Explain the following in detail

**1.Git Remote:** In Git, git remote is a command-line utility that allows you to manage remote repositories associated with your local repository. Here are some common subcommands and their purposes:

* **git remote -v**: Lists all remote repositories associated with the current repository along with their URLs. The -v option shows the URLs of remotes.
* **git remote add <name> <url>**: Adds a new remote repository with the specified name and URL.

**2.Git clone:** git clone is a Git command used to create a copy of a remote repository on your local machine.

**Usage:** git clone <repository-url> [<directory>]

**Explanation:**

* **<repository-url>:** Specifies the URL of the remote repository you want to clone. This can be a URL from services like GitHub, GitLab, Bitbucket, or even a custom Git server.
* **[<directory>] (optional):** Specifies the directory name where the cloned repository will be placed. If not provided, Git will create a new directory using the name of the cloned repository.

**3.Git push:** git push is a Git command used to upload local repository content to a remote repository.

**Usage:** git push <remote> <branch>

**Explanation:**

* **<remote>:** Specifies the name of the remote repository where you want to push your changes. Typically, this is origin, which is the default name Git gives to the remote repository you cloned from or added.
* **<branch>:** Specifies the branch on your local repository that you want to push to the remote repository.

**4.Git status:** git status is a Git command that displays the state of your working directory and staging area.

**Usage**: git status

When you run git status, Git provides information about:

* Which files are currently ready to be committed.

**5.Git log:** git log is a Git command used to display the commit history of a repository. It shows a list of commits in reverse chronological order (newest commits first).

**Usage:** git log

**Explanation:**

* When you run git log, Git displays a detailed list of commits, typically showing the following information for each commit:
  + Commit hash (a unique identifier for each commit).
  + Author information (name and email).
  + Date and time of the commit.

**6.Git restore**: In Git, git restore is a command used to restore files in your working directory to their state at a specific point in time.

**Usage:** git restore <file>

**Explanation:**

* **<file>:** Specifies the file or files you want to restore to their previous state

**7.Difference between git fetch and git pull**

