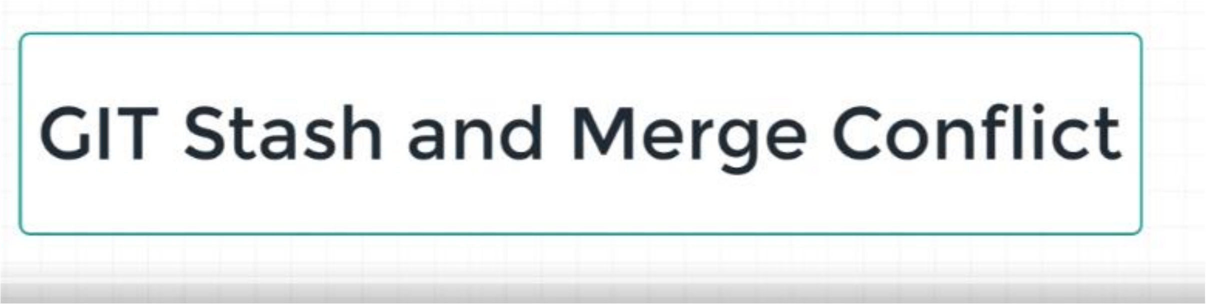


GIT Stash and Merge Conflict

What will happen if two people work on same functionality at the same time.

One of them merges the code for the feature with master.

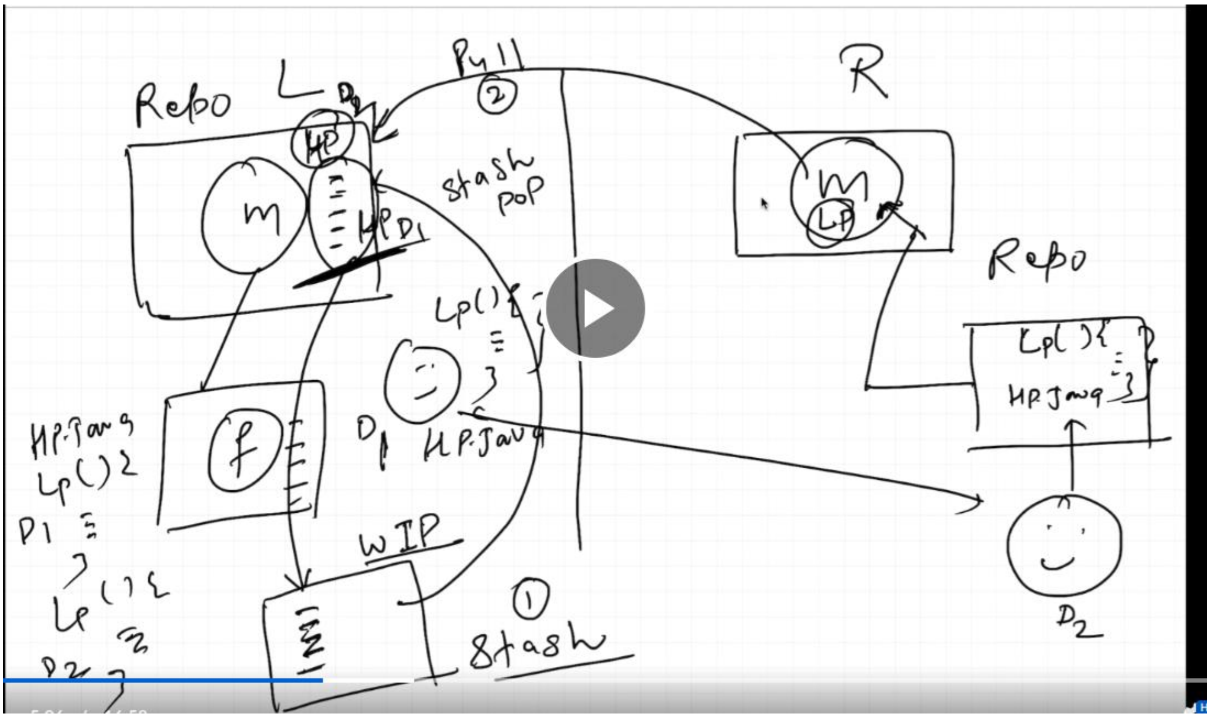
When other tries to take pull, then the other gets error. This is because there is some local change which is not yet committed. So, we need to at least commit the code in the local branch.

The other option is there is a memory called stash memory. We can move the wip code to stash memory and then take latest pull.

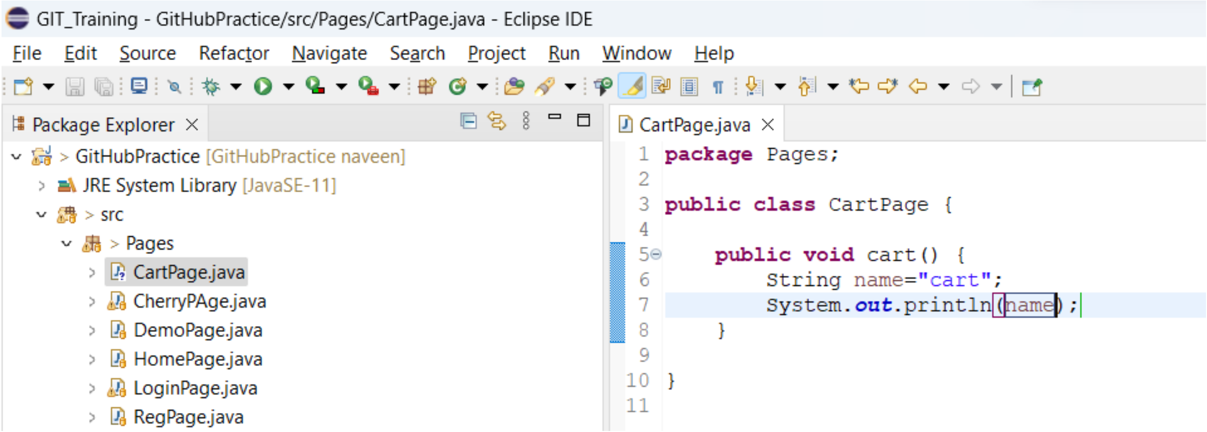
Stash pop –

After taking pull, why we do stash pop, so that our local code gets merged with the latest code which we pulled from remote branch.

We will get conflict here once we try to do stash pop.

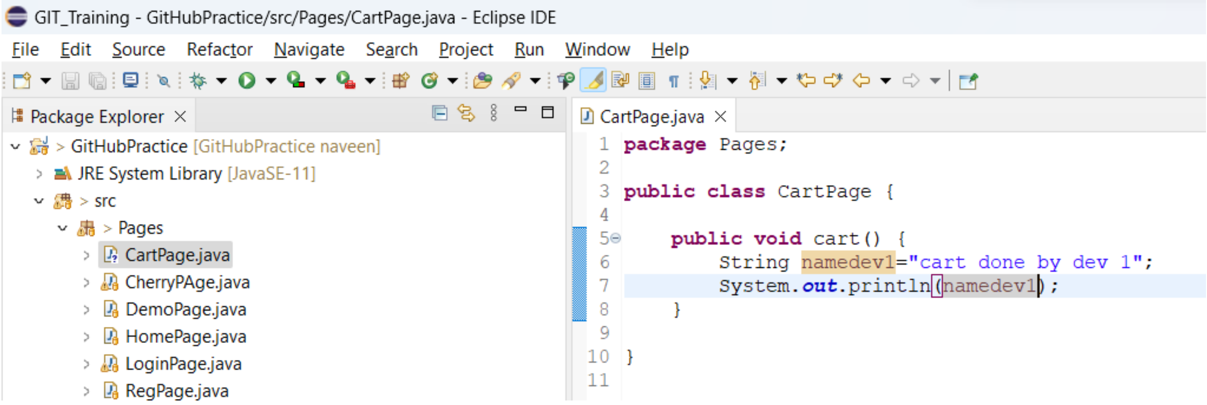


Code added-


GIT_Training - GitHubPractice/src/Pages/CartPage.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
GitHubPractice [GitHubPractice naveen]
JRE System Library [JavaSE-11]
src
Pages
CartPage.java
CherryPAge.java
DemoPage.java
HomePage.java
LoginPage.java
RegPage.java
CartPage.java

