed by hands-on exercises and preparation assessment questions that were created to enable you to fine-tune your knowledge and readiness for the certification exam.

Content Resources

The following resources have been created in collaboration with GitHub as recommended content that covers the learning objectives in each domain for the GitHub Foundations exam. Both Microsoft Learn and LinkedIn Learning provide a complete learning path for the exam, but offer a different learning experience.

Microsoft Learn

The GitHub Foundations learning path on MS Learn offers a streamlined and novice-friendly pathway designed to introduce you to the fundamental concepts and products of GitHub tailored for this exam. This learning journey will uncover the advantages of leveraging GitHub as a collaborative platform and delve into its fundamental functionalities and core concepts. By engaging with thoughtfully curated modules and hands-on exercises, you will establish a foundational comprehension of GitHub's indispensable tools, preparing you to better contribute to projects and foster efficient collaboration within the GitHub environment.

LinkedIn Learning

The Prepare for the GitHub Foundations Certification learning path on LinkedIn Learning is a comprehensive set of courses designed to equip you with the skills and knowledge necessary to excel in the GitHub Foundations exam. Through engaging video lessons, follow-along examples, and clear instructions, you'll gain a solid understanding of essential concepts such as version control, collaboration best practices, and repository navigation. The courses in this learning path are structured to not only help you master these fundamental GitHub skills, but also provide practical insights and real-world scenarios to reinforce your learning.

Domain 1: Introduction to Git and GitHub

Git and GitHub Basics
Describe version control
Define distributed version control
Describe Git
Describe GitHub
Explain the difference between Git and GitHub
Describe a GitHub repository
Describe a commit
Describe branching
Define a remote in Git terminology
Describe the GitHub flow

GitHub Entities
Describe the different GitHub accounts (personal, organization, enterprise)
Describe GitHub's products for personal accounts (free, pro)
Describe GitHub's products for organization accounts (free for organizations, teams)
Describe the different deployment options for GitHub Enterprise
Describe the features in the user profile (metadata, achievements, profile readme, repositories, pinned repositories, stars, etc.)

GitHub Markdown
Identify the text formatting toolbar on issue and pull request comments
Describe Markdown
Identify the basic formatting syntax (headings, links, task lists, comments, etc.)
Explain where to find and use slash commands

GitHub Desktop
Explain the difference between GitHub Desktop and github.com
Describe the available features with GitHub Desktop

GitHub Mobile
Describe the available features with GitHub Mobile
Explain how to manage notifications through the GitHub Mobile app

Domain 2: Working with GitHub Repositories

Understanding GitHub Repositories
Describe the components of a good README and the recommended repository files (LICENSE, CONTRIBUTING, CODEOWNERS)
Explain basic repository navigation
Explain how to create a new repository
Describe repository templates
Describe the different features to maintaining a repository
Describe how to clone a repository
Describe how to create a new branch
Explain how to add files to a repository
Identify how to view repository insights
Explain how to save a repository with stars
Explain feature previews

Domain 3: Collaboration Features

Issues
Describe how to link a PR to an issue
Describe how to create an issue
Describe the difference between an issue, discussion, and pull request
Explain how to create a branch from an issue
Identify how to assign issues
Describe how to search and filter issues
Describe how to pin an issue
Explain basic issue management
Explain the difference between issue templates and issue forms
Explain how to use keywords in issues

Pull requests

Describe a pull request

Explain how to create a new pull request

Describe the `base` and `compare` branches in a pull request

Explain the relationship of commits on a pull request

Describe draft pull requests

Describe the purpose of the pull request tabs (conversation, commits, checks, files changed)

Identify how to link activity within a pull request

Explain the different pull request statuses

Recognize how to comment on a posted link to a line or lines of code from a file

Describe code review with a codeowners file

Explain the different options for providing a code review on a pull request (comment, approve, request changes, suggested changes)

Discussions

Describe the difference between discussions and issues

Explain the options available with discussions (announcements, ideas, polls, Q&A, show and tell)

Identify how to mark a comment as an answer to a discussion

Explain how to convert a discussion to an issue

Recognize how to pin a discussion

Notifications

Describe how to manage notification subscriptions

Explain how to subscribe to notification threads

Describe how to find threads where you are at-mentioned

Identify the notification filtering options

Explain the different notification configuration options

Gists, Wikis, and GitHub Pages
Explain how to create a GitHub gist
Describe how to fork and clone a gist
Explain GitHub Wiki pages
Describe how to create, edit, and delete wiki pages
Explain the visibility of wiki pages
Describe GitHub Pages

Domain 4: Modern Development

GitHub Actions
Describe GitHub Actions (basic understanding)
Explain where you can use GitHub Actions within GitHub (general event types)
Explain where you can find existing GitHub Actions

GitHub Copilot
Describe GitHub Copilot
Describe the difference between GitHub Copilot for Individuals and GitHub Copilot for Business
Explain how to get started using GitHub Copilot

GitHub Codespaces
Describe GitHub Codespaces
Identify how do to start a GitHub codespace
Describe the codespace lifecycle
Describe the different customizations you can personalize with GitHub Codespaces
Recognize how to add and configure dev containers
Identify how to share a deep link to a GitHub codespace
Explain how to use the github.dev editor
Explain the differences between the github.dev editor and a GitHub Codespace

Domain 5: Project Management

Manage your work with GitHub Projects
Describe GitHub Projects
Explain the layout options for projects
Describe the configuration options for projects
Explain the difference between projects and projects classic
Explain the use of labels
Explain the use of milestones
Describe how to use and create template repos
Explain how to create, edit, and delete saved replies
Describe the benefits of using a saved reply
Recognize how to add assignees to issues and pull requests
Explain how to use project workflows
Describe project insights

Domain 6: Privacy, Security, and Administration

Authentication and Security
Explain how to secure your account with 2FA
Describe the different access permissions
Explain EMUs (Enterprise Managed Users)

GitHub Administration
Explain how to enable and disable features
Recognize repository permission levels
Identify the options for repository visibility
Explain repository privacy setting options (branch protections, codeowners, required reviewers)
Describe the main features and options in the Security tab
Define repository insights
Explain how to manage collaborators

Explain how to manage organization settings
Describe members, teams, and roles in a GitHub organization

Domain 7: Benefits of the GitHub Community

Describe the benefits of the open source community
Describe open source
Describe GitHub Sponsors
Describe how GitHub advances open source projects
Identify how to follow people (receive notifications, discover projects in their community)
Explain how to follow organizations (receive notifications about their activity)
Describe the GitHub Marketplace and its purpose

Describe how to apply the benefits of open source
Describe InnerSource
Identify the differences between InnerSource and open source
Describe forking
Describe the components of a discoverable repository
Describe when to use issue templates
Describe when to use pull request templates