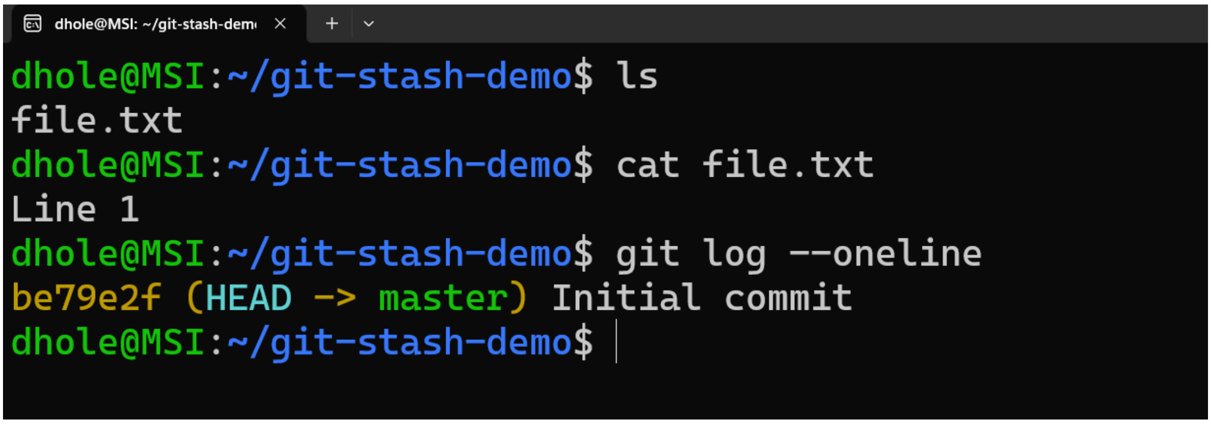
What is git stash?

git stash temporarily shelves (or stashes) changes you've made to your working directory so you can work on something else, then come back and re-apply them later.

📁 Practical Scenario Setup

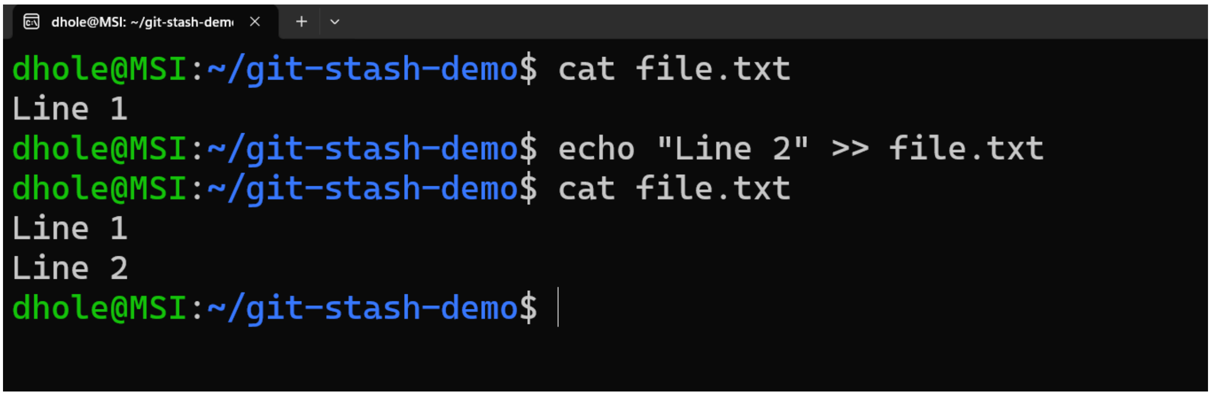
1. **Initialize a Git repo and create a file:**

mkdir stash-demo && cd stash-demo
git init
echo "Line 1" > file.txt
git add file.txt
git commit -m "Initial commit"



