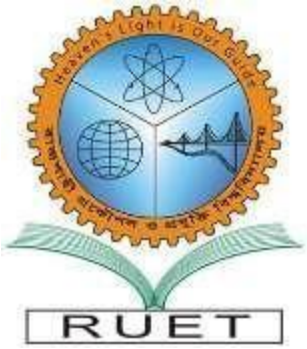
Heaven's Light Is Our Guide

Rajshahi University of Engineering & Technology



Department of Electrical & Computer Engineering

Experiment No: 01

Course Code: ECE-3118

Course Name: Software Engineering & Information Design Sessional

Submitted To:

Oishi Jyoti

Assistant Professor,

Dept. of ECE, RUET

Submitted By:

Humaira Tasnim Adiba

Roll: 2010002

Year: 3rd year (ODD)

Experiment No: 01

Name of the Experiment: Study of different GIT commands.

Objectives:

- To get speed, data integrity, and support for distributed, non-linear workflows.
- Git's branching features are among its greatest benefits.
- Git branches are inexpensive and simple to combine, in contrast to centralized version control systems.
- This makes it easier to employ the feature branch process, which is very popular among Git users.
- Feature branches give any update to your codebase an isolated environment.

Theory:

Git is a distributed version control system that tracks changes in any set of computer files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows.[1] By far, the most widely used modern version control system in the world today is Git. It is developed to co-ordinate the work among the developers. The version control allows us to track and work together with our team members at the same workspace. It is a version control application that allows us to keep track of all the changes that we make in the files of our project. Every time we make changes in files of an existing project, we can push those changes to a repository. Other developers are allowed to pull your changes from the repository and continue to work with the updates that you added to the project files.[2] Some of the basic operations in Git are:

- Initialize
- Add
- Commit
- Pull
- Push

Git Commands:

1. **Git Config Command:**
   This command configures the user. This command sets the author's name and email address to be used with your commits

```
Tasnim@Adiba MINGW64 ~
$ git config --global user.name "adibaruet"

Tasnim@Adiba MINGW64 ~
$ git config --global user.email "2010002@student.ruet.ac.bd"
```

2. **Git init Command:**
   This command is used to create a local empty repository.

```
Tasnim@Adiba MINGW64 ~
$ git init
Initialized empty Git repository in C:/Users/Tasnim/.git/
```

3. **Git Add Command:**
   This command is used to add one or more files to staging area.

```
Tasnim@Adiba MINGW64 ~ (master)
$ git add git.txt
```

4. **Git commit Command:**
   This command commits any files added in the repository with git add and also commits any files you've changed since then.

```
Tasnim@Adiba MINGW64 ~ (master)
$ git commit -m "hello"
```

5. **Git Clone Command:**

   This command is used to make a copy of a repository from an existing URL.

```
Tasnim@Adiba MINGW64 ~ (master)
$ git clone https://githib.com//git
Cloning into 'git'...
```

6. Git Branch Command:

This command lists all the branches available in the repository.

7. Git Status Command:

The status command is used to display the state of the working directory and the staging area.