1. Version Control System

2. Types of Version Control System

3. Version Control System Tools

4. What is Git?

5. Git Features

6. Git Workflow

7. Parallel Development

8. Hands-On

**Without Version Control System:**

1. No collaboration between the team. So no records are maintained.
2. Once saved, all the changes are permanent and cannot be reverted.
3. Downtime may occur due to faulty updates, since there is no back up maintained.

**Version Control System**

Version control system is the management of changes to documents, project programs, large web sites and other collection of information. These changes are usually termed as “versions”.

Version Control System Types:

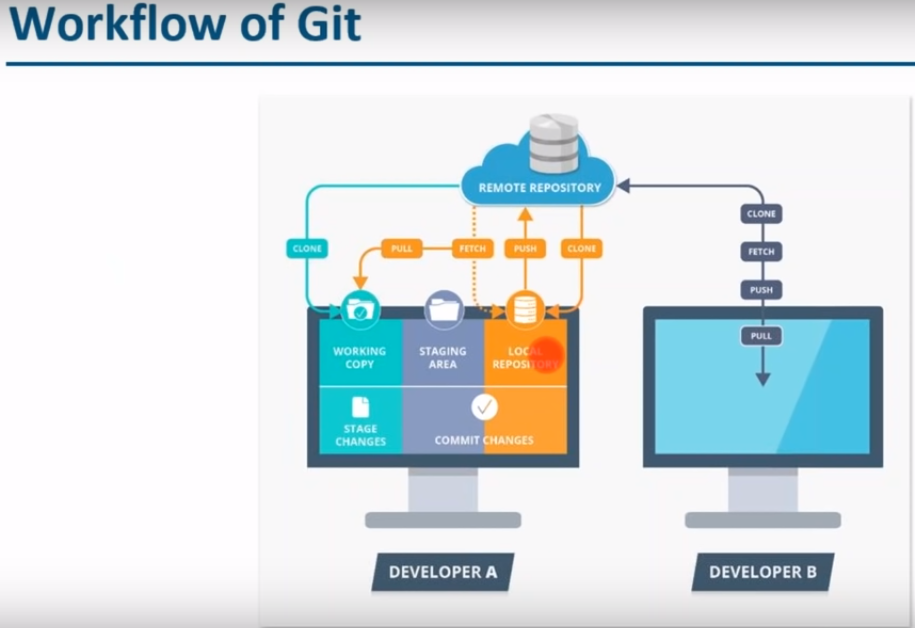
1. Local VCS
2. Centralized VCS
3. Distributed VCS

**Version Control System Tools:**

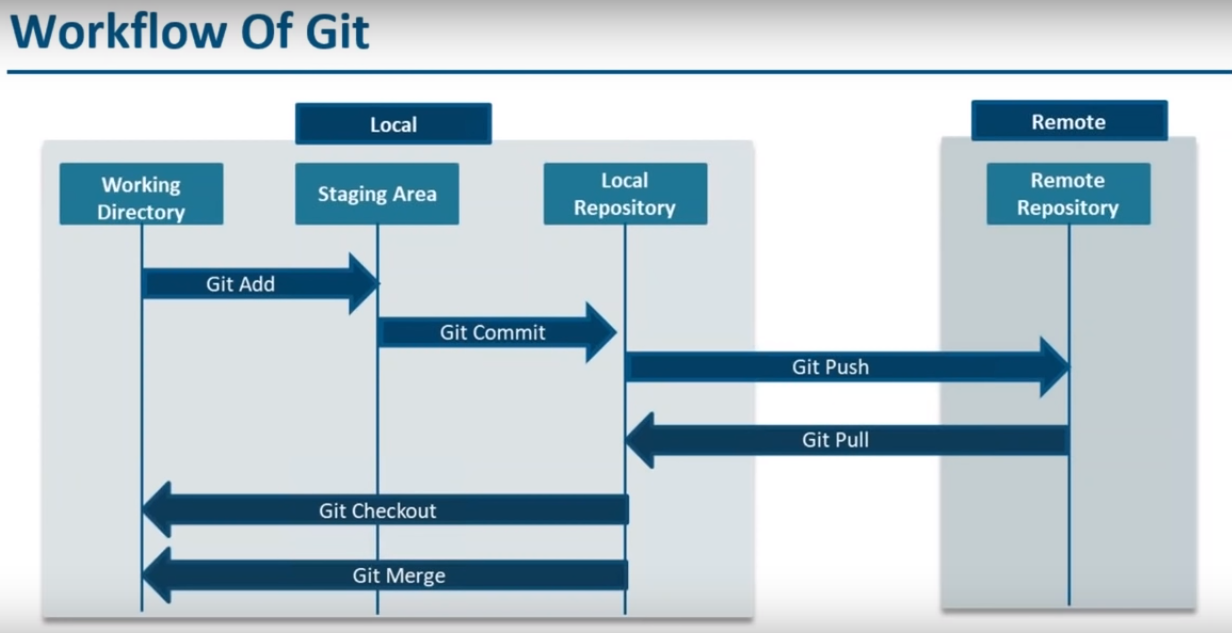
1. Git (Economical, non-linear dev of s/w, snapshots, distributed, integrity, Branching)
2. Apache Subversion
3. Concurrent Version Systems.
4. Team Foundation Server
5. Mercurial

