

GIT-1

```
MINGW64/c/Users/Lenovo
Lenovo@AvengX MINGW64 ~
$ git --version
git version 2.49.0.windows.1

Lenovo@AvengX MINGW64 ~
$ git config --global user.name "AvengX"

Lenovo@AvengX MINGW64 ~
$ git config --global user.name "ayushraj7654321@gmail.com"

Lenovo@AvengX MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=schannel
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=ayushraj7654321@gmail.com
```

```
Lenovo@AvengX MINGW64 ~
$ mkdir ~/GitDemo

Lenovo@AvengX MINGW64 ~
$ cd ~/GitDemo

Lenovo@AvengX MINGW64 ~/GitDemo
$ git init
Initialized empty Git repository in C:/Users/Lenovo/GitDemo/.git/

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ cat > welcome.txt <<'EOF'
Welcome to GitDemo
This repository was created for the Git hands-on lab.
EOF

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    welcome.txt

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

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ cat welcome.txt
Welcome to GitDemo
This repository was created for the Git hands-on lab.

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git add welcome.txt
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git commit -m "Add welcome.txt"
Author identity unknown

*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
```



```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git add welcome.txt
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git commit -m "Add welcome.txt"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'Lenovo@AvengX.(none)')

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ $ git config --global user.name "AvengX"
bash: $: command not found

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ $ git config --global user.name "AvengX"
bash: $: command not found

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git config --global user.name "AvengX"
git config --global user.email "ayushraj7654321@gmail.com"

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git add welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git commit -m "Add welcome.txt"
[master (root-commit) f39c932] Add welcome.txt
1 file changed, 2 insertions(+)
create mode 100644 welcome.txt

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ ls -al
total 25
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:16 ./
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:14 ../
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:13 .git/
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:16 GitDemo/
-rw-r--r-- 1 Lenovo 197121 73 Aug  9 09:09 welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ echo "welcome to the version control">>welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ ls -al
total 25
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:16 ./
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:14 ../
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:13 .git/
drwxr-xr-x 1 Lenovo 197121  0 Aug  9 09:16 GitDemo/
-rw-r--r-- 1 Lenovo 197121 104 Aug  9 09:17 welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ cat welcome.txt
Welcome to GitDemo
This repository was created for the Git hands-on lab.
welcome to the version control

Lenovo@AvengX MINGW64 ~/GitDemo (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:   welcome.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        GitDemo/

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

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git add welcome.txt
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git commit
hint: Waiting for your editor to close the file... notepad++ -multiInst -nosession -wait: line 1: notepad++: command not found
error: there was a problem with the editor 'notepad++ -multiInst -nosession -wait'
Please supply the message using either -m or -F option.

```

```
GitDemo/

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git pull origin master
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git push origin master
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git remote add origin <https://github.com/AvengX/GitDemo.git>
bash: syntax error near unexpected token 'newline'

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git remote add origin https://github.com/AvengX/GitDemo.git

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git branch -M main
git push -u origin main
```