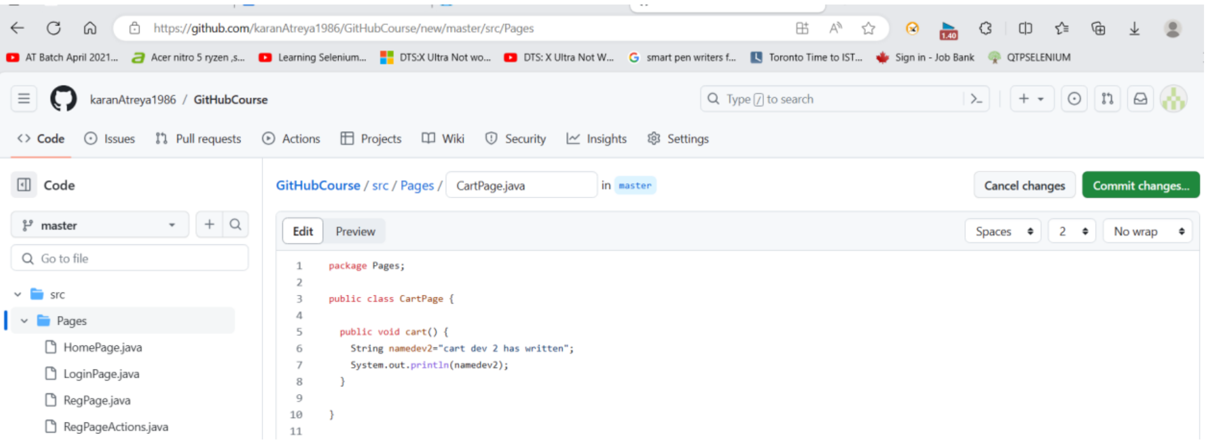
```java
package Pages;

public class CartPage {

    public void cart() {
        String name="cart";
        System.out.println(name);
    }

}
```

Code updated-


GIT_Training - GitHubPractice/src/Pages/CartPage.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
GitHubPractice [GitHubPractice naveen]
JRE System Library [JavaSE-11]
src
Pages
CartPage.java
CherryPAge.java
DemoPage.java
HomePage.java
LoginPage.java
RegPage.java
CartPage.java

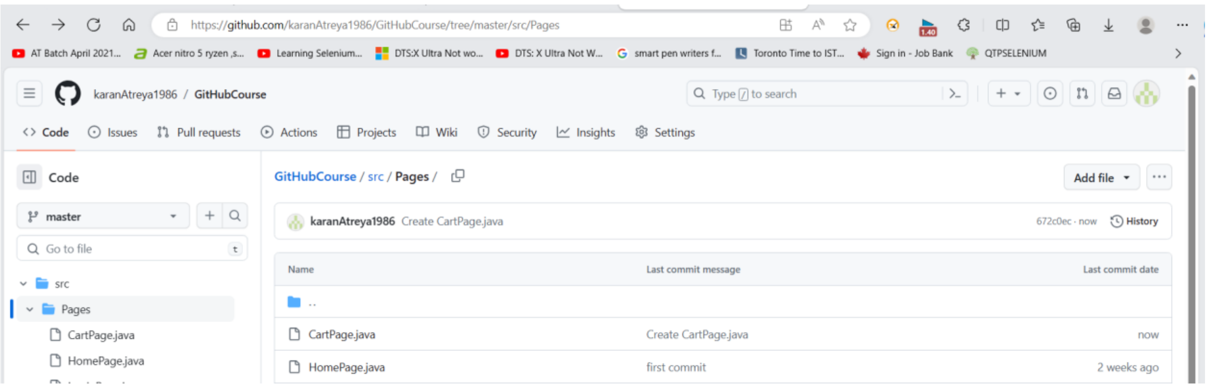
```java
package Pages;

public class CartPage {

    public void cart() {
        String namedev1="cart done by dev 1";
        System.out.println(namedev1);
    }

}
```

Committing same set of code in remote to simulate the real time-



```java
package Pages;

public class CartPage {

  public void cart() {
    String namedev2="cart dev 2 has written";
    System.out.println(namedev2);
  }

}
```

Cart page is present-

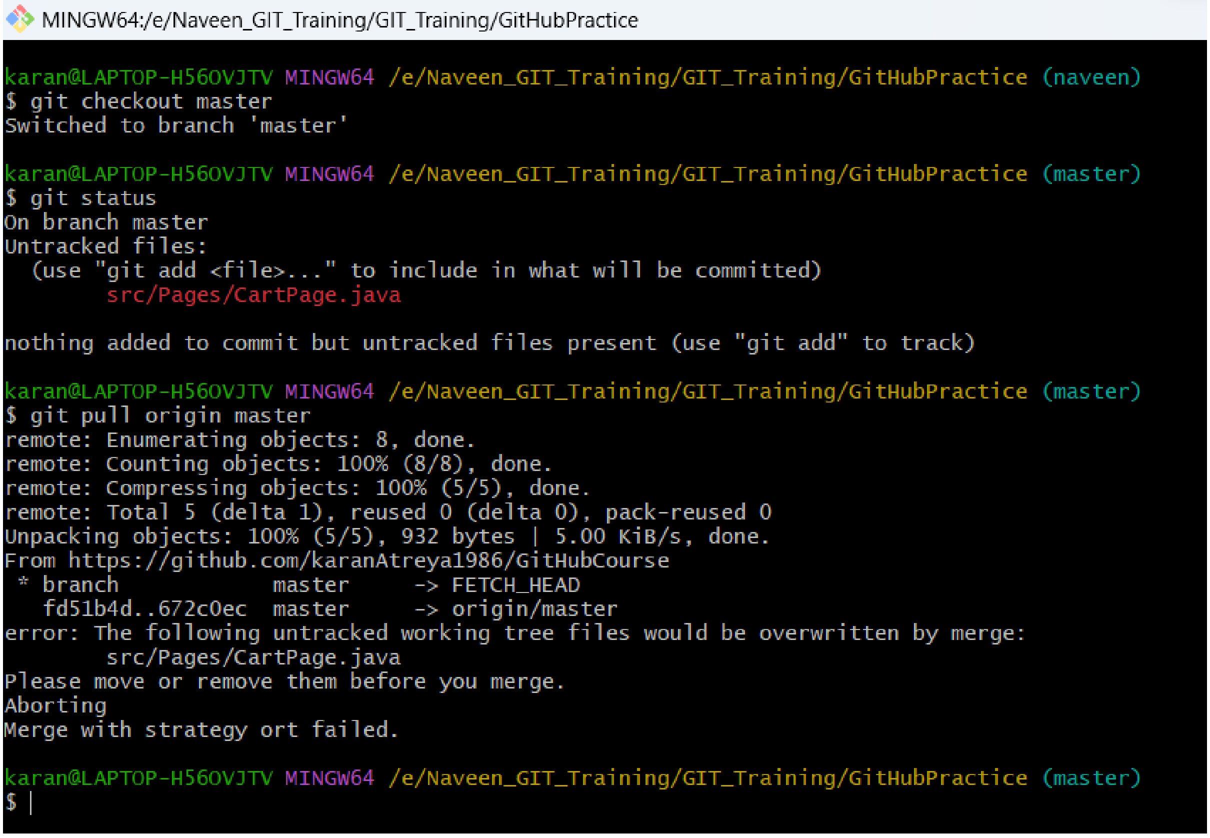


GitHubCourse / src / Pages /

karanAtreya1986 Create CartPage.java 672c0ec · now

Name	Last commit message	Last commit date
..		
CartPage.java	Create CartPage.java	now
HomePage.java	first commit	2 weeks ago