**Git:** Git is an open source Distributed Version Control System (DVCS), which records changes, made to the files emphasis on speed, data integrity and distributed, non-linear workflows.

**Git Workflow Diagram:**



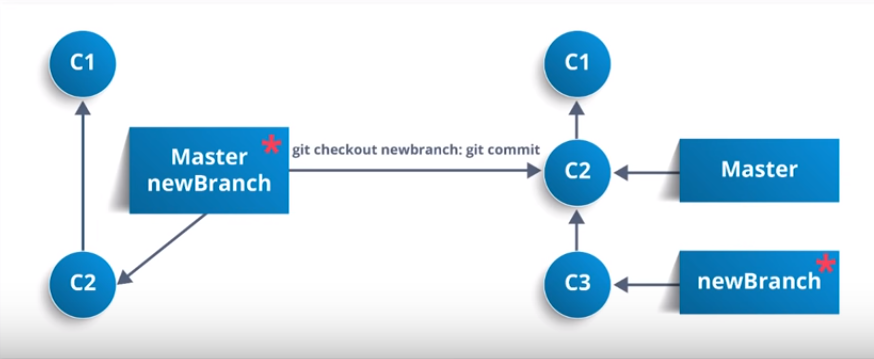
**Git Workflow Diagram:**



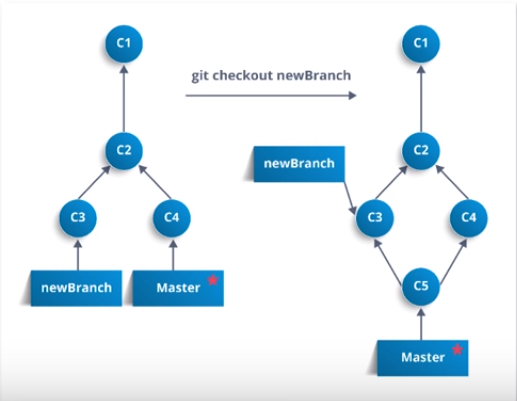
**Parallel Development:**

1. **Branching**
2. **Merging**
3. **Rebasing**

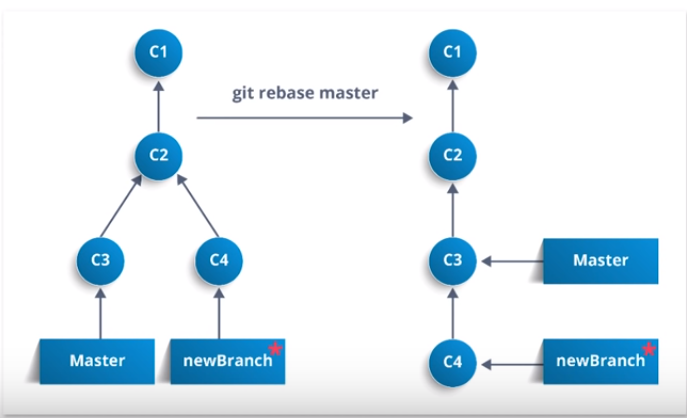
**Branching:** Branching is an integral part of any version control systems. Unlike other VC’s Git does not create a new copy of existing files for new branch. It points to snapshot of the changes you have made in the system.