GIT-2

```
MINGW64:/c/Users/Lenovo/GitDemo

Lenovo@AvengX MINGW64 ~
$ git config --global user.name "AvengX"
git config --global user.email "ayushraj7654321@gmail.com"

Lenovo@AvengX MINGW64 ~
$ git config --global core.editor "\"C:/Program Files/Notepad++/notepad++.exe\" -multiInst -nosession -wait"

Lenovo@AvengX MINGW64 ~
$ mkdir GitDemo
cd GitDemo
git init
mkdir: cannot create directory 'GitDemo': File exists
Reinitialized existing Git repository in C:/Users/Lenovo/GitDemo/.git/

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ *.log
bash: *.log: command not found

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ echo "test log" > debug.log

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ mkdir -p log

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ echo "inside log folder" > log/app.log

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add .gitignore
git commit -m "Add .gitignore to ignore logs"
fatal: pathspec '.gitignore' did not match any files
[main 1c13fa7] Add .gitignore to ignore logs
1 file changed, 1 insertion(+)

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ls -a
./ ../ .git/ GitDemo/ debug.log log/ welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ echo "*.log" > .gitignore
echo "log/" >> .gitignore
echo "/log/" >> .gitignore

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ls -a
./ ../ .git/ .gitignore GitDemo/ debug.log log/ welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git commit -m "Add .gitignore to ignore logs"
[main 581a9c1] Add .gitignore to ignore logs
1 file changed, 3 insertions(+)
create mode 100644 .gitignore

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GitDemo/

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

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git check-ignore -v debug.log
git check-ignore -v log/app.log
.gitignore:1:*.log          debug.log
.gitignore:3:/log/         log/app.log

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git rm --cached *.log
fatal: pathspec 'debug.log' did not match any files

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git rm -r --cached log/
fatal: pathspec 'log/' did not match any files

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git commit -m "Remove log files from repo and ignore them via .gitignore"
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GitDemo/

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

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git remote add origin https://github.com/AvengX/GitDemo.git
error: remote origin already exists.

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -M main
git push -u origin main
info: please complete authentication in your browser...
To https://github.com/AvengX/GitDemo.git
 ! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://github.com/AvengX/GitDemo.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$

```

GIT-3

```
MINGW64/c/Users/Lenovo/GitDemo
Lenovo@AvengX MINGW64 ~
$ git clone https://github.com/AvengX/GitDemo.git
fatal: destination path 'GitDemo' already exists and is not an empty directory.

Lenovo@AvengX MINGW64 ~
$ cd GitDemo

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git config --global diff.tool p4merge

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git config --global difftool.p4merge.cmd "\"C:/Program Files/Perforce/p4merge.exe\" \"$LOCAL\" \"$REMOTE\""

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git config --global difftool.prompt false

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git config --global merge.tool p4merge

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git config --global mergetool.p4merge.cmd "\"C:/Program Files/Perforce/p4merge.exe\" \"$LOCAL\" \"$REMOTE\" \"$BASE\" \"$MERGED\""

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git config --global mergetool.p4merge.trustExitCode true

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch GitNewBranch

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout -b GitNewBranch
fatal: a branch named 'GitNewBranch' already exists

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git switch -C GitNewBranch
fatal: a branch named 'GitNewBranch' already exists

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch
* GitNewBranch
* main

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -a
* GitNewBranch
* main

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -vv
* GitNewBranch 581a9c1 Add .gitignore to ignore logs
* main          581a9c1 Add .gitignore to ignore logs
```

```
MINGW64/c/Users/Lenovo/GitDemo
Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GitDemo/

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

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ echo "Hello from GitNewBranch" > feature.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add feature.txt
warning: in the working copy of 'feature.txt', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git commit -m "Add feature.txt on GitNewBranch"
[main b5985a7] Add feature.txt on GitNewBranch
 1 file changed, 1 insertion(+)
 create mode 100644 feature.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GitDemo/

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

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git log --oneline --graph --decorate -n 5
* b5985a7 (HEAD -> main) Add feature.txt on GitNewBranch
* 581a9c1 (GitNewBranch) Add .gitignore to ignore logs
* 1c13fa7 Add .gitignore to ignore logs
* f39c932 Add welcome.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout master
error: pathspec 'master' did not match any file(s) known to git

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout master
error: pathspec 'master' did not match any file(s) known to git

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git switch master
fatal: invalid reference: master

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git switch master
```

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git switch master
fatal: invalid reference: master

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch
  GitNewBranch
* main

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout -b GitNewBranch
fatal: a branch named 'GitNewBranch' already exists

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ echo "Hello" > f.txt
git add f.txt
git commit -m "Add f.txt"
warning: in the working copy of 'f.txt', LF will be replaced by CRLF the next time Git touches it
[main 31b35d3] Add f.txt
 1 file changed, 1 insertion(+)
 create mode 100644 f.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout main
Already on 'main'

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git merge GitNewBranch
Already up to date.

