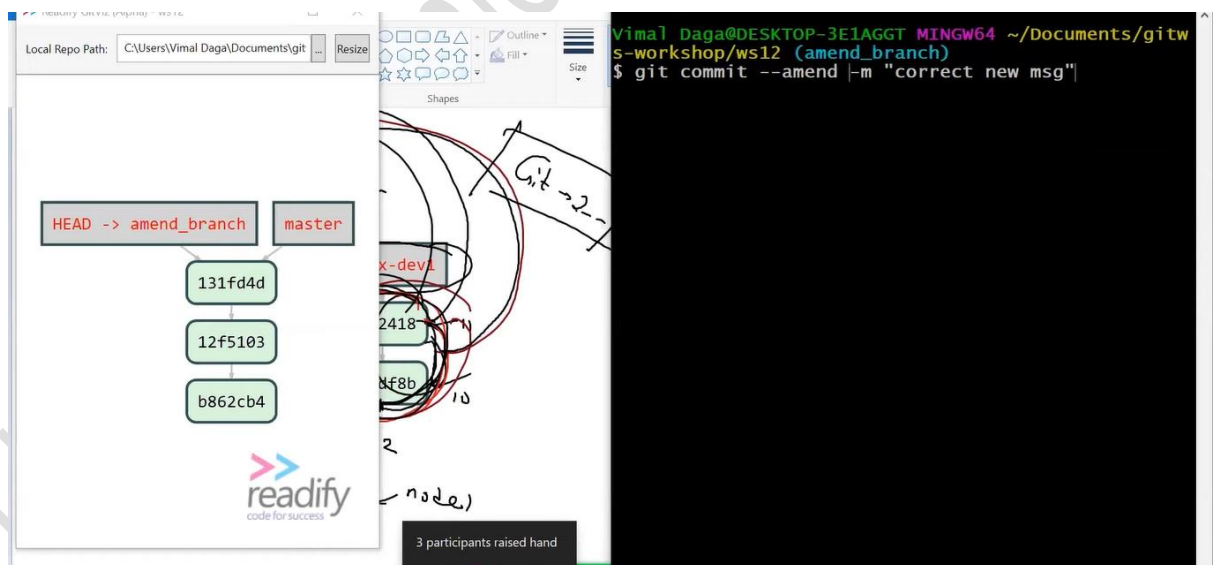




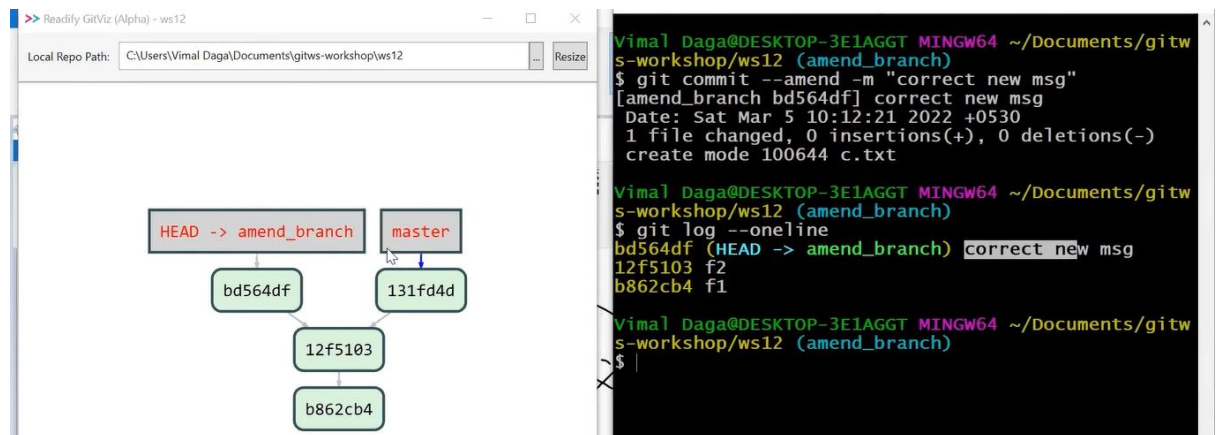
## Github Session 09

### Summary 05-03-2022

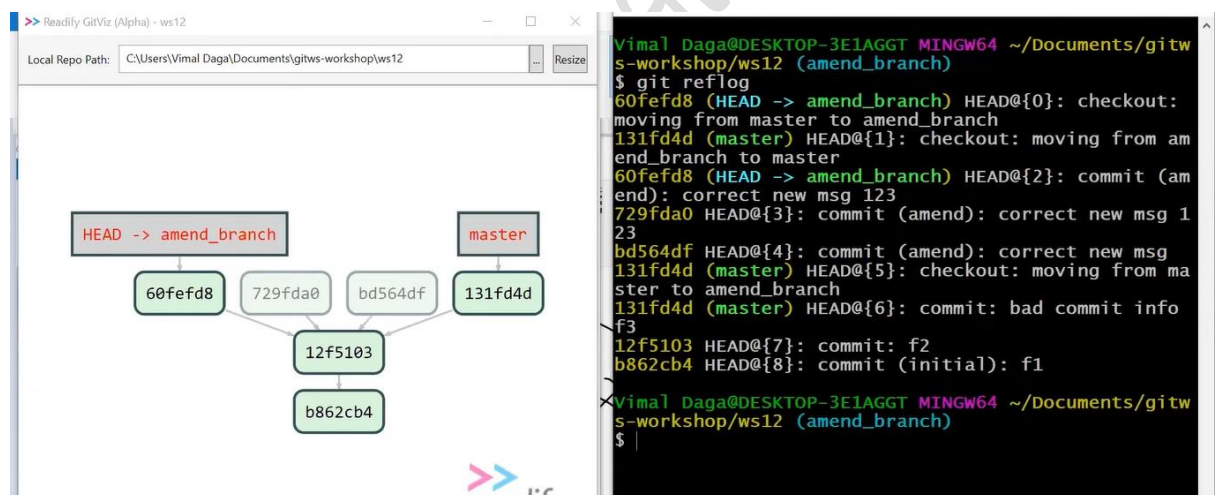
- If we want to go in the history or wants to change something in the history of the commit then we use Amend.
- Before making any changes in the commit, you should create a new branch for it first.
- Suppose the requirement is to change the comment of any commit then use command ***git commit --amend -m "new message"***