**Merging:** Merging integrates the changes made in different branches into one single branch

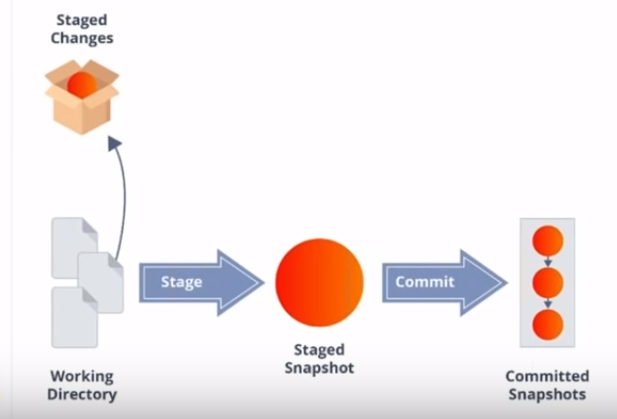


**Rebasing:** Used when changes made in one branch needs to be reflected in another branch

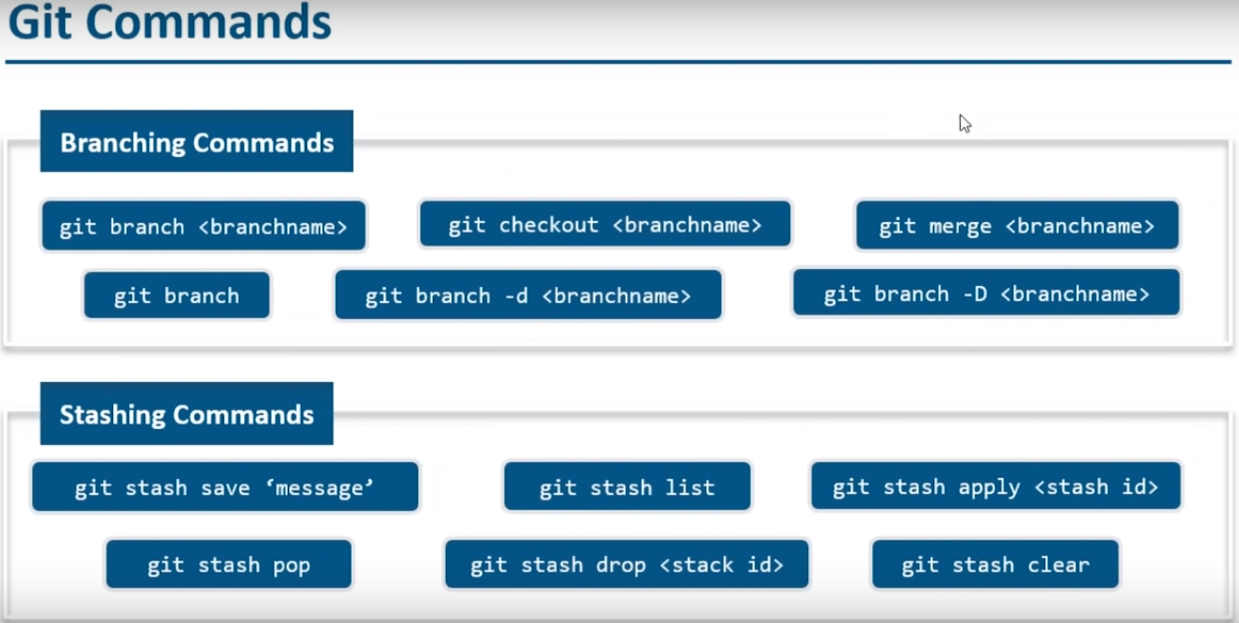


**Stashing:**

Used when uncommitted changes made in any branch is stored in temporary staging area and which will be committed to reflected in another branch.