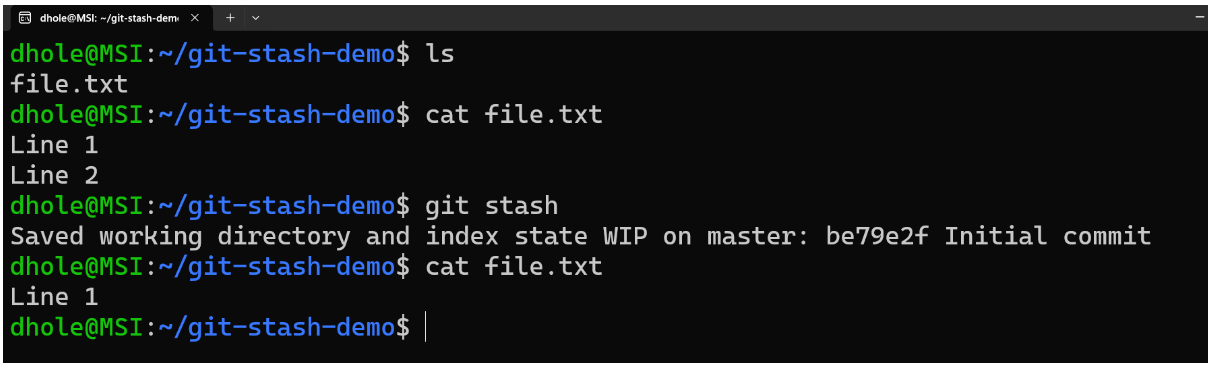
```
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
dhole@MSI:~/git-stash-demo$ git log --oneline
be79e2f (HEAD -> master) Initial commit
dhole@MSI:~/git-stash-demo$
```

Make some changes:
echo "Line 2" >> file.txt



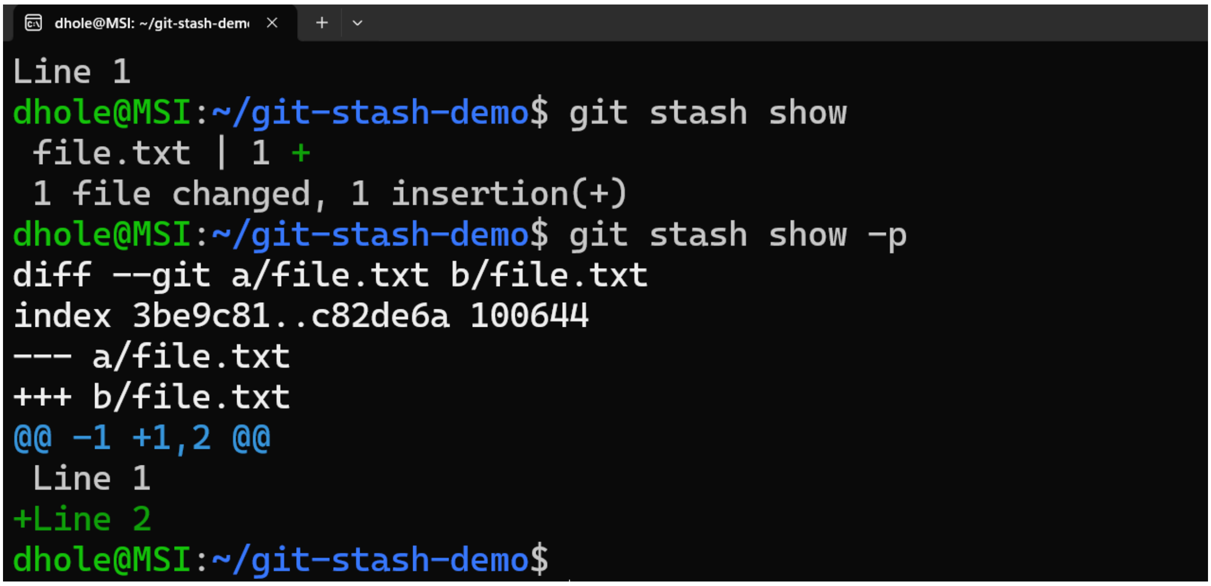
```
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
dhole@MSI:~/git-stash-demo$ echo "Line 2" >> file.txt
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
Line 2
dhole@MSI:~/git-stash-demo$
```

git stash – Save your changes



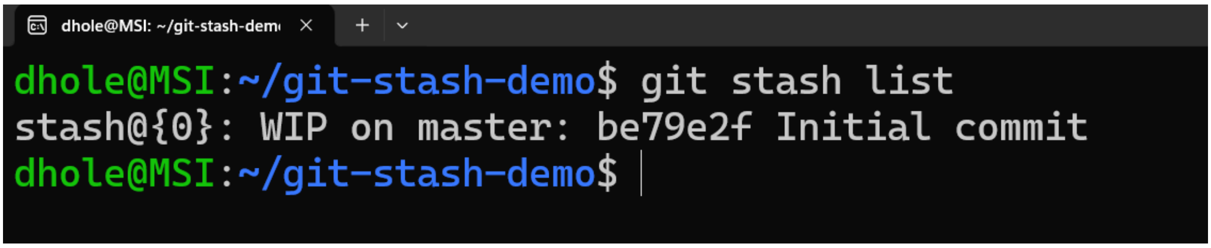
```
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
Line 2
dhole@MSI:~/git-stash-demo$ git stash
Saved working directory and index state WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
dhole@MSI:~/git-stash-demo$
```

git stash show – View summary of change and git stash show -p for full difference