When you try to take pull we get error-

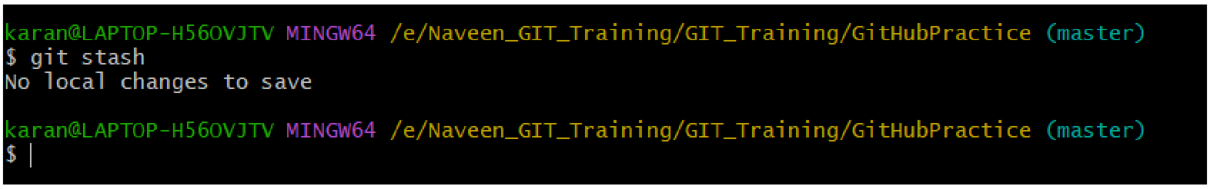


MINGW64:/e/Naveen_GIT_Training/GIT_Training/GitHubPractice

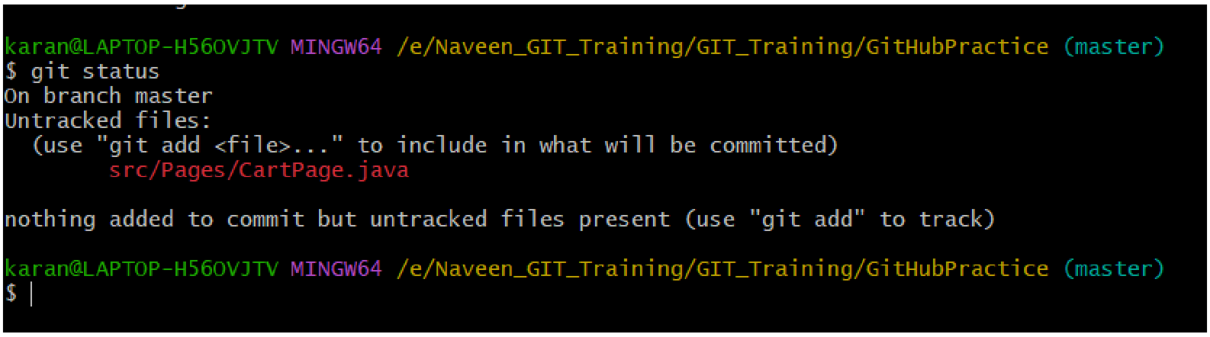
```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (naveen)
$ git checkout master
Switched to branch 'master'

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        src/Pages/CartPage.java

nothing added to commit but untracked files present (use "git add" to track)

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git pull origin master
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), 932 bytes | 5.00 KiB/s, done.
From https://github.com/karanAtreya1986/GitHubCourse
 * branch            master     -> FETCH_HEAD
   fd51b4d..672c0ec  master     -> origin/master
error: The following untracked working tree files would be overwritten by merge:
        src/Pages/CartPage.java
Please move or remove them before you merge.
Aborting
Merge with strategy ort failed.

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

Now we need to either remove the local copy of the file or commit the code or stash the code or delete the file from workspace. Till we don’t this pull will be aborted.

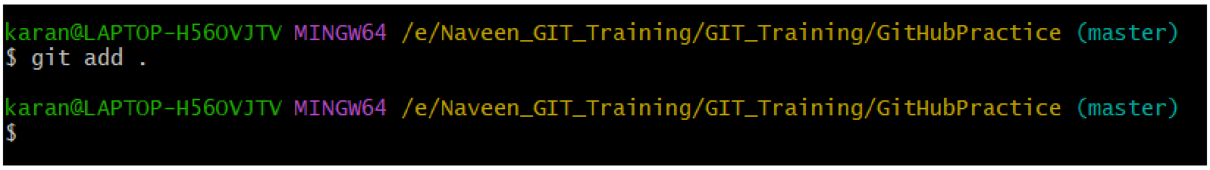
For stash, the code has to be first added using git add, else nothing will be saved as seen below-



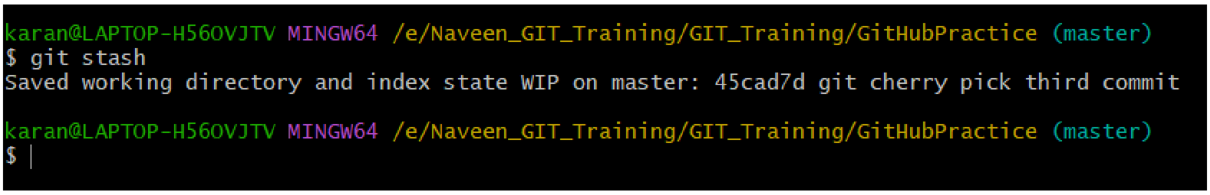
```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git stash
No local changes to save

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```



```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        src/Pages/CartPage.java

nothing added to commit but untracked files present (use "git add" to track)

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

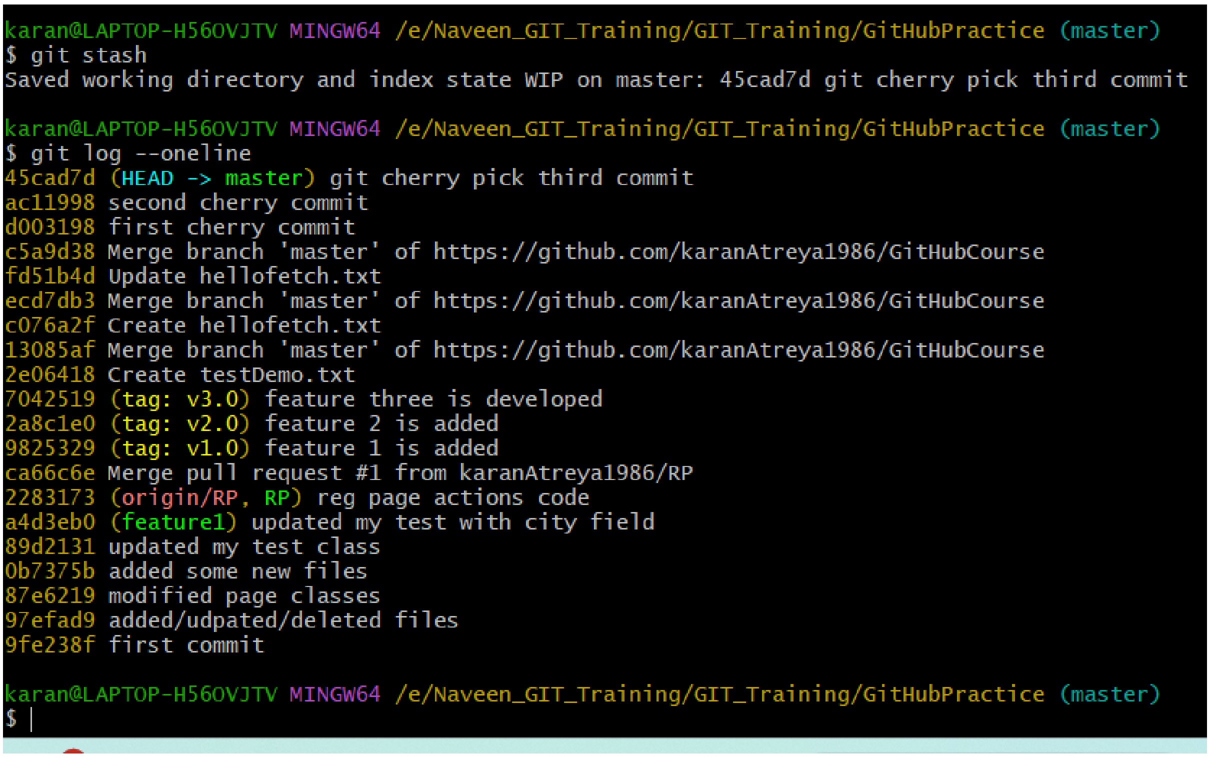