**Git Hands On:**

PS00603038@HIS-PCS-2698957 MINGW64 ~

$ pwd

/c/Users/ps00603038

PS00603038@HIS-PCS-2698957 MINGW64 ~

$ ls -a

./ .sbt/ Downloads/ ntuser.dat.LOG2 Searches/

../ .scalaide/ Favorites/ NTUSER.DAT{016888bd-6c6f-11de-8d1d-001e0bcde3ec}.TM.blf SendTo@

.android/ .sts4/ IdeaProjects/ NTUSER.DAT{016888bd-6c6f-11de-8d1d-001e0bcde3ec}.TMContainer00000000000000000001.regtrans-ms 'Start Menu'@

.aws/ .tooling/ IntelGraphicsProfiles/ NTUSER.DAT{016888bd-6c6f-11de-8d1d-001e0bcde3ec}.TMContainer00000000000000000002.regtrans-ms target/

.bash\_history .viminfo Links/ ntuser.ini Templates@

.config/ AppData/ 'Local Settings'@ ntuser.pol Videos/

.dnx/ 'Application Data'@ Music/ Pictures/ workspace/

.gitconfig Contacts/ 'My Documents'@ PrintHood@

.idea-build/ Cookies@ NetHood@ project/

.IdeaIC2019.1/ Desktop/ NTUSER.DAT Recent@

.ivy2/ Documents/ ntuser.dat.LOG1 'Saved Games'/

PS00603038@HIS-PCS-2698957 MINGW64 ~

$ cd Desktop/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop (master)

$ ls

'~$ReportBase\_Dataprofiling\_Status\_Tracker\_050719.xlsx' desktop.ini 'Project Tracker Final.xlsx' Sites/

BlueJeans.lnk\* 'git error.JPG' 'Project Tracker.docx' 'SpringToolSuite4 - Shortcut.lnk'\*

CRST/ gitdemo/ putty\_1.exe\* Testrepo/

'CRST\_ BIDW\_Jul\_2019Timesheets.xlsx' gittutorial/ 'qlickview Error.JPG' 'TOS\_DQ-win-x86\_64 - Shortcut.lnk'\*

'CRST\_ BIDW\_Team\_2019Timesheets.xlsx' 'Microsoft SQL Server Management Studio.lnk'\* 'Qlik Sense Desktop.lnk'\*

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop (master)

$ cd gittutorial/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ ls -a

./ ../ sampleproject/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ cd sampleproject/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ ls

index.html sourcecode.hql

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ cd..

bash: cd..: command not found

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ cd ..

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ ls

sampleproject/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git init

Initialized empty Git repository in C:/Users/ps00603038/Desktop/gittutorial/.git/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git add .

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git commit -m "this is the first commit to the project"

[master (root-commit) a42ae5e] this is the first commit to the project

2 files changed, 22 insertions(+)

create mode 100644 sampleproject/index.html

create mode 100644 sampleproject/sourcecode.hql

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git status

On branch master

nothing to commit, working tree clean

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git log

commit a42ae5ece0c30a1e5f54ed8806a48d9b78e5ec42 (HEAD -> master)

Author: Srikanth <parshasrikanth14@gmail.com>

Date: Thu Sep 5 11:42:05 2019 +0530

this is the first commit to the project

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git branch feature1

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git branch

feature1

\* master

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (master)

$ git checkout feature1

Switched to branch 'feature1'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (feature1)

$ git branch

\* feature1

master

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (feature1)

$ git log

commit a42ae5ece0c30a1e5f54ed8806a48d9b78e5ec42 (HEAD -> feature1, master)

Author: Srikanth <parshasrikanth14@gmail.com>

Date: Thu Sep 5 11:42:05 2019 +0530

this is the first commit to the project

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (feature1)

$ pwd

/c/Users/ps00603038/Desktop/gittutorial

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial (feature1)

