### ****Part 1: Git Basics****

1. **git init initializes a new repository.** ✅
2. **git add stages files for commit.** ✅
3. **.gitignore lists files and directories to be ignored by Git.** ✅
4. **A Git repository is a local directory with Git version control.** ✅
5. **git --version checks the current Git version.** ✅
6. **The .git/ folder contains the repository metadata.** ✅
7. **The default branch in Git (historically) is master.** ✅
8. **git status shows the status of your repository.** ✅
9. **git clone copies an existing repository to a new directory.** ✅
10. **git rm removes files from the working directory and staging area.** ✅

### ****Part 2: Git Commits****

1. **git commit records changes to the repository.** ✅
2. **git commit -m commits changes with a message.** ✅
3. **git log shows the commit history.** ✅
4. **git reset HEAD~1 --soft undoes the last commit without losing changes.** ✅
5. **git commit --amend amends the last commit.** ✅
6. **A commit hash is a unique identifier for each commit.** ✅
7. **HEAD represents the currently checked-out branch.** ✅
8. **git reset --hard discards unstaged changes.** ✅
9. **git diff shows all differences (between commits, staged changes, and working directory).** ✅
10. **A commit message describes changes in a commit.** ✅

### ****Part 3: Branching and Merging****

1. **Both git branch <branch-name> and git checkout -b <branch-name> create a new branch.** ✅
2. **git branch -d <branch> deletes a branch.** ✅
3. **git merge merges two branches.** ✅
4. **A merge conflict indicates overlapping changes in branches.** ✅
5. **git branch lists all branches in the repository.** ✅
6. **Both git checkout <branch-name> and git switch <branch-name> switch branches.** ✅
7. **Both git rebase <branch> and git pull --rebase rebase a branch.** ✅
8. **A fast-forward merge moves the HEAD pointer to the target branch.** ✅
9. **git branch -m <new-name> renames a branch.** ✅
10. **git stash saves changes temporarily.** ✅

### ****Part 4: Remote Repositories****

1. **git remote lists remote repositories.** ✅
2. **git push pushes changes to the remote repository.** ✅
3. **git pull pulls changes from the remote repository.** ✅
4. **git clone <url> clones a remote repository.** ✅
5. **git fetch downloads changes without merging.** ✅
6. **git remote add <name> <url> adds a remote repository.** ✅
7. **origin refers to the default remote repository name.** ✅
8. **git branch -r shows remote branches.** ✅
9. **git push origin --delete <branch> deletes a remote branch.** ✅
10. **git pull downloads changes and merges them into the current branch.** ✅

### ****Part 5: Advanced Git****

1. **git rebase moves or rewrites commits to a different base.**
2. **git tag lists tags in a repository.**
3. **git tag -a <tag-name> -m "<message>" creates an annotated tag.**
4. **git fetch downloads changes but does not merge them, whereas git pull downloads and merges them.**
5. **git cherry-pick <commit-hash> applies a specific commit to the current branch.**
6. **Merge conflicts are resolved by manually editing conflicting files, staging them, and committing the changes.**
7. **git log tracks changes in Git using logs.**
8. **git reset moves the HEAD to a previous commit, optionally modifying the working directory.**
9. **git reset --hard <commit> performs a hard reset, erasing all changes.**
10. **Git hooks are scripts that run automatically before or after Git events, such as commits or pushes.**

Subversion (SVN)