

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1 (master)
$ mkdir new-project

welcome@HP MINGW64 ~/Desktop/git2/demo1 (master)
$ cd new-project

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git init
Initialized empty Git repository in C:/Users/welcome/Desktop/git2/demo1/new-project/
.git/

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ |
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1 (master)
$ mkdir new-project

welcome@HP MINGW64 ~/Desktop/git2/demo1 (master)
$ cd new-project

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git init
Initialized empty Git repository in C:/Users/welcome/Desktop/git2/demo1/new-project/
.git/

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ touch myfile.txt

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git init
Reinitialized existing Git repository in C:/Users/welcome/Desktop/git2/demo1/new-project/
.git/

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git add myfile.txt

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git add .

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ |
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git add myfile.txt

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git add .

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git commit -m "file created"
[master (root-commit) e6367a7] file created
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 myfile.txt

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git log
commit e6367a7c8df349213a6d24922f6a9cf0f7cff9e2 (HEAD -> master)
Author: bookiteazy <Bookiteazy@gmail.com>
Date:   Tue Jan 2 15:59:26 2024 +0530

    file created

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git log --oneline
e6367a7 (HEAD -> master) file created

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ |
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git diff

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git diff myfile.txt

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ |
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$ git remote -v
origin https://github.com/majhus001/addrecord.git (fetch)
origin https://github.com/majhus001/addrecord.git (push)

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$ |
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git remote add origin https://github.com/majhus001/addrecord.git

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git remote -v
origin https://github.com/majhus001/addrecord.git (fetch)
origin https://github.com/majhus001/addrecord.git (push)

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git pull origin master
fatal: couldn't find remote ref master

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git branch
* master

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 211 bytes | 70.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/majhus001/addrecord.git
```

```
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$ git commit -m "updated"
[myfirst-branch 11214fc] updated
 1 file changed, 5 insertions(+)
 create mode 100644 index.html

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$ git push origin myfirst-branch
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 307 bytes | 307.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'myfirst-branch' on GitHub by visiting:
remote:     https://github.com/majhus001/addrecord/pull/new/myfirst-branch
remote:
To https://github.com/majhus001/addrecord.git
 * [new branch]      myfirst-branch -> myfirst-branch

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$
```

```
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$ git checkout -b ADD-branch
Switched to a new branch 'ADD-branch'

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (ADD-branch)
$
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (myfirst-branch)
$ git checkout -b ADD-branch
Switched to a new branch 'ADD-branch'

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (ADD-branch)
$
```

```
MINGW64:/c/Users/welcome/Desktop/git2/demo1/new-project
fatal: There is no merge in progress (MERGE_HEAD missing).

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git branch
  ADD-branch
* master
  myfirst-branch

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git status
On branch master
nothing to commit, working tree clean

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git merge myfirst-branch
Updating e6367a7..11214fc
Fast-forward
 index.html | 5 +++++
 1 file changed, 5 insertions(+)
 create mode 100644 index.html

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ |
```

```
welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git branch -d myfirst-branch
Deleted branch myfirst-branch (was 11214fc).

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git fetch --prune origin

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$ git branch
  ADD-branch
* master

welcome@HP MINGW64 ~/Desktop/git2/demo1/new-project (master)
$
```

```
File Actions Edit View Help
GNU nano 7.0                                     mysql.sh *
#!/bin/bash

# MySQL login details
MYSQL_USER="root"
MYSQL_PASSWORD="password"

# Function to execute a MySQL query and print the results
execute_query() {
    local query=$1
    local database=$2
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" "$database" -e "$query"
}

# Function to create a new database
create_database() {
    local database=$1
    execute_query "CREATE DATABASE $database"
}

# Function to create a new table in a database
create_table() {
    local database=$1
    local table=$2
    local schema=$3
    execute_query "CREATE TABLE $table ($schema)" "$database"
}

# Function to insert data into a table
insert_data() {
    local database=$1
    local table=$2
    local data=$3
    execute_query "INSERT INTO $table VALUES ($data)" "$database"
}

# Function to update data in a table
update_data() {
    local database=$1
    local table=$2
    local data=$3
    execute_query "UPDATE $table SET $data" "$database"
}

# Function to delete data from a table
delete_data() {
    local database=$1
    local table=$2
    local condition=$3
    execute_query "DELETE FROM $table WHERE $condition" "$database"
}

# Function to select data from a table
select_data() {
    local database=$1
    local table=$2
    local condition=$3
    local columns=$4
    execute_query "SELECT $columns FROM $table WHERE $condition" "$database"
}

# Function to drop a database
drop_database() {
    local database=$1
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "DROP DATABASE $database"
}

# Function to show databases
show_databases() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW DATABASES"
}

# Function to show tables in a database
show_tables() {
    local database=$1
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW TABLES IN $database"
}

# Function to show columns in a table
show_columns() {
    local database=$1
    local table=$2
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW COLUMNS FROM $table IN $database"
}

# Function to show indexes in a table
show_indexes() {
    local database=$1
    local table=$2
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW INDEXES FROM $table IN $database"
}

# Function to show status of a database
show_status() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW STATUS"
}

# Function to show variables of a database
show_variables() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW VARIABLES"
}

# Function to show grants for a user
show_grants() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW GRANTS FOR '$USER'@'$HOST'"
}

# Function to show processlist
show_processlist() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW PROCESSLIST"
}

# Function to show errors
show_errors() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW ERRORS"
}

# Function to show warnings
show_warnings() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW WARNINGS"
}

# Function to show status of a connection
show_status() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW STATUS"
}

# Function to show variables of a connection
show_variables() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW VARIABLES"
}

# Function to show grants for a user
show_grants() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW GRANTS FOR '$USER'@'$HOST'"
}

# Function to show processlist
show_processlist() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW PROCESSLIST"
}

# Function to show errors
show_errors() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW ERRORS"
}

# Function to show warnings
show_warnings() {
    mysql -u "$MYSQL_USER" -p"$MYSQL_PASSWORD" -e "SHOW WARNINGS"
}
```

```
# Function to process files in a directory
process_files() {
    echo "Processing files in the current directory"
    for file in *; do
        echo "Processing: $file"
        # Add your processing logic here
    done
    echo "Processing complete."
}

# Function to list files in a directory
list_files() {
    echo "Listing files in the current directory"
    ls -l
}

# Function to create a new file
create_file() {
    echo "Enter the name of the new file:"
    read -r filename
    touch "$filename"
    echo "File '$filename' created successfully."
}

# User and Group Management Tasks

# Function to list users
list_users() {
    echo "Listing users:"
    cut -d: -f1 /etc/passwd
}

# Function to list groups
list_groups() {
    echo "Listing groups:"
    cut -d: -f1 /etc/group
}

# Function to create a new user
create_user() {
    echo "Enter the username of the new user:"
    read -r username
    sudo useradd "$username"
    echo "User '$username' created successfully."
```