Git - Stash Operation

Suppose you are implementing a new feature for your product. Your code is in progress and suddenly a customer escalation comes. Because of this, you have to keep aside your new feature work for a few hours. You cannot commit your partial code and also cannot throw away your changes. So you need some temporary space, where you can store your partial changes and later on commit it.

In Git, the stash operation takes your modified tracked files, stages changes, and saves them on a stack of unfinished changes that you can reapply at any time.

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

Now, you want to switch branches for customer escalation, but you don’t want to commit what you’ve been working on yet; so you’ll stash the changes. To push a new stash onto your stack, run the **git stash** command.

```
[jerry@CentOS project]$ git stash
Saved working directory and index state WIP on master: e86f062 Added my_strcpy funct
HEAD is now at e86f062 Added my_strcpy function
```

Now, your working directory is clean and all the changes are saved on a stack. Let us verify it with the **git status** command.

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

Now you can safely switch the branch and work elsewhere. We can view a list of stashed changes by 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 resolved the customer escalation and you are back on your new feature looking for your half-done code, just execute the **git stash pop** command, to remove the changes from the stack and place them in the 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 branch master
# Changed but not updated:
# (use "git add ..." to update what will be committed)
# (use "git checkout -- ..." to discard changes in working directory)
#
#
modified: string.c
#
# Untracked files:
# (use "git add ..." to include in what will be committed)
#
#
string
no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (36f79dfedae4ac20e2e8558830154bd6315e72d4)

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