```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git add .

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$
```



```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git stash
Saved working directory and index state WIP on master: 45cad7d git cherry pick third commit

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

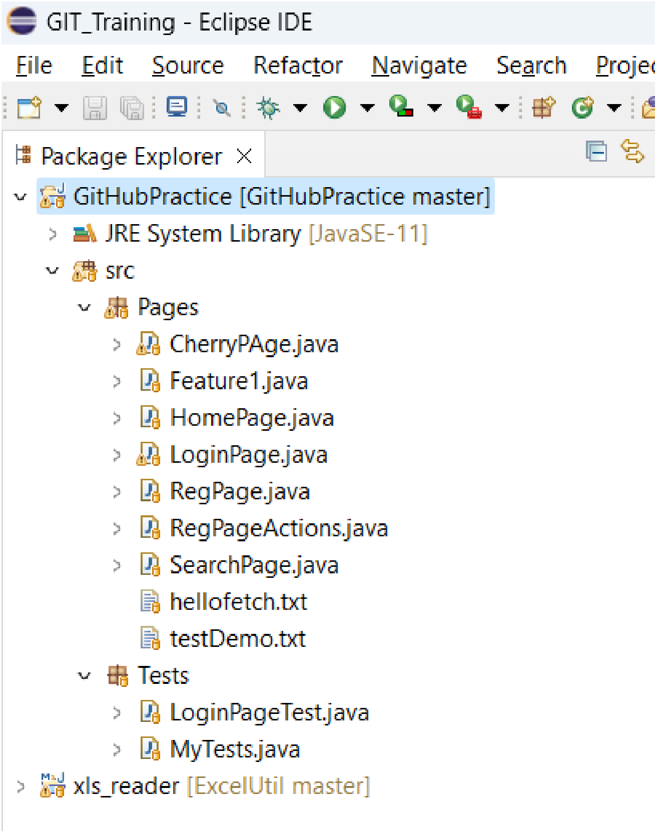
Last commit also moved to stash-



```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git stash
Saved working directory and index state WIP on master: 45cad7d git cherry pick third commit

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git log --oneline
45cad7d (HEAD -> master) git cherry pick third commit
ac11998 second cherry commit
d003198 first cherry commit
c5a9d38 Merge branch 'master' of https://github.com/karanAtreya1986/GitHubCourse
fd51b4d Update hellofetch.txt
ecd7db3 Merge branch 'master' of https://github.com/karanAtreya1986/GitHubCourse
c076a2f Create hellofetch.txt
13085af Merge branch 'master' of https://github.com/karanAtreya1986/GitHubCourse
2e06418 Create testDemo.txt
7042519 (tag: v3.0) feature three is developed
2a8c1e0 (tag: v2.0) feature 2 is added
9825329 (tag: v1.0) feature 1 is added
ca66c6e Merge pull request #1 from karanAtreya1986/RP
2283173 (origin/RP, RP) reg page actions code
a4d3eb0 (feature1) updated my test with city field
89d2131 updated my test class
0b7375b added some new files
87e6219 modified page classes
97efad9 added/udpated/deleted files
9fe238f first commit

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

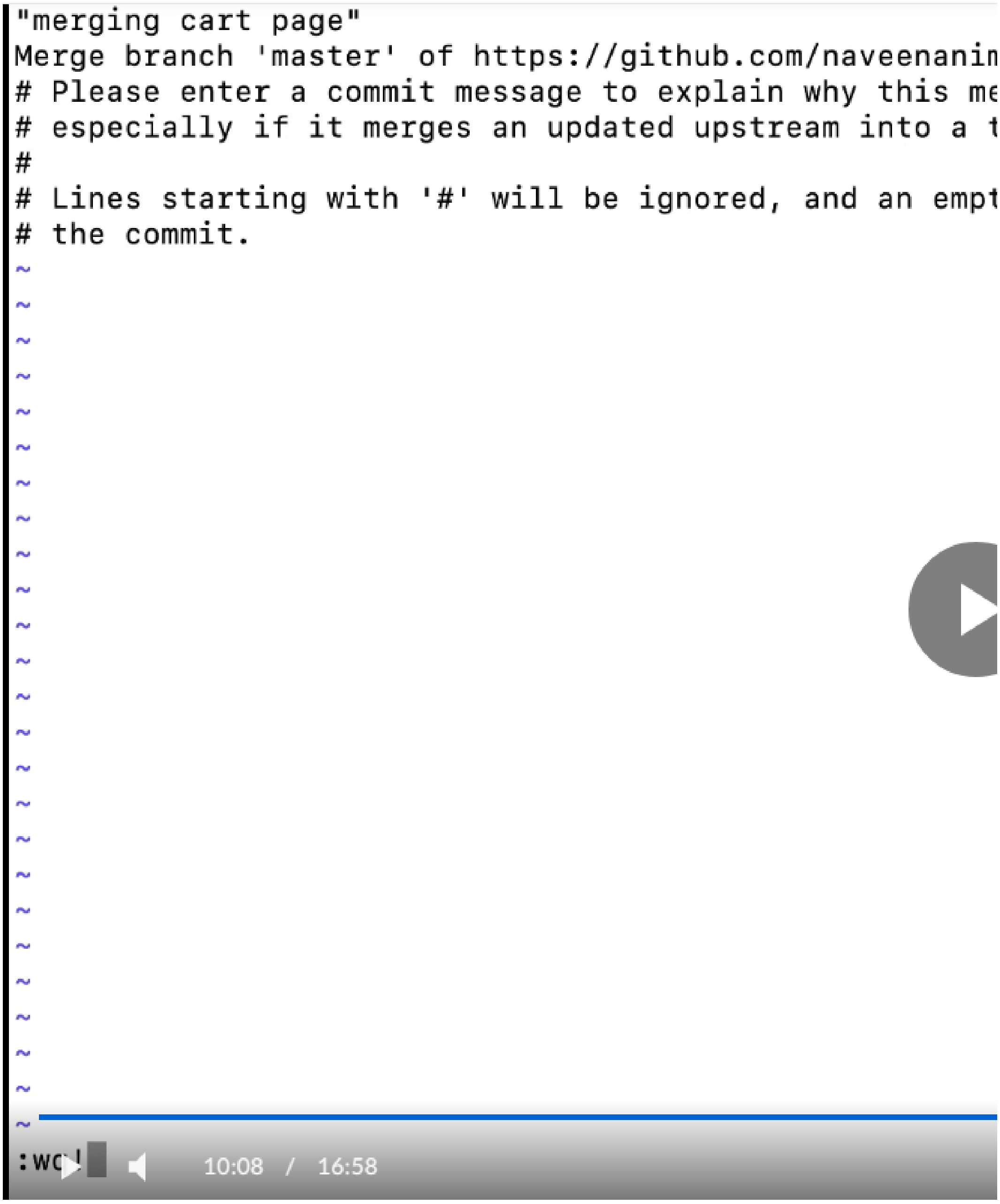
Cart page gone from ide-

5

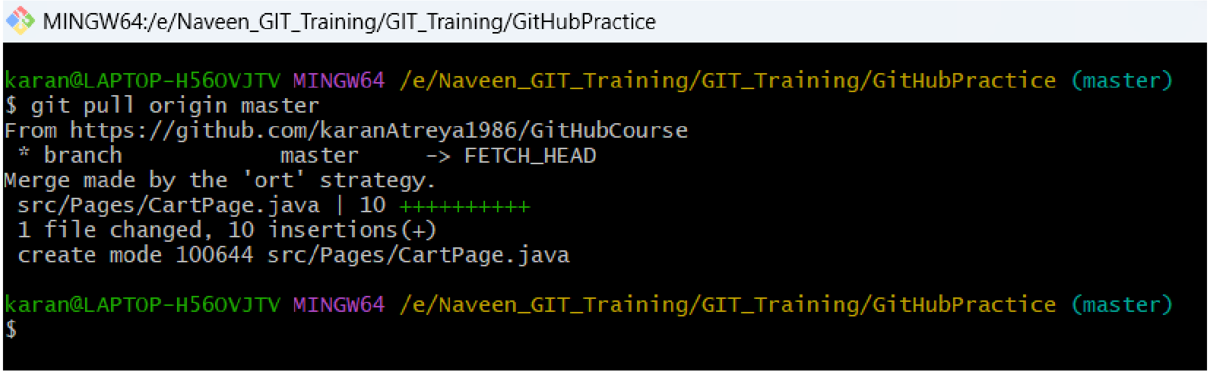