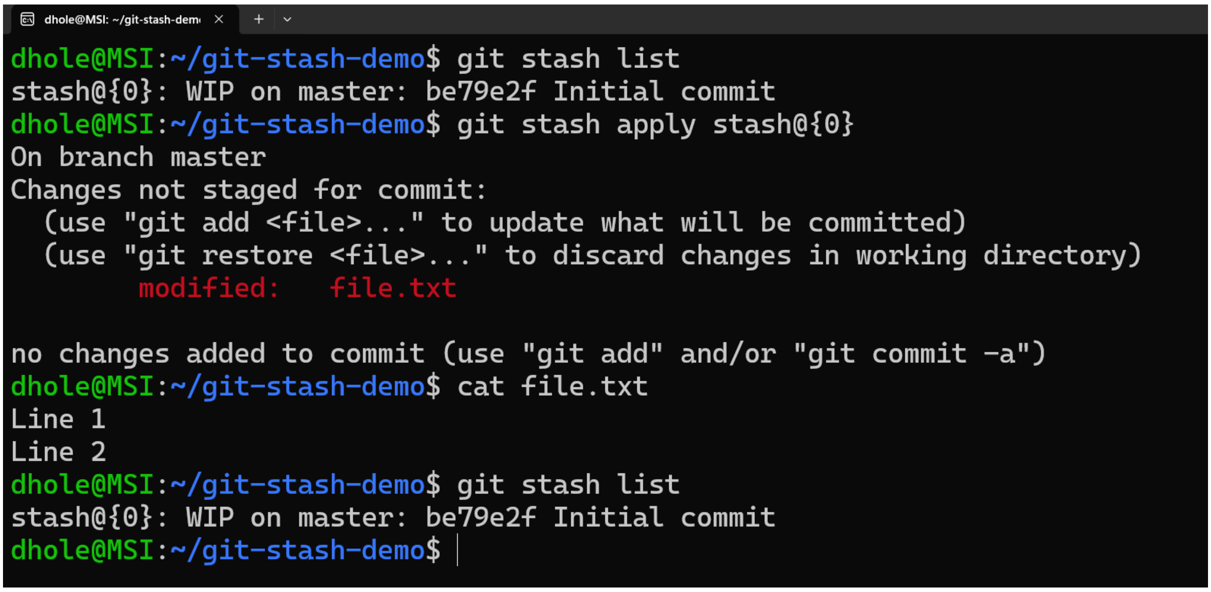
```
Line 1
dhole@MSI:~/git-stash-demo$ git stash show
 file.txt | 1 +
 1 file changed, 1 insertion(+)
dhole@MSI:~/git-stash-demo$ git stash show -p
diff --git a/file.txt b/file.txt
index 3be9c81..c82de6a 100644
--- a/file.txt
+++ b/file.txt
@@ -1 +1,2 @@
 Line 1
+Line 2
dhole@MSI:~/git-stash-demo$
```

git stash list for view stash



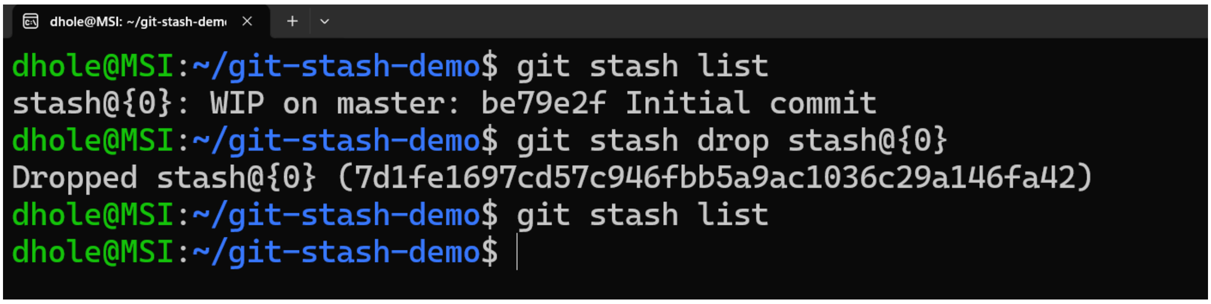
```
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$
```

git stash apply – Re-apply changes (keep stash) like copy paste
Flow—Stash memory-Copy-working Directory



