Multi Choice Type Question In Git, what is the purpose of the staging area (also known as the index)? Marks : 1 Negative Marks : 0 image Answer here To store backup copies of files To hold changes that are ready to be committed To create a new branch for development To store remote repositories' information

The correct answer is:

**To hold changes that are ready to be committed**

The staging area (or index) in Git is where you place changes that you want to include in your next commit. It acts as a buffer between your working directory and the repository.

**You said:**

Multi Choice Type Question When you clone a Git repository, what do you get in your local workspace? Marks : 1 Negative Marks : 0 image Answer here Only the latest commit in the master branch The entire version history and all branches Only the staged changes The remote repository's settings

The correct answer is:

**The entire version history and all branches**

When you clone a Git repository, Git copies the full repository—including all commits, branches, and tags—into your local workspace, giving you a complete copy of the project’s history.

**You said:**

Multi Choice Type Question What command is used to save changes to the local repository in Git? Marks : 1 Negative Marks : 0 image Answer here git push git commit git fetch git add

The correct answer is:

**git commit**

In Git, git commit is used to save the staged changes from the staging area to the local repository. git add stages changes, and git push sends commits to a remote repository.

**You said:**

Multi Choice Type Question In Git, how do you discard local changes and revert to the last committed state of a file? Marks : 1 Negative Marks : 0 image Answer here git reset git checkout git discard git clean

The correct answer is:

**git checkout**

In Git, git checkout -- <filename> is used to discard local changes in a file and revert it back to the last committed state. (git reset is used for un-staging changes, and git clean removes untracked files.)

**You said:**

Multi Choice Type Question In Git, what is a "branch"? Marks : 1 Negative Marks : 0 image Answer here A specific folder in the repository A separate working directory on your computer A pointer to a specific commit in the version history A backup of the entire repository

The correct answer is:

**A pointer to a specific commit in the version history**

In Git, a branch is essentially a movable pointer to a commit. It allows you to develop features or fixes independently without affecting the main codebase until you merge changes.

**You said:**

Multi Choice Type Question What is a "merge conflict" in Git? Marks : 1 Negative Marks : 0 image Answer here When two branches are merged successfully When there are conflicts between the contents of the working directory and the staging area When multiple users try to push changes simultaneously When there are conflicts between the changes in different branches that are being merged

The correct answer is:

**When there are conflicts between the changes in different branches that are being merged**

A merge conflict occurs when Git cannot automatically reconcile differences between branches during a merge, requiring manual resolution.

**You said:**

Multi Choice Type Question What command is used to create a new branch in Git? Marks : 1 Negative Marks : 0 image Answer here git branch git new-branch git create-branch git checkout

The correct answer is:

**git branch**

In Git, git branch <branch-name> is used to create a new branch. To switch to it immediately, you can use git checkout <branch-name> or the combined command git switch -c <branch-name>.

**You said:**

Multi Choice Type Question What does "HEAD" represent in Git? Marks : 1 Negative Marks : 0 image Answer here The most recent commit in the repository The main branch of the repository The staging area where changes are prepared for commit The remote repository's URL

The correct answer is:

**The most recent commit in the repository**

In Git, HEAD is a pointer that refers to the current commit you are working on, usually the tip of the current branch. It moves as you make new commits or switch branches.

**You said:**

Multi Choice Type Question How can you resolve a merge conflict in Git? Marks : 1 Negative Marks : 0 image Answer here By deleting the conflicting files and re-creating them By creating a new branch for the conflicting changes By picking one version and discarding the other conflicting changes By reverting to the previous commit

The correct answer is:

**By picking one version and discarding the other conflicting changes**

To resolve a merge conflict in Git, you manually choose which changes to keep—either from one branch, the other, or a combination—then stage the resolved files and commit the merge.

**You said:**

Multi Choice Type Question In Git, how can you create a new branch and switch to it at the same time? Marks : 1 Negative Marks : 0 image Answer here git new-branch [branch\_name] git create-branch [branch\_name] git checkout [branch\_name] git checkout -b [branch\_name]

The correct answer is:

**git checkout -b [branch\_name]**

This command creates a new branch named [branch\_name] and immediately switches to it in a single step.

**You said:**

Multi Choice Type Question How do you delete a file from the working directory and stage the deletion? Marks : 1 Negative Marks : 0 image Answer here git delete filename git rm filename git remove filename git discard filename

The correct answer is:

**git rm filename**

In Git, git rm <filename> removes the file from the working directory and stages the deletion so it will be included in the next commit.

**You said:**

Multi Choice Type Question How do you recover a deleted file that has not yet been committed? Marks : 1 Negative Marks : 0 image Answer here git restore filename git undo filename git recover filename git reset filename

The correct answer is:

**git restore filename**

In Git, git restore <filename> restores a deleted (or modified) file in the working directory to its last committed state if the deletion has not yet been committed.

**You said:**

Multi Choice Type Question Which command will show you changes made to a file in your working directory compared to the staging area? Marks : 1 Negative Marks : 0 image Answer here git status git diff git log git changes

The correct answer is:

**git diff**

In Git, git diff shows the differences between your working directory and the staging area, highlighting changes that have not yet been staged for commit.

**You said:**

Multi Choice Type Question What command do you use to check which files are currently staged for the next commit? Marks : 1 Negative Marks : 0 image Answer here git status git staged git diff --cached git commit --list

The correct answer is:

**git status**

In Git, git status displays the files that are staged for the next commit as well as untracked and modified files in the working directory.

**You said:**

Multi Choice Type Question Which command is used to view the history of changes for a specific file? Marks : 1 Negative Marks : 0 image Answer here git history filename git log filename git diff filename git changes filename

The correct answer is:

**git log filename**

In Git, git log <filename> shows the commit history related to that specific file, including commit messages, authors, and timestamps.

Top of Form