```
GitHubPractice [GitHubPractice master]
  JRE System Library [JavaSE-11]
  src
    Pages
      CherryPAge.java
      Feature1.java
      HomePage.java
      LoginPage.java
      RegPage.java
      RegPageActions.java
      SearchPage.java
      hellofetch.txt
      testDemo.txt
    Tests
      LoginPageTest.java
      MyTests.java
xls_reader [ExcelUtil master]
```

Give merge reason on naveens machine-

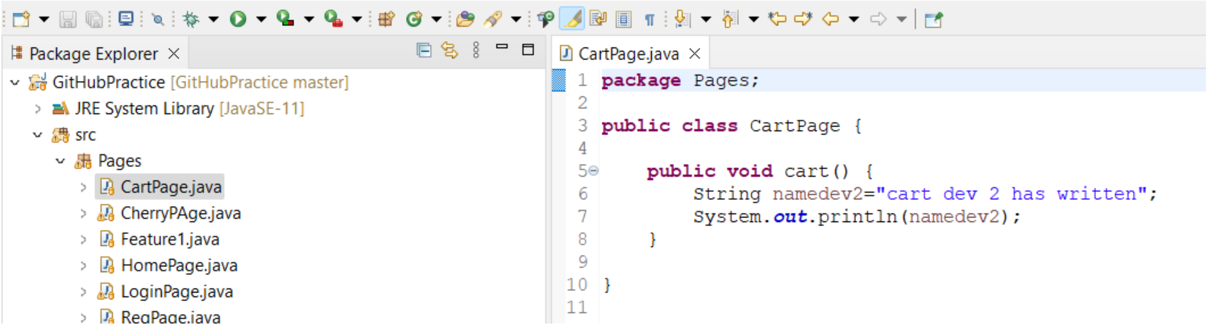


```
"merging cart page"
Merge branch 'master' of https://github.com/naveenanin
# Please enter a commit message to explain why this me
# especially if it merges an updated upstream into a t
#
# Lines starting with '#' will be ignored, and an empt
# the commit.
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq!
```



```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git pull origin master
From https://github.com/karanAtreya1986/GitHubCourse
 * branch            master     -> FETCH_HEAD
Merge made by the 'ort' strategy.
 src/Pages/CartPage.java | 10 ++++++++++
 1 file changed, 10 insertions(+)
 create mode 100644 src/Pages/CartPage.java

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$
```

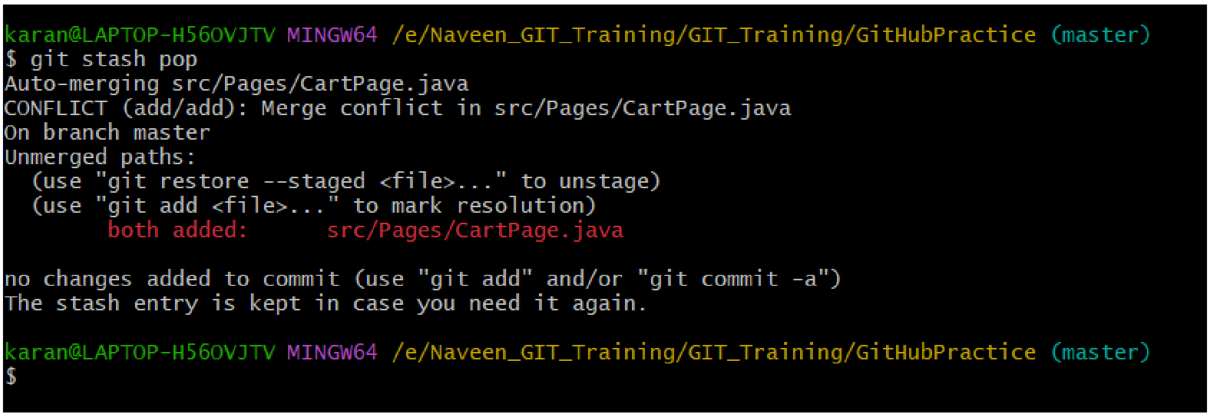
Refresh ide-



```
package Pages;

public class CartPage {

    public void cart() {
        String namedev2="cart dev 2 has written";
        System.out.println(namedev2);
    }

}
```

We got the dev2 change.

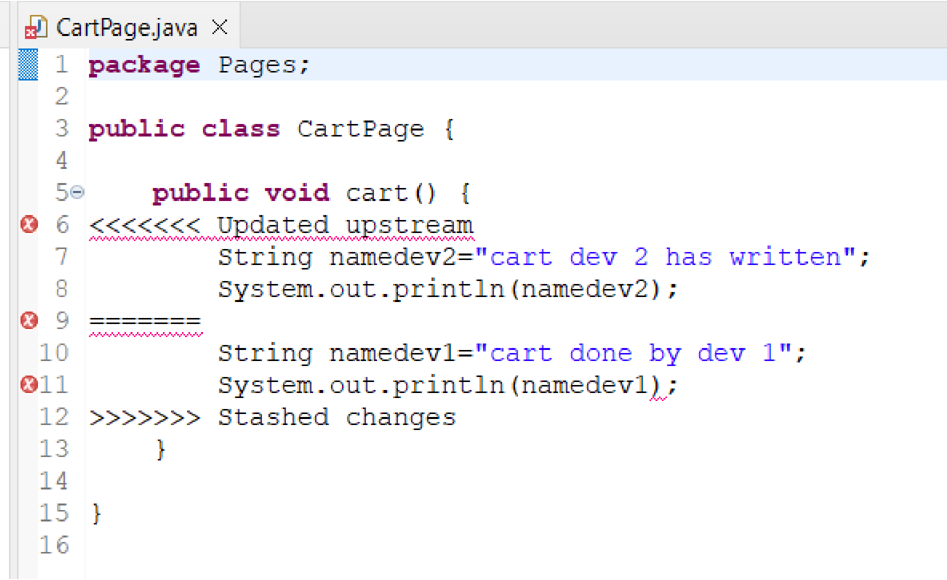
Now how to get our change back-



