Assignment Day 4

1. **How to check if git is available on your system?**

Ans. The command $ git --version or $ git -v is used to check if the git is installed in our system or not, if it’s not installed in our system then an error will come. With the help of this command we can also know which version of the git is installed in our system.

2. **How to initialize a new git repository?**

Ans. To initialize a new git repository, follow these steps:

a. Navigate to the directory that you want to turn into a git repository using the terminal or command prompt.
b. Run the following command: $git init

This will create a new git repository in the current directory. You can verify that the repository was created by running git status, which should display the current status of the repository.

Note: Initializing a repository will create a new .git directory in the current directory, which will contain all the necessary files and metadata for the repository. Do not modify the contents of this directory directly, as it is managed by git.

3. **How to tell git about your name and email?**

Ans. You can tell git about your name and email by using the following commands:-

```
$ git config --global user.name "Your Name"
```

```
$ git config --global user.email "your.email@example.com"
```

The --global option sets your name and email globally, so they will be associated with all of your git repositories. If you want to set your name and email for a specific repository, you can run these commands without the --global option while inside that repository's directory.You can verify that your name and email were set correctly by running the following command:-

```
$ git config --list
```

This will list all of your git configuration settings, including your user name and email.

Assignment Day 4

4. **How to add a file to the staging area?**

Ans. To add a file to the staging area in Git, use the following command in your terminal or command prompt:-

```
$ git add <file_name>
```

Replace <file_name> with the name of the file you want to add to the staging area. You can also use . to add all files in the current directory and its subdirectories, or use a wildcard pattern to add multiple files. For example:-

```
$ git add .

$ git add *.txt
```

5. **How to remove a file from the staging area?**

Ans. To remove a file from the staging area in Git, use the following command in your terminal or command prompt:-

```
$ git rm - - cached <file>
```

This command is used to move/remove files from staging area to working area.

6. **How to make a commit?**

Ans. To make a commit in Git, you need to first add your changes to the staging area using the git add command, then use the following command in your terminal or command prompt:-

```
$ git commit -m "Your commit message here"
```

Replace "Your commit message here" with a brief and meaningful description of the changes you have made. The commit message should explain why the changes were made and what they do.

Assignment Day 4

7. How to send your changes to a remote repository?

Ans. To send your changes to a remote repository in Git, you need to first make a commit and then use the following command in your terminal or command prompt:-

```
$ git push <remote_name> <branch_name>
```

Replace <remote_name> with the name of your remote repository (e.g., origin) and <branch_name> with the name of the branch you want to push to the remote repository (e.g., master). If you have set up your remote repository as the default upstream branch for your current branch, you can use a simplified version of the above command:-

```
$ git push
```

This will push your changes to the remote repository and update the remote branch with your latest commits. If there are any conflicts between your local branch and the remote branch, you need to resolve them before you can push your changes.

8. What is the difference between clone and pull?

Ans. In git, "git clone" and "git pull" are two different Git commands that are used to retrieve code from a remote repository.

"git clone" is used to create a local copy of a remote repository. When you run git clone, Git downloads the entire repository, including all its files and history, to your local machine. The local copy of the repository is automatically set up as a remote named origin and you can start making changes and committing them to your local branch.

"git pull" is used to update your local copy of a remote repository with the latest changes from the remote. When you run git pull, Git fetches the latest changes from the remote repository and merges them into your current branch. git pull is used to synchronize your local repository with the remote repository and bring it up to date.

In summary, git clone is used to create a local copy of a remote repository, while git pull is used to update a local repository with changes from a remote repository.