Department of
Computer Science and Engineering

Title: Version Control Part II (Using Git)

Software Testing & Quality Assurance Lab CSE 434

Green University of Bangladesh

1 Objective(s)

• Ability to rename local and remote Git branch.

• Ability to create and delete local and remote Git branch.

2 How To Rename a Local and Remote Git Branch

Git is a software package used for tracking software as it moves through stages of development. Git uses branching to maintain a central repository of code while creating a copy to make changes on. In this guide, learn how to change the name of a Git branch on a local system or remote repository.

2.1 Rename Local Branch

To rename a branch in Git:

1. Enter the name of the branch you want to change in your command-line interface:

```
git checkout old-branch
```

You should receive confirmation that the branch is checked out.

```
[phoenixnap@localhost git-app]$ git checkout old-branch
Switched to branch 'old-branch'
```

2. Rename the branch by entering the command:

```
git branch -m new-name
```

Alternatively, you can use a single command. If you're not already in the master, switch to it:

```
git checkout master
```

Enter the following to change a branch name:

```
git branch -m old-name new-name
```

3. Verify the renaming was successful by checking the status :

```
git branch -a
```

The output confirms that the branch was successfully renamed, as shown below.

```
[phoenixnap@localhost git-app]$ git branch -m new-name
[phoenixnap@localhost git-app]$ git branch -a
  master
* new-name
```

This is useful if you created a new branch and pushed your remote repository's changes to discover the branch name was incorrect.

Note: Replace old-name with the actual name of the branch you want to change. Replace new-name with the name of the branch you want to use going forward.

2.2 Rename a Remote Git Branch

1. Verify the local branch has the correct name:

```
git branch -a
```

2. Next, delete the branch with the old name on the remote repository: item Verify the local branch has the correct name:

```
git push origin --delete old-name
```

The output confirms that the branch was deleted.

```
[phoenixnap@localhost git-app]$ git push origin --delete old-branch
 - [deleted]          old-branch
```

3. Finally, push the branch with the correct name, and reset the upstream branch:

```
git push origin -u new-name
```

Alternatively, you can overwrite the remote branch with a single command:

```
git push origin :old-name new-name
```

Resetting the upstream branch is still required:

```
git push origin -u new-name
```

Now you know how to rename a local or remote Git branch, even if it has been loaded on a remote repository.

3 How to Create a New Branch in Git

Git is an open-source version-control system for tracking changes during the software development life cycle. It's mutually independent branching model makes it stand out. Branches can be based on previous versions of the software to maintain the integrity of current progress while working on a bug fix or new feature. This guide will detail multiple options to create a new branch in Git.

3.1 Create a New Git Branch

There are many ways to create a new Git branch. In most cases it comes down to whether you are creating a branch from the main branch or, for example, a new commit or tag.

1. One common method of creating a new branch is with the command:

This does not automatically switch to that branch. To switch Git branches, enter the following command:

3.2 Create New Git Branch From Current Branch

1. The easiest and most popular way of creating a Git branch is:

This creates a new branch from the current branch. It also automatically switches to the new branch.

3.3 Create New Git Branch From a Different Branch

1. To create a new branch from a different branch, run the following command:

3.4 Create a Branch from a Commit

A commit is a command that saves the changes made in the code. A project may have multiple commits as it's revised and improved.

1. Find the hash key for a specific commit:

2. The log contains the hash key.

3. Create a branch from an older commit:

No need to enter the whole hash key, just the first few characters. View the git log again, and you'll see the new branch listed.
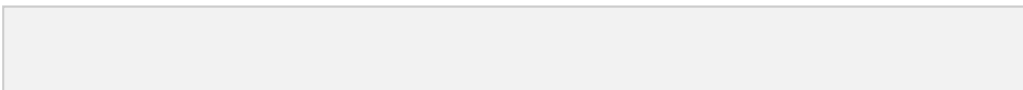
This method is especially helpful if you need to go back to a previous version of the software to fix a bug without removing any existing features.

Use git checkout to switch to the newly created branch.

3.5 Create a Branch from a Tag

A tag is a final, unchangeable version of a commit. Where a commit can be edited, tagged versions are usually permanent.

1. To create a branch from this tag, use the command:

2. To switch to this branch:

3.6 Create a Branch Using Detached HEAD State

Detached HEAD state happens when you check out a commit that's not formally part of a branch.

1. To test, use git log to get the hash of a commit, then enter:

Replace 6009fc with the actual hash value from the system. The system prints the following output:

You are in 'detached HEAD' state. You can look around, make experimental changes and commit them, and you can discard any commits you make in this state without impacting any branches by performing another checkout.
Just like the warning outlines, you can make changes based on the commit. Changes are lost if you don't save them.

2. To save any changes, stage it and then enter the following:

3. To add the changes into the master, use the following:

3.7 Create a Branch from a Remote Branch

1. To create a new branch locally based on an existing remote branch, use the –track option:

2. Alternatively, use the git checkout command to keep the original remote branch name:

The git checkout command automatically creates the remote branch locally with the original name.

3.8 Create a Branch in a Remote Repository

Use the git push command to create a new branch in a remote repository based on a local

branch:

The command automatically creates the branch in a remote repository. The -u option ensures a tracking connection.

3.9 How to Delete a Git Branch

To delete a git branch use the command:

The output confirms that the branch has been deleted.

4 Lab Task (Please implement yourself and show the output to the instructor)

1. Rename a local and remote Git branch.

2. Create and delete a local and remote Git branch.

5 Lab Exercise (Submit as a report)

1. Rename a local and remote Git branch.

2. Create and delete a local and remote Git branch.

6 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.

7 Resources

https://phoenixnap.com/kb/how-to-rename-git-branch-local-remote

https://phoenixnap.com/kb/git-create-new-branch