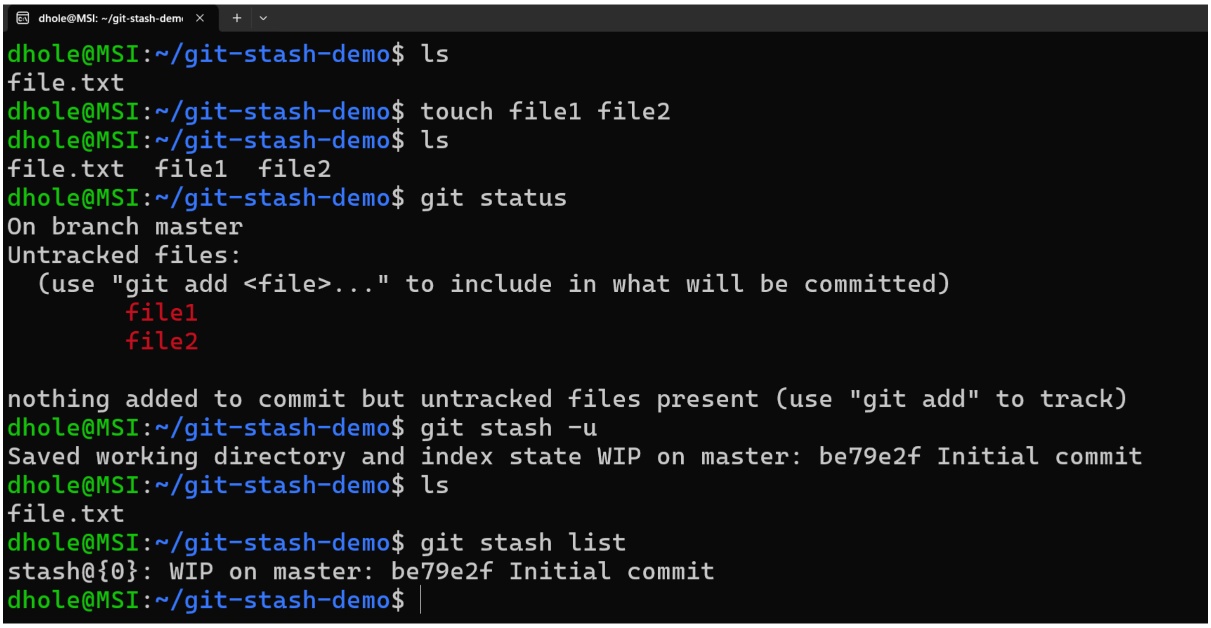
```
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ git stash apply stash@{0}
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   file.txt

no changes added to commit (use "git add" and/or "git commit -a")
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
Line 2
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$
```

git stash drop – Delete a specific stash



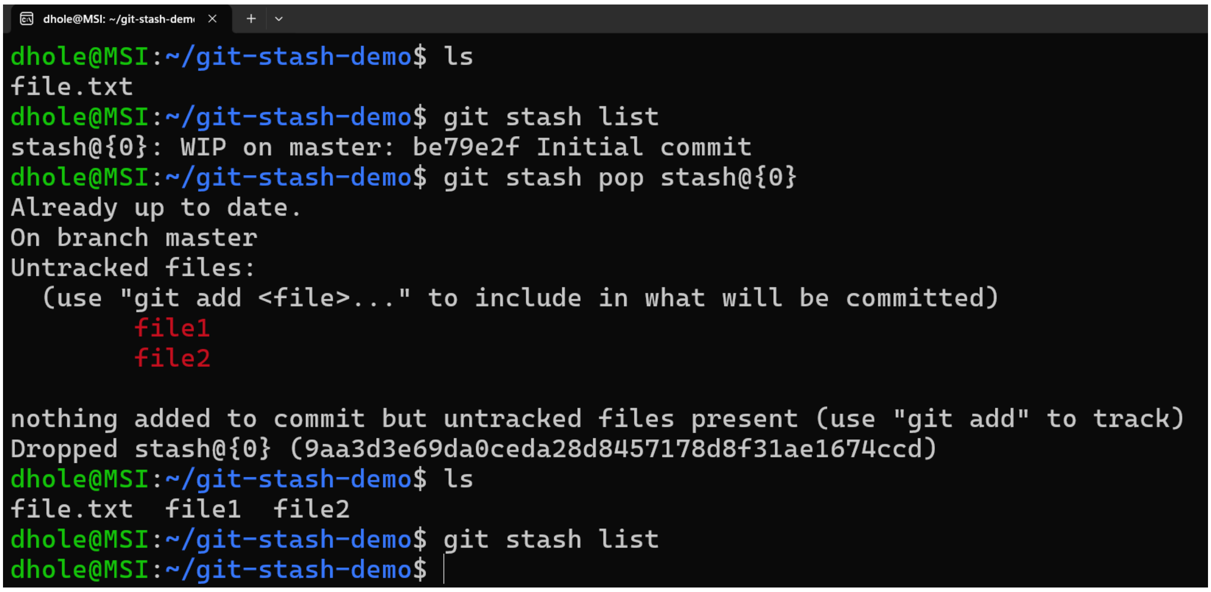
```
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ git stash drop stash@{0}
Dropped stash@{0} (7d1fe1697cd57c946fbb5a9ac1036c29a146fa42)
dhole@MSI:~/git-stash-demo$ git stash list
dhole@MSI:~/git-stash-demo$
```

