

Stash operation in Git

Suppose you are implementing a new feature of your product. Your code is in progress, suddenly a visitor. Because of this, you have to go out for a few hours. You cannot commit this code and cannot throw it away at your changes. So you need some temporary space, where you can keep these local changes and then return to commit it.

Suppose you are implementing a new feature of your product. Your code is in progress, suddenly a visitor. Because of this, you have to go out for a few hours. You cannot commit this code and cannot throw it away at your changes. So you need some temporary space, where you can keep these local changes and then return to commit it.

In Git, stash operations help you stash edited files, changes, and save them on a stack of uncompleted changes, and from there you can re-apply at any time.

```
[jerry @ CentOS project] $ git status -s
M string.c
?? string
```

Now you want to switch the branch by an unexpected guest, but you don't want to commit what you did on it, so you stash the changes. To push these new stash into the stack, run git stash.

```
[jerry @ CentOS project] $ git stash
?ang ghi th? m?c làm vi?c và ch? m?c c? s? d? li?
u WIP trên master: e86f062 Added my_strcpy function
HEAD is now at e86f062 Added my_strcpy function
```

Now your working directory is clean and all changes are kept on a stack. Please verify it with the git status command.

```
[jerry @ CentOS project] $ git status -s
?? string
```

Now you can safely move the branch and do anything. We can observe a list of stash changes using the git stash list command.

```
[jerry @ CentOS project] $ git stash list
stash @ {0}: WIP on master: e86f062 Added my_strcpy function
```

Suppose you have solved the problem with the new customer and you go back to making your unfinished code, you just need to perform the git stash pop command, to move the changes from the stack and put them into Current working directory.

```
[jerry @ CentOS project] $ git status -s
?? string
```

```
[jerry @ CentOS project] $ git stash pop
```

The above command will produce the following result:

```
# On the master branch
# Changed but not updated:
# (use "git add ." to update what will be committed)
# (use "git checkout - ." to discard changes in directory working)
#
#
modified: string.c
#
# Untracked files:
# (use "git add ." to include in what will be committed)
#
#
string
không thay ??i thêm vào commit (s? d?
ng "git add" and / or "git commit -a")
Dropped refs / stash @ {0} (36f79dfedae4ac20e2e8558830154bd6315e72d4)

[jerry @ CentOS project] $ git status -s
M string.c
?? string
```

According to Tutorialspoint

Previous article: Update activity in Git

Next article: Online Repository in Git

You finished reading the article "**Stash operation in Git**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.