```
karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git stash pop
Auto-merging src/Pages/CartPage.java
CONFLICT (add/add): Merge conflict in src/Pages/CartPage.java
On branch master
Unmerged paths:
  (use "git restore --staged <file>..." to unstage)
  (use "git add <file>..." to mark resolution)
        both added:      src/Pages/CartPage.java

no changes added to commit (use "git add" and/or "git commit -a")
The stash entry is kept in case you need it again.

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$
```

We get merge conflict.

See how conflicts look in ide-



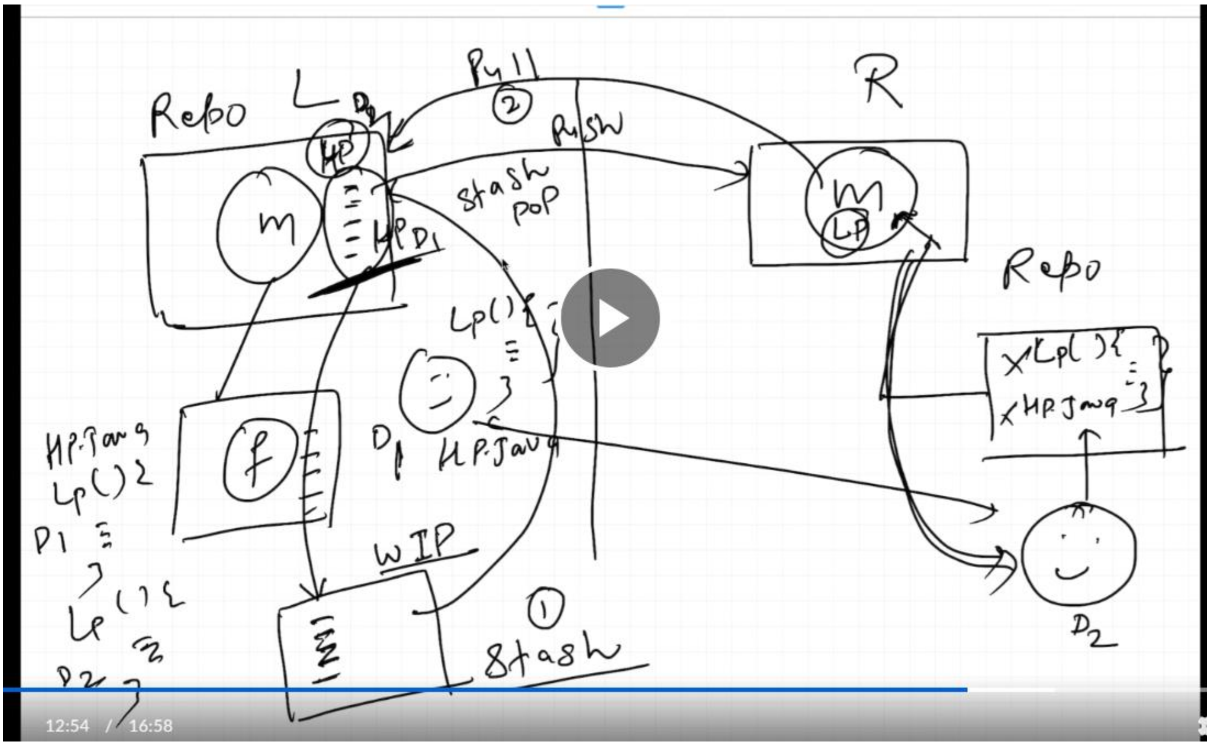
```
package Pages;

public class CartPage {

    public void cart() {
<<<<<<< Updated upstream
        String namedev2="cart dev 2 has written";
        System.out.println(namedev2);
=======
        String namedev1="cart done by dev 1";
        System.out.println(namedev1);
>>>>>>> Stashed changes
    }

}
```

Anything above “======” and upto “<<<<<<<<<” means code coming from remote end.

Updated upstream – its coming from remote end.

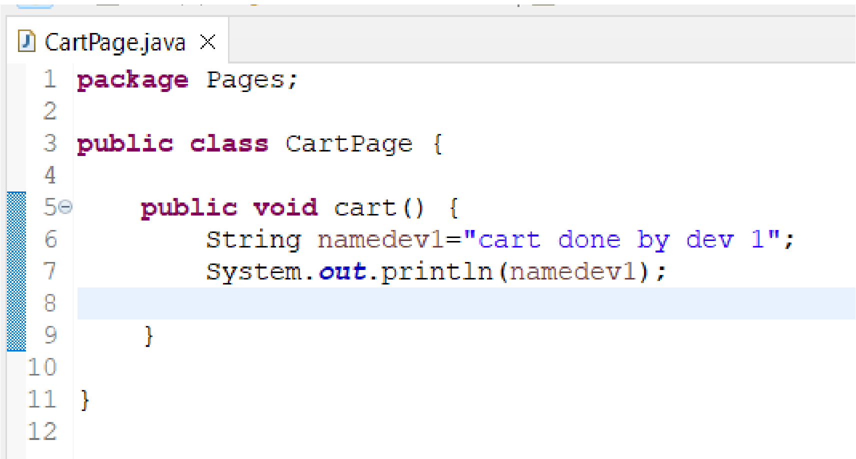
Stashed changes – means our local code is stashed.

Below "=========" and upto ">>>>>>>>>>" is our code.

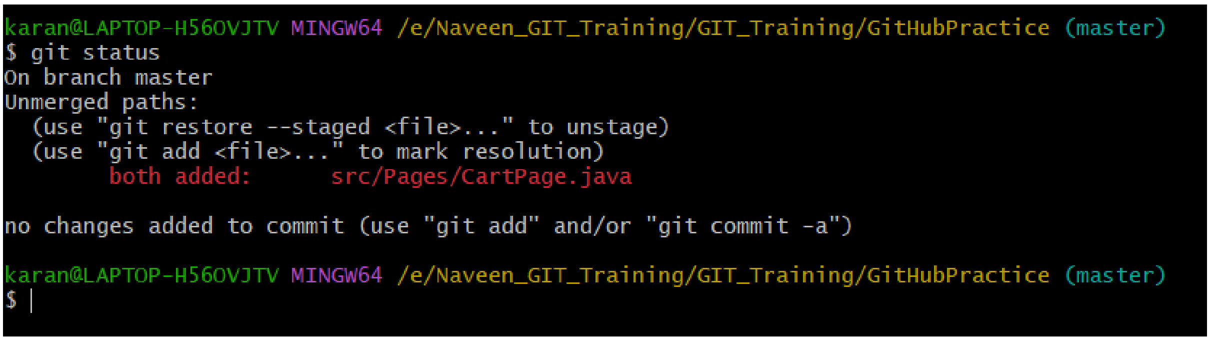


Now if suppose d1 code has to remain. D1 code has to be pushed to remote master.

D2 has to take pull from remote master. D2 may get conflict. D2 will delete his or her code and accept d1 code.

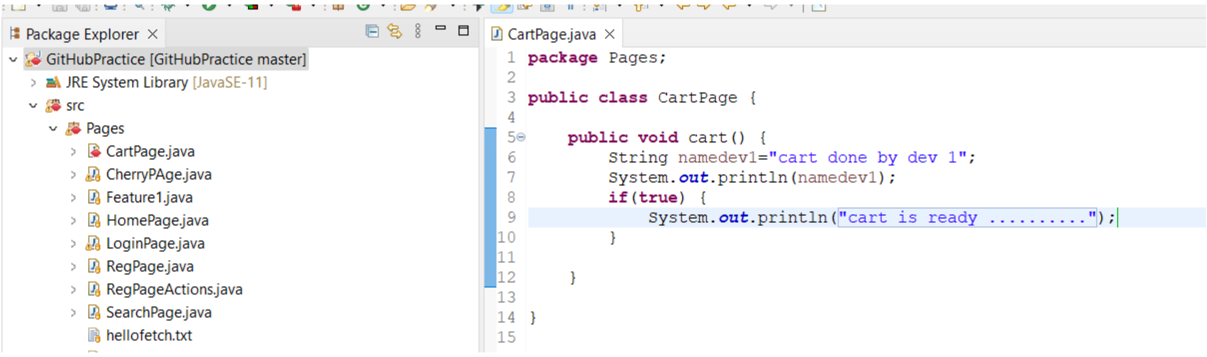