$ cd sampleproject/

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ gedit index.html

bash: gedit: command not found

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git diff

**diff --git a/sampleproject/index.html b/sampleproject/index.html**

**index 6b4e410..4fdea14 100644**

**--- a/sampleproject/index.html**

**+++ b/sampleproject/index.html**

@@ -1,6 +1,9 @@

<h1>Hello Wold<h1>

+ <!-- this is really very impartant file -->

+<h1>Hello AWS development environment<h1>

+

<h1>Hello Scala<h1>

<h1>Hello Spark<h1>

<h1>Hello Hive<h1>

<h1>Hello Sqoop<h1>

-<h1>Hello BigData<h1>

\ No newline at end of file

+<h1>Hello BigData<h1>

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git add index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git commit -m "A comment was added and also new headers are added"

[feature1 d113c45] A comment was added and also new headers are added

1 file changed, 4 insertions(+), 1 deletion(-)

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git checkout master

Switched to branch 'master'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git merge feature1

error: Your local changes to the following files would be overwritten by merge:

sampleproject/index.html

Please commit your changes or stash them before you merge.

Aborting

Updating a42ae5e..d113c45

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ 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: index.html

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

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git add index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git commit -m "this is my second commit"

[master 31b27d0] this is my second commit

1 file changed, 1 insertion(+), 1 deletion(-)

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git merge feature1

Auto-merging sampleproject/index.html

Merge made by the 'recursive' strategy.

sampleproject/index.html | 3 +++

1 file changed, 3 insertions(+)

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git remote add origin https://github.com/SrikanthParsha14/gitexample.git

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git log

commit 0e96122ed3e11bc88dc9f2420dc5f9b2e6c5531c (HEAD -> master)

Merge: 31b27d0 d113c45

Author: Srikanth <parshasrikanth14@gmail.com>

Date: Thu Sep 5 12:07:58 2019 +0530

Merge bransich 'feature1'

merge feature1 brach with master

commit 31b27d04b5d246994261bb9a6c7ba23447100fad

Author: Srikanth <parshasrikanth14@gmail.com>

Date: Thu Sep 5 12:07:47 2019 +0530

this is my second commit

commit d113c45f26b7d2899d8e0531ba58c53d61380535 (feature1)

Author: Srikanth <parshasrikanth14@gmail.com>

Date: Thu Sep 5 11:52:51 2019 +0530

A comment was added and also new headers are added

commit a42ae5ece0c30a1e5f54ed8806a48d9b78e5ec42

Author: Srikanth <parshasrikanth14@gmail.com>

Date: Thu Sep 5 11:42:05 2019 +0530

this is the first commit to the project

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git checkout -b feature2

Switched to a new branch 'feature2'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature2)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature2)

$ git diff

**diff --git a/sampleproject/index.html b/sampleproject/index.html**

**index 4fdea14..6e8dba8 100644**

**--- a/sampleproject/index.html**

**+++ b/sampleproject/index.html**

@@ -1,9 +1,8 @@

<h1>Hello Wold<h1>

- <!-- this is really very impartant file -->

+<!-- this is really very impartant file -->

<h1>Hello AWS development environment<h1>

-

-<h1>Hello Scala<h1>

+<h1>Continuous integrationa and continuous delivery<h1>

<h1>Hello Spark<h1>

<h1>Hello Hive<h1>

<h1>Hello Sqoop<h1>

-<h1>Hello BigData<h1>

+<h1>CI/CD<h1>

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature2)

$ git stash save 'feature2 changed'

Saved working directory and index state On feature2: feature2 changed

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature2)

$ git stash list

stash@{0}: On feature2: feature2 changed

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature2)

$ git checkout feature1

Switched to branch 'feature1'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git stash apply stash@{0}

error: Your local changes to the following files would be overwritten by merge:

sampleproject/index.html

Please commit your changes or stash them before you merge.

Aborting

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git add .

