

# GIT - RENAME OPERATION

[http://www.tutorialspoint.com/git/git\\_rename\\_operation.htm](http://www.tutorialspoint.com/git/git_rename_operation.htm)

Copyright © tutorialspoint.com

Till now, both Tom and Jerry were using manual commands to compile their project. Now, Jerry decides to create Makefile for their project and also give a proper name to the file "string.c".

```
[jerry@CentOS project]$ pwd
/home/jerry/jerry_repo/project

[jerry@CentOS project]$ ls
README src

[jerry@CentOS project]$ cd src/

[jerry@CentOS src]$ git add Makefile

[jerry@CentOS src]$ git mv string.c string_operations.c

[jerry@CentOS src]$ git status -s
A Makefile
R string.c -> string_operations.c
```

Git is showing **R** before file name to indicate that the file has been renamed.

For commit operation, Jerry used -a flag, that makes git commit automatically detect the modified files.

```
[jerry@CentOS src]$ git commit -a -m 'Added Makefile and renamed strings.c to
string_operations.c '

[master 94f7b26] Added Makefile and renamed strings.c to string_operations.c
1 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 src/Makefile
rename src/{string.c => string_operations.c} (100%)
```

After commit, he pushes his changes to the repository.

```
[jerry@CentOS src]$ git push origin master
```

The above command will produce the following result:

```
Counting objects: 6, done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 396 bytes, done.
Total 4 (delta 0), reused 0 (delta 0)
To gituser@git.server.com:project.git
7d9ea97..94f7b26 master -> master
```

Now, other developers can view these modifications by updating their local repository.