```
package Pages;

public class CartPage {

    public void cart() {
        String namedev1="cart done by dev 1";
        System.out.println(namedev1);

    }

}

```

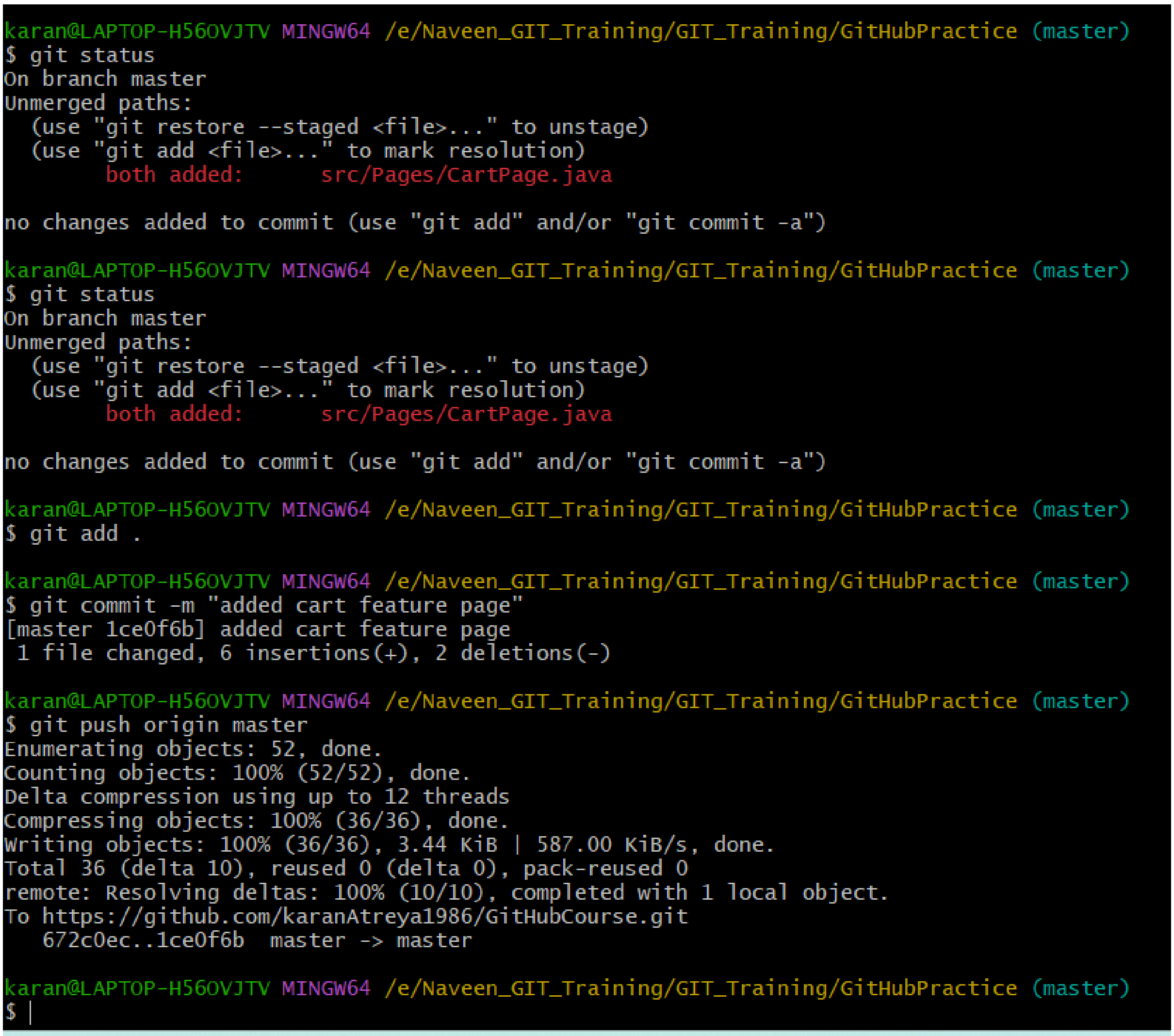


```
karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git status
On branch master
Unmerged paths:
  (use "git restore --staged <file>..." to unstage)
  (use "git add <file>..." to mark resolution)
        both added:      src/Pages/CartPage.java

no changes added to commit (use "git add" and/or "git commit -a")