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git commit -m "index.html is saved without any changes'

> "

On branch feature1

nothing to commit, working tree clean

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git stash apply stash@{0}

On branch feature1

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: index.html

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

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git commit -m "CI/CD added and Continuous integration and continuous delivery is added"

On branch feature1

Changes not staged for commit:

modified: index.html

no changes added to commit

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature1)

$ git checkout -b feature3

Switched to a new branch 'feature3'

M sampleproject/index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git add index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git commit -m "RDBMS to HDFS and HDFS to RDBMS comment is added"

[feature3 76197c0] RDBMS to HDFS and HDFS to RDBMS comment is added

1 file changed, 4 insertions(+), 5 deletions(-)

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git checkout master

Switched to branch 'master'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ vi index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git add index.html

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git commit -m "added spark sql"

[master b26b229] added spark sql

1 file changed, 1 insertion(+), 1 deletion(-)

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ ls -a

./ ../ index.html sourcecode.hql

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ vi sourcecode.hql

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git commit -a -m "a comment was added"

[master 66f0b62] a comment was added

1 file changed, 1 insertion(+), 1 deletion(-)

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (master)

$ git checkout feature3

Switched to branch 'feature3'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git rebase master

First, rewinding head to replay your work on top of it...

Applying: RDBMS to HDFS and HDFS to RDBMS comment is added

Using index info to reconstruct a base tree...

M sampleproject/index.html

Falling back to patching base and 3-way merge...

Auto-merging sampleproject/index.html

CONFLICT (content): Merge conflict in sampleproject/index.html

error: Failed to merge in the changes.

hint: Use 'git am --show-current-patch' to see the failed patch

Patch failed at 0001 RDBMS to HDFS and HDFS to RDBMS comment is added

Resolve all conflicts manually, mark them as resolved with

"git add/rm <conflicted\_files>", then run "git rebase --continue".

You can instead skip this commit: run "git rebase --skip".

To abort and get back to the state before "git rebase", run "git rebase --abort".

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3|REBASE 1/1)

$ git rebase --abort

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git am --show-current-path

error: unknown option `show-current-path'

usage: git am [<options>] [(<mbox> | <Maildir>)...]

or: git am [<options>] (--continue | --skip | --abort)

-i, --interactive run interactively

-3, --3way allow fall back on 3way merging if needed

-q, --quiet be quiet

-s, --signoff add a Signed-off-by line to the commit message

-u, --utf8 recode into utf8 (default)

-k, --keep pass -k flag to git-mailinfo

--keep-non-patch pass -b flag to git-mailinfo

-m, --message-id pass -m flag to git-mailinfo

--keep-cr pass --keep-cr flag to git-mailsplit for mbox format

--no-keep-cr do not pass --keep-cr flag to git-mailsplit independent of am.keepcr

-c, --scissors strip everything before a scissors line

--whitespace <action>

pass it through git-apply

--ignore-space-change

pass it through git-apply

--ignore-whitespace pass it through git-apply

--directory <root> pass it through git-apply

--exclude <path> pass it through git-apply

--include <path> pass it through git-apply

-C <n> pass it through git-apply

-p <num> pass it through git-apply

--patch-format <format>

format the patch(es) are in

--reject pass it through git-apply

--resolvemsg ... override error message when patch failure occurs

--continue continue applying patches after resolving a conflict

-r, --resolved synonyms for --continue

--skip skip the current patch

--abort restore the original branch and abort the patching operation.

--quit abort the patching operation but keep HEAD where it is.

--show-current-patch show the patch being applied.

--committer-date-is-author-date

lie about committer date

--ignore-date use current timestamp for author date

--rerere-autoupdate update the index with reused conflict resolution if possible

-S, --gpg-sign[=<key-id>]

GPG-sign commits

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git push -u origin master

Logon failed, use ctrl+c to cancel basic credential prompt.

remote: Invalid username or password.

fatal: Authentication failed for 'https://github.com/SrikanthParsha14/gitexample.git/'

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)

$ git push -u origin master

Enumerating objects: 22, done.

Counting objects: 100% (22/22), done.

Delta compression using up to 4 threads

Compressing objects: 100% (17/17), done.

Writing objects: 100% (22/22), 2.09 KiB | 1.05 MiB/s, done.

Total 22 (delta 4), reused 0 (delta 0)

remote: Resolving deltas: 100% (4/4), done.

To https://github.com/SrikanthParsha14/gitexample.git

\* [new branch] master -> master

Branch 'master' set up to track remote branch 'master' from 'origin'.

PS00603038@HIS-PCS-2698957 MINGW64 ~/Desktop/gittutorial/sampleproject (feature3)