

PRACTICE QUESTIONS

Microsoft

GH-600

GitHub Certified: Agentic AI Developer

SHARED BY [u/jazzybeacon1995](#) on aiexamcerts

EDITION	Free Set 2
QUESTIONS	10
SHARED	October 18, 2026
EXAM	Microsoft GH-600, GitHub Certified: Agentic AI Developer

Shared by u/jazzybeacon1995 on aiexamcerts

10 practice questions for the Microsoft GH-600 exam, GitHub Certified: Agent AI Developer, each followed by its answer and a short explanation. More free sets and the full practice exam: aiexamcerts.com/microsoft/gh-600. Spotted a mistake? Email support@aiexamcerts.com with the exam code GH-600 and the question number.

1. A team wants an agent to create branches and pull requests on its own, but a person must still decide whether anything merges. Which setup meets both goals?

- A The agent opens draft pull requests and required human reviews gate every merge
- B The agent has write access to the default branch so it can merge as soon as tests pass
- C The agent merges its own pull requests, and a weekly report lists them for review
- D The agent is blocked from creating branches, and developers copy its code by hand

Answer: A

Draft pull requests let the agent do the work while required human review decides what merges, which keeps both speed and control. Direct merge rights are tempting because passing tests feels like sufficient evidence, but tests do not replace review. Reviewing after the fact is too late for a merge that already happened.

2. Which description best defines a success criterion for an agent task?

- A The total runtime of the agent in minutes, measured from the moment the task was assigned until the agent posts its final message in the thread
- B An observable condition the output must meet, such as passing tests and changes limited to a stated set of files
- C The number of tool calls the agent made during the run
- D The confidence score the model reports for its own answer

Answer: B

A success criterion is an observable condition, such as passing tests or a bounded set of changed files, that a reviewer can check. Runtime and tool call counts are measurable but say nothing about whether the output is right. A self-reported confidence score is tempting because it is a number, but it is not an independent check.

3. Once an agent has been given a structured plan, monitoring its runs is unnecessary, because the plan guarantees the outcome.

- A True
- B False

Answer: B (False)

A plan is a proposal about what the agent intends to do, and it does not guarantee what happens during execution. Logs, traces, and workflow artifacts are still needed to confirm that the run followed the approved plan. The plan is tempting as a safety net, but it only makes monitoring easier.

4. Which repository file controls the tools, dependencies, and build steps that a Copilot coding agent sets up before it starts working?
- A The README badge section of the repository home page
 - B The billing contact stored at the organization level
 - C The copilot-setup-steps workflow in the .github/workflows folder
 - D The CODEOWNERS file that assigns reviewers to each path, and that also decides which packages the agent installs during its setup steps

Answer: C

The copilot-setup-steps workflow defines the environment the coding agent starts in, including dependencies and setup commands. The CODEOWNERS file is tempting because it is also read during review, but it assigns reviewers rather than configuring the environment. The README and billing contact have no effect on the agent's setup.

5. A CI workflow should invoke an agent only on pull requests that carry a specific label. Which trigger configuration does this?
- A on: push with no branch filter, so every commit runs the agent
 - B on: schedule with a cron entry that runs every five minutes, so the agent checks every open pull request on that timer and reacts to its labels then
 - C on: workflow_dispatch with no inputs, started manually by a developer
 - D on: pull_request with the labeled event type, and an if condition that checks for the label name

Answer: D

The pull_request trigger with the labeled type fires when a label is added, and an if condition can then check for the specific label. A push trigger is tempting because it is the common default, but it fires on every commit regardless of labels. Scheduled and manual triggers do not depend on pull request labels at all.

6. In a GitHub Actions workflow, the pull_request trigger can be narrowed to particular target branches by adding a branches _____ under that trigger.

Answer: filter

A branches filter under the pull_request trigger limits the workflow to pull requests that target the listed branches. A paths filter is tempting when the goal is to limit by changed files rather than by target branch. The accepted answer is filter, with the plural filters as a variant.

7. An agent needs to call a deployment API. Which credential approach keeps its access least privileged?
- A A short-lived token scoped only to the single deployment action and its target environment
 - B A long-lived organization owner token stored as a plain-text environment variable
 - C Shared administrator credentials copied from a developer's laptop
 - D The token that protects the repository's default branch settings, which also grants write access to every environment and secret the organization owns

Answer: A

A short-lived token scoped to one action and one environment limits the damage if it leaks and grants no more than the deployment needs. A long-lived owner token is tempting because it never expires during a run, but it grants far more access than required. Copied administrator credentials and branch protection tokens are both overly broad.

8. Which error-handling practice fits an agent that calls an external API that sometimes returns HTTP 429 (too many requests)?
- A Retry immediately in a tight loop until the call succeeds, with no delay between attempts and no upper limit on how many times the retry may run
 - B Retry with exponential backoff, and escalate to a human after a fixed number of attempts
 - C Treat every error as a success so that the workflow continues without stopping
 - D Restart the entire pipeline from its first step each time a call fails

Answer: B

Exponential backoff spaces out retries so the rate limit can clear, and a cap with escalation prevents an endless loop. An immediate retry loop is tempting because it is the simplest code, but it makes the rate limit worse. Treating errors as success and restarting the pipeline both hide the failure or waste work.

9. Which two properties make a rollback plan for an agent's file changes reliable? (Choose two.)
- A Each agent action is committed separately, so any single step can be reverted through version control
 - B The agent overwrites the previous version of each file in place
 - C A recorded list of the actions taken, with the state before each one
 - D The agent performs the rollback itself with no record of what it reverted
 - E Rollback is skipped for changes that the agent has already reviewed itself

Answer: A, C

Separate commits let a reviewer revert one step cleanly, and a recorded list of actions with prior state shows exactly what to undo. Overwriting files in place is tempting for simplicity, but it destroys the earlier state. Self-performed rollbacks with no record and skipping rollback for self-reviewed work both make recovery unverifiable.

10. An agent is configured with a repository-scoped token and tries to push to a second repository it is not configured for. What is the most likely result?
- A The push succeeds because the agent inherits every repository the user can see
 - B The push is queued until a person approves it manually, and the workflow waits in that queue for as long as the approval takes to arrive
 - C The push is refused by the permission check, and the refusal is recorded as an environment constraint
 - D The agent switches to the second repository's default branch automatically

Answer: C

A token scoped to one repository is refused when used against another, and that refusal should be logged as an environment constraint for diagnosis. Inheriting every visible repository is tempting because it describes a user account, but the agent is bound to its token. Automatic branch switching and manual queueing are not how the permission boundary works.