karan@LAPTOP-H56OVJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

Added more code to cart page-

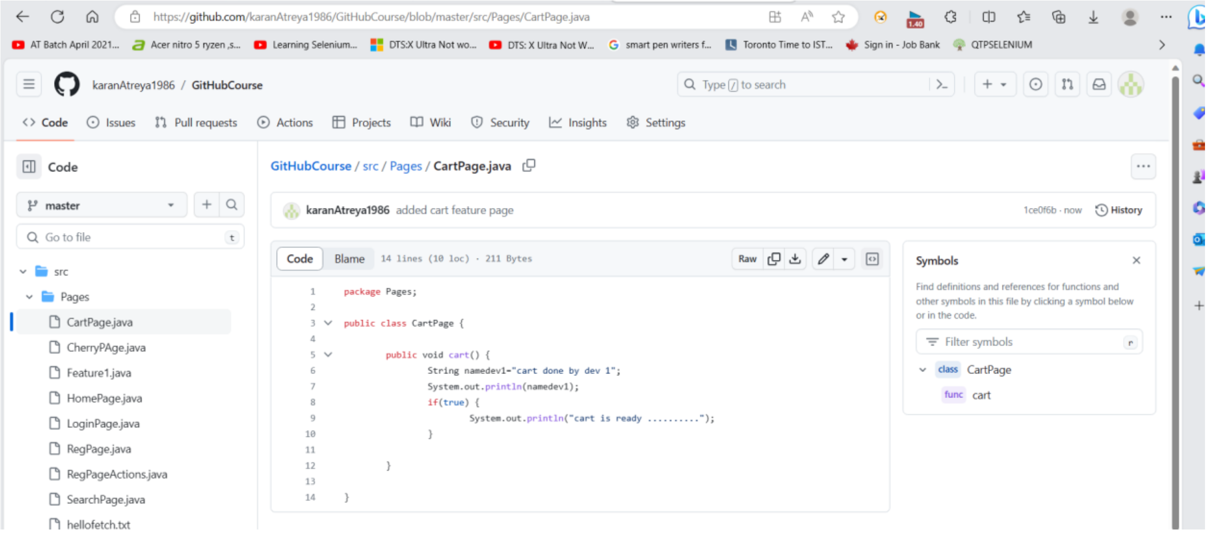


```java
package Pages;

public class CartPage {

    public void cart() {
        String namedev1="cart done by dev 1";
        System.out.println(namedev1);
        if(true) {
            System.out.println("cart is ready ..........");
        }

    }

}
```



```
karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git status
On branch master
Unmerged paths:
  (use "git restore --staged <file>..." to unstage)
  (use "git add <file>..." to mark resolution)
        both added:      src/Pages/CartPage.java

no changes added to commit (use "git add" and/or "git commit -a")

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git status
On branch master
Unmerged paths:
  (use "git restore --staged <file>..." to unstage)
  (use "git add <file>..." to mark resolution)
        both added:      src/Pages/CartPage.java

no changes added to commit (use "git add" and/or "git commit -a")

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git add .

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git commit -m "added cart feature page"
[master 1ce0f6b] added cart feature page
 1 file changed, 6 insertions(+), 2 deletions(-)

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git push origin master
Enumerating objects: 52, done.
Counting objects: 100% (52/52), done.
Delta compression using up to 12 threads
Compressing objects: 100% (36/36), done.
Writing objects: 100% (36/36), 3.44 KiB | 587.00 KiB/s, done.
Total 36 (delta 10), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (10/10), completed with 1 local object.
To https://github.com/karanAtreya1986/GitHubCourse.git
   672c0ec..1ce0f6b  master -> master

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

Code updated in gitlab-



https://github.com/karanAtreya1986/GitHubCourse/blob/master/src/Pages/CartPage.java

GitHubCourse / src / Pages / CartPage.java

karanAtreya1986 added cart feature page 1ce0f6b · now

14 lines (10 loc) · 211 Bytes

```java
package Pages;

public class CartPage {

        public void cart() {
                String namedev1="cart done by dev 1";
                System.out.println(namedev1);
                if(true) {
                        System.out.println("cart is ready ..........");
                }

        }

}
```

Git stash –

We can stash from index/stage/commit areas.

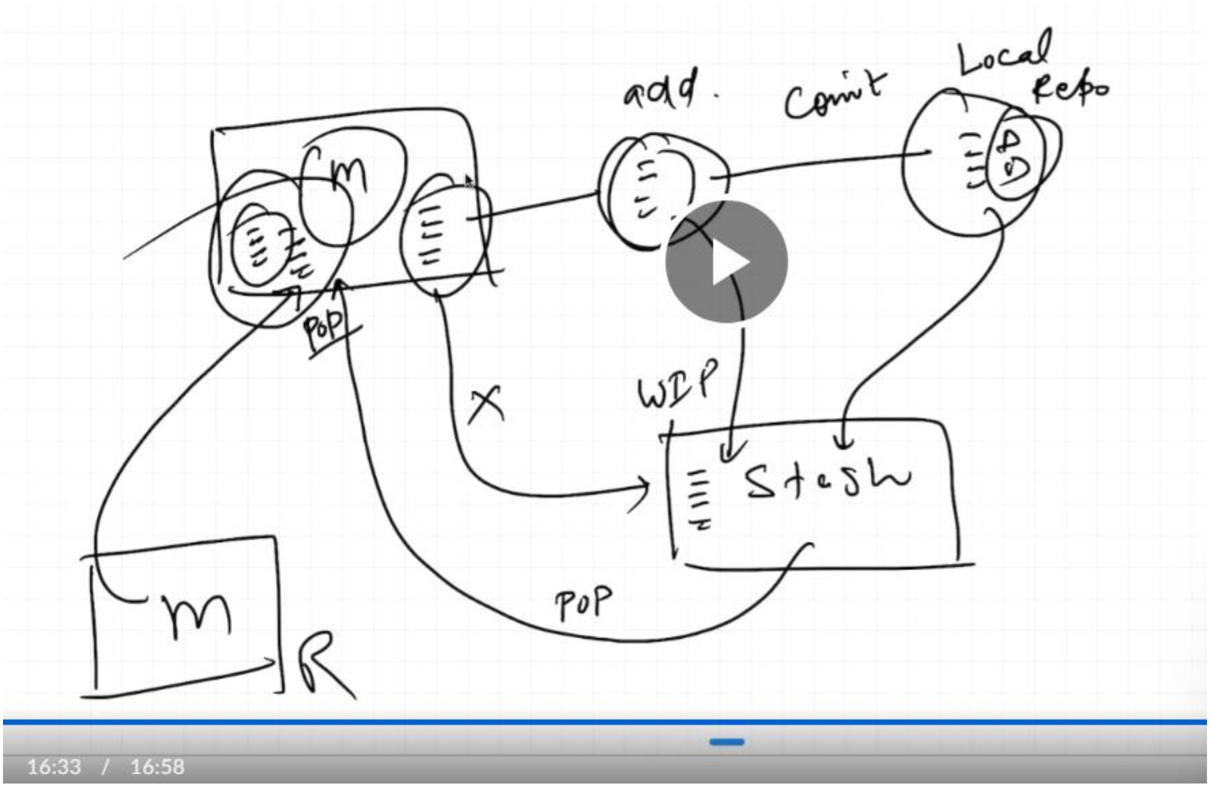
For stashing, the code has to be at least added using “git add”.

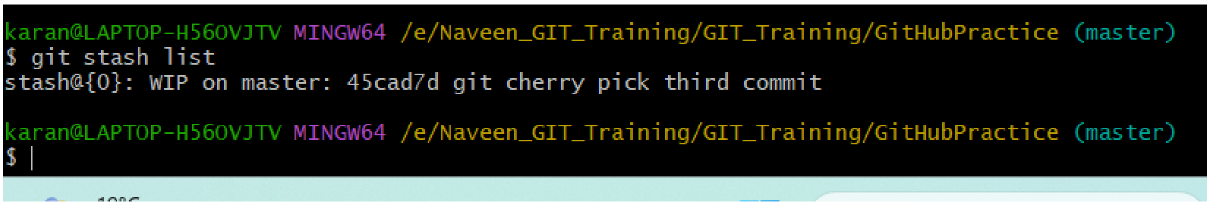
Git committed code can also come in stash.

So, when we stash all the code which is added using “git add” and “git commit” will be added to stash memory.

Previous commits will also go into the stash memory when we do “git stash”.

Git stash won’t take codes from workspace directly. Code has to be either added or committed to local repo for git stash to work.





```
karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ git stash list
stash@{0}: WIP on master: 45cad7d git cherry pick third commit

karan@LAPTOP-H560VJTV MINGW64 /e/Naveen_GIT_Training/GIT_Training/GitHubPractice (master)
$ |
```

{0} is like an array because stash memory might be saving in array format.

0th entry has this commit which was stashed.