git stash -u for untracked file stash



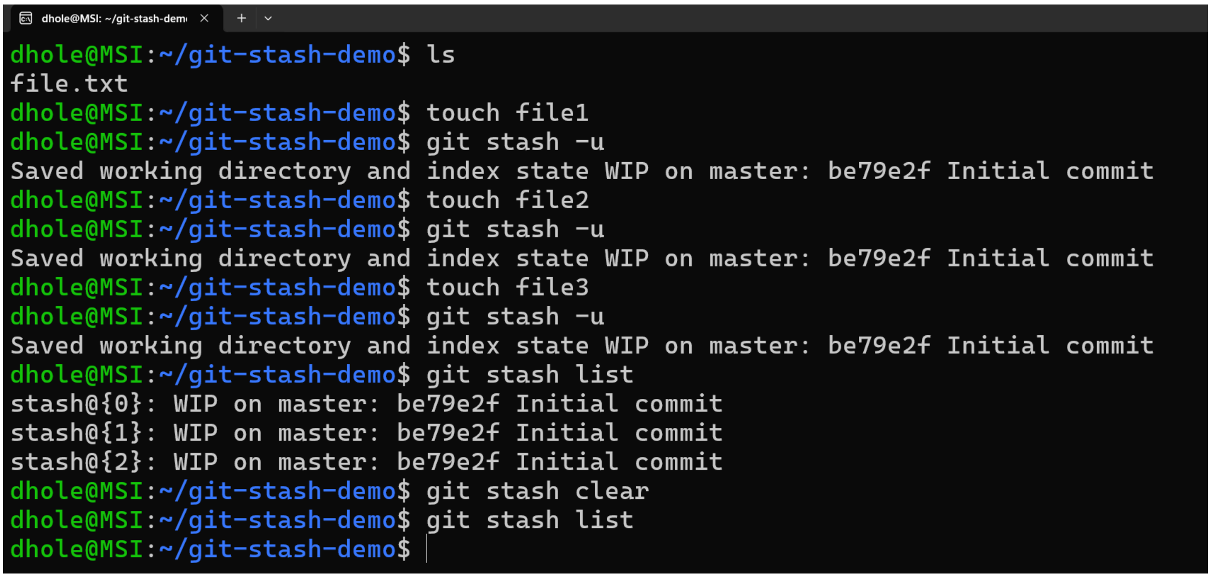
```
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ touch file1 file2
dhole@MSI:~/git-stash-demo$ ls
file.txt  file1  file2
dhole@MSI:~/git-stash-demo$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file1
        file2

nothing added to commit but untracked files present (use "git add" to track)
dhole@MSI:~/git-stash-demo$ git stash -u
Saved working directory and index state WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$
```