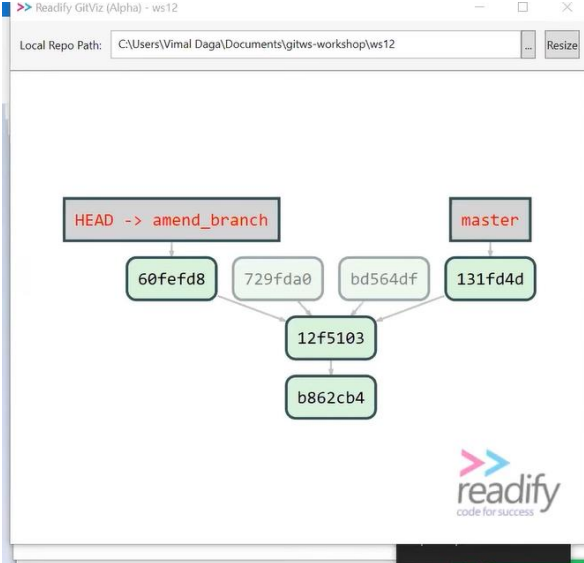
- What exactly amend does is, it creates a one more new commit area and copy the data from the previous commit with some changes we made in it.



- The new commit which has new changes in it is known as the **Reachable** commit and the previous or the old commits are known as the **orphaned** commits.
- To see the entire history of the commits either they are reachable or the orphaned commits we use the command **git reflog**



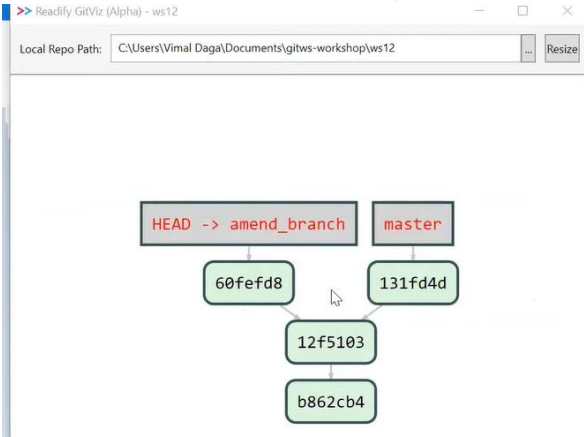
- We can still move the HEAD to the orphaned commits using the **git checkout <commit id>**
- To remove the unwanted or the expired commits, first we put all those in a garbage collector using the command **git reflog expire --expire-unreachable=now --all**



```

Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$ git reflog expire
Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$ git reflog expire --expire-unreachable=now
Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$ git reflog expire --expire-unreachable=now --all
Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$ git reflog
60fef88 (HEAD -> amend_branch) HEAD@{0}: checkout:
moving from master to amend_branch
131fd4d (master) HEAD@{1}: checkout: moving from am
end_branch to master
131fd4d (master) HEAD@{2}: checkout: moving from ma
ster to amend_branch
131fd4d (master) HEAD@{3}: commit: bad commit info
f3
12f5103 HEAD@{4}: commit: f2
b862cb4 HEAD@{5}: commit (initial): f1
Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$
  
```

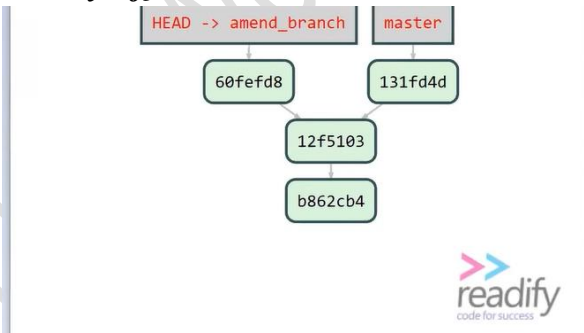
- Once the unwanted commits are collected in the garbage collector, we will now delete them using the command ***git gc --prune=now***



```

Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$ git gc --prune=now
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (8/8), done.
Total 8 (delta 2), reused 0 (delta 0), pack-reused
0
Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$
  
```

- To see the history we can use the ***git reflog HEAD@{time u want to see history of}***



```

Vimal Daga@DESKTOP-3E1AGGT MINGW64 ~/Documents/gitw
s-workshop/ws12 (amend_branch)
$ git reflog HEAD@{2.minutes.ago}
60fef88 (HEAD -> amend_branch) HEAD@{Sat Mar 5 10:2
4:31 2022 +0530}: checkout: moving from master to a
mend_branch
131fd4d (master) HEAD@{Sat Mar 5 10:22:59 2022 +053
0}: checkout: moving from amend_branch to master
131fd4d (master) HEAD@{Sat Mar 5 10:15:30 2022 +053
0}: checkout: moving from master to amend_branch
131fd4d (master) HEAD@{Sat Mar 5 10:12:21 2022 +053
0}: commit: bad commit info f3
12f5103 HEAD@{Sat Mar 5 10:11:21 2022 +0530}: commi
t: f2
b862cb4 HEAD@{Sat Mar 5 10:10:29 2022 +0530}: commi
t (initial): f1
  
```