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^[[200-git fetch origin
bash: $'\E[200-git': command not found

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git pull origin master # or 'main' if applicable
fatal: couldn't find remote ref master

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git fetch origin
git pull origin main
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 861 bytes | 123.00 KiB/s, done.
From https://github.com/AvengX/GitDemo
 * [new branch]      main      -> origin/main

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git fetch origin
git pull origin main
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 861 bytes | 123.00 KiB/s, done.
From https://github.com/AvengX/GitDemo
 * [new branch]      main      -> origin/main
From https://github.com/AvengX/GitDemo
 * branch            main      -> FETCH_HEAD
fatal: refusing to merge unrelated histories

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git diff --name-only main..GitNewBranch
f.txt
feature.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git diff main..GitNewBranch
diff --git a/f.txt b/f.txt
deleted file mode 100644
index e965047..0000000
--- a/f.txt
+++ /dev/null
@@ -1 +0,0 @@
-Hello

diff --git a/feature.txt b/feature.txt
deleted file mode 100644
index b956f80..0000000
--- a/feature.txt
+++ /dev/null
@@ -1 +0,0 @@
-Hello From GitNewBranch

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git difftool -d main..GitNewBranch
/mingw64/libexec/git-core/git-mergetool--lib: line 172: C:/Program Files/Perforce/p4merge.exe: No such file or directory
warning: failed: 127

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git difftool main..GitNewBranch
/mingw64/libexec/git-core/git-mergetool--lib: line 172: C:/Program Files/Perforce/p4merge.exe: No such file or directory
fatal: external diff died, stopping at f.txt

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git merge GitNewBranch
Already up to date.

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git status

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GitDemo/

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

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git mergetool
No files need merging

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add <file-with-conflict>
bash: syntax error near unexpected token `newline'

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add GitDemo
error: 'GitDemo/' does not have a commit checked out
fatal: adding files failed

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    GitDemo/

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

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ rm -rf GitDemo/.git

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add GitDemo/
git commit -m "Add GitDemo folder"

On branch main
nothing to commit, working tree clean

Lenovo@AvengX MINGW64 ~/GitDemo (main)

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add GitDemo
fatal: pathspec 'GitDemo' did not match any files

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git add GitDemo

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git commit -m "Resolve merge conflicts: GitNewBranch -> master"
On branch main
nothing to commit, working tree clean

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git log --oneline --graph --decorate --all
* 31b35d3 (HEAD -> main) Add f.txt
* b5985a7 Add Feature.txt on GitNewBranch
* 581a9c1 (GitNewBranch) Add .gitignore to ignore logs
* 1c13fa7 Add .gitignore to ignore logs
* f39c932 Add welcome.txt
* 2faec45 (origin/main, origin/HEAD) Create .gitignore

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -d GitNewBranch
Deleted branch GitNewBranch (was 581a9c1).

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -D GitNewBranch
error: branch 'GitNewBranch' not found

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git push origin --delete GitNewBranch
error: unable to delete 'GitNewBranch': remote ref does not exist
error: failed to push some refs to 'https://github.com/AvengX/GitDemo.git'

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch
* main

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -r
  origin/HEAD -> origin/main
  origin/main

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -a
* main
  remotes/origin/HEAD -> origin/main
  remotes/origin/main

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout -b GitNewBranch main
Switched to a new branch 'GitNewBranch'

Lenovo@AvengX MINGW64 ~/GitDemo (GitNewBranch)
$ git branch -d GitNewBranch
error: cannot delete branch 'GitNewBranch' used by worktree at 'C:/Users/Lenovo/GitDemo'

Lenovo@AvengX MINGW64 ~/GitDemo (GitNewBranch)
$ ^C

Lenovo@AvengX MINGW64 ~/GitDemo (GitNewBranch)
$ git checkout main
Switched to branch 'main'

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch -d GitNewBranch
Deleted branch GitNewBranch (was 31b35d3).