git stash pop - Cut paste
Stash memory-Cut-working Directory



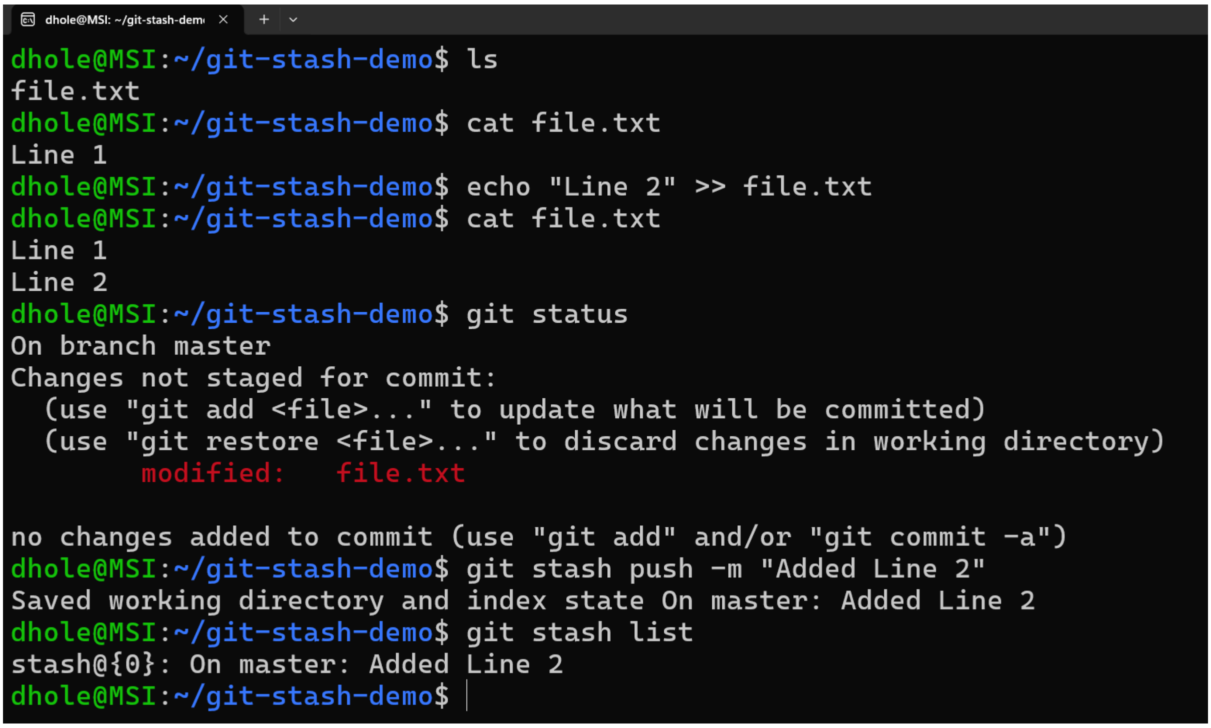
```
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ git stash pop stash@{0}
Already up to date.
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file1
        file2

nothing added to commit but untracked files present (use "git add" to track)
Dropped stash@{0} (9aa3d3e69da0ceda28d8457178d8f31ae1674ccd)
dhole@MSI:~/git-stash-demo$ ls
file.txt  file1  file2
dhole@MSI:~/git-stash-demo$ git stash list
dhole@MSI:~/git-stash-demo$ 
```

git stash clear all- Remove **all** stashes



```
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ touch file1
dhole@MSI:~/git-stash-demo$ git stash -u
Saved working directory and index state WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ touch file2
dhole@MSI:~/git-stash-demo$ git stash -u
Saved working directory and index state WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ touch file3
dhole@MSI:~/git-stash-demo$ git stash -u
Saved working directory and index state WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: WIP on master: be79e2f Initial commit
stash@{1}: WIP on master: be79e2f Initial commit
stash@{2}: WIP on master: be79e2f Initial commit
dhole@MSI:~/git-stash-demo$ git stash clear
dhole@MSI:~/git-stash-demo$ git stash list
dhole@MSI:~/git-stash-demo$ 
```

git stash push -m "Message" only work in Staged



```
dhole@MSI:~/git-stash-demo$ ls
file.txt
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
dhole@MSI:~/git-stash-demo$ echo "Line 2" >> file.txt
dhole@MSI:~/git-stash-demo$ cat file.txt
Line 1
Line 2
dhole@MSI:~/git-stash-demo$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   file.txt

no changes added to commit (use "git add" and/or "git commit -a")
dhole@MSI:~/git-stash-demo$ git stash push -m "Added Line 2"
Saved working directory and index state On master: Added Line 2
dhole@MSI:~/git-stash-demo$ git stash list
stash@{0}: On master: Added Line 2
dhole@MSI:~/git-stash-demo$
```