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$

```


GIT-4

```
Lenovo@AvengX MINGW64 ~
$ git config --global merge.tool p4merge

Lenovo@AvengX MINGW64 ~
$ git config --global mergetool.p4merge.cmd 'p4merge "$LOCAL" "$BASE" "$REMOTE" "$MERGED"'

Lenovo@AvengX MINGW64 ~
$ git config --global diff.tool p4merge

Lenovo@AvengX MINGW64 ~
$ git config --global difftool.p4merge.cmd 'p4merge "$LOCAL" "$REMOTE"'

Lenovo@AvengX MINGW64 ~
$ git config --global mergetool.keepBackup true
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout master
git pull origin master          # if repo has remote (optional)
git status
error: pathspec 'master' did not match any file(s) known to git
fatal: couldn't find remote ref master
On branch main
nothing to commit, working tree clean

Lenovo@AvengX MINGW64 ~/GitDemo (main)
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (GitWork)
$ git checkout -b GitWork
fatal: a branch named 'GitWork' already exists

Lenovo@AvengX MINGW64 ~/GitDemo (GitWork)
$ cat > hello.xml <<'EOF'
<hello>Branch version: original from GitWork.</hello>
EOF

Lenovo@AvengX MINGW64 ~/GitDemo (GitWork)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (GitWork)
$ git commit -m "Add hello.xml on GitWork (initial)"
[GitWork 7c52337] Add hello.xml on GitWork (initial)
 1 file changed, 1 insertion(+)
 create mode 100644 hello.xml
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (GitWork)
$ cat > hello.xml <<'EOF'
<hello>Branch version: UPDATED content from GitWork.</hello>
EOF

Lenovo@AvengX MINGW64 ~/GitDemo (GitWork)
$ git status
git add hello.xml
git commit -m "Update hello.xml in GitWork"
On branch GitWork
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:   hello.xml

no changes added to commit (use "git add" and/or "git commit -a")
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it
[GitWork 5f5778b] Update hello.xml in GitWork
 1 file changed, 1 insertion(+), 1 deletion(-)
```

```

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git branch master

Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout master
Switched to branch 'master'

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ cat > hello.xml <<'EOF'
<hello>Master version: different content from master.</hello>
EOF

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git add hello.xml
git commit -m "Add hello.xml on master with different content"
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git touches it
[master d18bcba] Add hello.xml on master with different content
1 file changed, 1 insertion(+)
create mode 100644 hello.xml

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git log --oneline --graph --decorate --all
* d18bcba (HEAD -> master) Add hello.xml on master with different content
| * 5f5778b (GitWork) Update hello.xml in GitWork
| * 7c52337 Add hello.xml on GitWork (initial)
|/
* 31b35d3 (main) Add f.txt
* b5985a7 Add feature.txt on GitNewBranch
* 581a9c1 Add .gitignore to ignore logs
* 1c13fa7 Add .gitignore to ignore logs
* f39c932 Add welcome.txt
* 2faec45 (origin/main, origin/HEAD) Create .gitignore

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git diff master..GitWork -- hello.xml
diff --git a/hello.xml b/hello.xml
index 29648c9..ca494dc 100644
--- a/hello.xml
+++ b/hello.xml
@@ -1,1 @@
-<hello>Master version: different content from master.</hello>
+<hello>Branch version: UPDATED content from GitWork.</hello>

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git difftool -t p4merge master..GitWork -- hello.xml
/mingw64/libexec/git-core/git-mergetool--lib: line 172: p4merge: command not found
fatal: external diff died, stopping at hello.xml

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git merge GitWork
Auto-merging hello.xml
CONFLICT (add/add): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git mergetool
Merging:
hello.xml

Normal merge conflict for 'hello.xml':
  {local}: created file
  {remote}: created file
/mingw64/libexec/git-core/git-mergetool--lib: line 176: p4merge: command not found
merge of hello.xml failed

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git mergetool -t p4merge
Merging:
hello.xml

Normal merge conflict for 'hello.xml':
  {local}: created file
  {remote}: created file
/mingw64/libexec/git-core/git-mergetool--lib: line 176: p4merge: command not found
merge of hello.xml failed

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git log --oneline --graph --decorate --all
* d18bcba (HEAD -> master) Add hello.xml on master with different content
| * 5f5778b (GitWork) Update hello.xml in GitWork
| * 7c52337 Add hello.xml on GitWork (initial)
|/
* 31b35d3 (main) Add f.txt
* b5985a7 Add feature.txt on GitNewBranch
* 581a9c1 Add .gitignore to ignore logs
* 1c13fa7 Add .gitignore to ignore logs
* f39c932 Add welcome.txt
* 2faec45 (origin/main, origin/HEAD) Create .gitignore

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git diff master..GitWork
diff --git a/hello.xml b/hello.xml
index 29648c9..ca494dc 100644
--- a/hello.xml
+++ b/hello.xml
@@ -1,1 @@
-<hello>Master version: different content from master.</hello>
+<hello>Branch version: UPDATED content from GitWork.</hello>

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git difftool -t p4merge master..GitWork
/mingw64/libexec/git-core/git-mergetool--lib: line 172: p4merge: command not found
fatal: external diff died, stopping at hello.xml

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git checkout master
git merge GitWork
hello.xml: needs merge
error: you need to resolve your current index first
error: Merge is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both added:      hello.xml

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

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git mergetool -t p4merge
Merging:
hello.xml

Normal merge conflict for 'hello.xml':
  {local}: created file
  {remote}: created file
/mingw64/libexec/git-core/git-mergetool--lib: line 176: p4merge: command not found
merge of hello.xml failed

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git add <conflicted-file>
git commit -m "Merge GitWork into master with conflict resolution"
bash: syntax error near unexpected token `newline'
U      hello.xml
error: Committing is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$ git status
echo "*.orig" >> .gitignore      # example backup pattern
git add .gitignore
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
        both added:      hello.xml

no changes added to commit (use "git add" and/or "git commit -a")
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

Lenovo@AvengX MINGW64 ~/GitDemo (master|MERGING)
$

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ echo "*.orig" >> .gitignore
git add .gitignore
git commit -m "Ignore merge backup files"
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
[master 71a7547] Ignore merge backup files
1 file changed, 1 insertion(+)

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git branch
  GitWork
  main
* master

```

```

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git branch -d GitWork
Deleted branch GitWork (was 5f5778b).

```

```
Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git log --oneline --graph --decorate
* 71a7547 (HEAD -> master) Ignore merge backup files
* 6a9bf4b Merge GitWork into master with conflict resolution
| \
| * 5f5778b Update hello.xml in GitWork
| * 7c52337 Add hello.xml on GitWork (initial)
* | d18bcba Add hello.xml on master with different content
| /
* 31b35d3 (main) Add f.txt
* b5985a7 Add feature.txt on GitNewBranch
* 581a9c1 Add .gitignore to ignore logs
* 1c13fa7 Add .gitignore to ignore logs
* f39c932 Add welcome.txt
```

GIT-5

```
Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git checkout master
git status
Already on 'master'
On branch master
nothing to commit, working tree clean
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git branch -a
main
* master
remotes/origin/HEAD -> origin/main
remotes/origin/main
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (main)
$ git checkout master
git pull origin master
Switched to branch 'master'
fatal: couldn't find remote ref master
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git checkout -b Git-T03-HOL_002
fatal: a branch named 'Git-T03-HOL_002' already exists

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git add .
git commit -m "Add changes for Git-T03-HOL_002"
On branch master
nothing to commit, working tree clean

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git push -u origin Git-T03-HOL_002
branch 'Git-T03-HOL_002' set up to track 'origin/Git-T03-HOL_002'.
Everything up-to-date

Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git branch -r
origin/Git-T03-HOL_002
origin/HEAD -> origin/main
origin/main
```

```
Lenovo@AvengX MINGW64 ~/GitDemo (master)
$ git fetch origin
git log --oneline origin/Git-T03-HOL_002
31b35d3 (origin/Git-T03-HOL_002, Git-T03-HOL_002) Add f.txt
b5985a7 Add feature.txt on GitNewBranch
581a9c1 Add .gitignore to ignore logs
1c13fa7 Add .gitignore to ignore logs
f39c932 Add welcome.txt
```