

# Sarang Deshpande

## GitHub

GitHub is the Distributed Version Control System. In GitHub the Large projects can be handled.

Software Development process is made easy with the help of GitHub. While working in a team, we need to establish a coordination, with the help of this tool this can be achieved. It provides space to store the large number of Files remotely. Also, from the local system one can upload files to form a Collection. This Collection of files is Called as Repository. GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. This tutorial teaches you GitHub essentials like repositories, branches, commits, and pull requests.

### CLI Operations:

It is a good idea to introduce yourself to Git with your name and public email address before doing any operation. The easiest way to do so is:

```
$ git config --global user.name "Your Name Comes Here"
$ git config --global user.email you@yourdomain.example.com
```

### Importing a new project

```
$ mkdir projectFile
```

```
$ cd project
$ git init
```

Git will reply:

Initialized empty Git repository in .git/

You've now initialized the working directory—you may notice a new directory created, named ".git".

Next, tell Git to take a snapshot of the contents of all files under the current directory (note the **.**), with **git add**:

```
$ git add .
```

This snapshot is now stored in a temporary staging area which Git calls the "index". You can permanently store the contents of the index in the repository with **git commit**:

```
$ git commit -m "message"
```

This will prompt you for a commit message. You've now stored the first version of your project in Git.

### **Making changes**

Modify some files, then add their updated contents to the index:

```
$ git add file1.txt
```

```
$ git add file2.txt
```

You can also get a brief summary of the situation with **git status**:

```
$ git status
On branch master
Changes to be committed:
  Your branch is up to date with 'origin/master'.
  (use "git restore --staged <file>..." to unstage)

    modified:   file1.txt
    modified:   file2.txt
```

If you need to make any further adjustments, do so now, and then add any newly modified content to the index. Finally, commit your changes with:

```
$ git commit
```

This will again prompt you for a message describing the change, and then record a new version of the project.

### **Viewing project history**

At any point you can view the history of your changes using

```
$ git log
```

### **Managing branches**

A single Git repository can maintain multiple branches of development. To create a new branch named "subbranch", use

```
$ git branch subbranch
```

If you now run

```
$ git branch
```

you'll get a list of all existing branches:

```
subbranch
```

```
* master
```

## To go to the different branches on the Repository

```
$ git checkout master
```

```
$ git checkout subbranch
```

## To merge the created branch with the master branch

```
$ git merge subbranch
```

```
MINGW64/c/Users/Sarang/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
Sarang@SARANG MINGW64 ~
$ cd eclipse-workspace/hdfc_fsd/
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd
$ ls
bin/ src/
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd
$ cd bin/
Adapter_pattern/ hdfc_fsd/ java_program_practice/
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd
$ cd bin/Adapter_pattern/
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
$ ls
BankDemo.class BankDetail.class Credit.class Customer.class
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
$ git config --global user.name "Sarang"
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
$ git config --global user.email "sarangdeshpande719@gmail.com"
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
$ git init
Initialized empty Git repository in C:/Users/Sarang/eclipse-workspace/hdfc_fsd/b
in/Adapter_pattern/.git/
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ ls -la
total 18
drwxr-xr-x 1 Sarang 197611  0 Jul 14 21:18 ./
drwxr-xr-x 1 Sarang 197611  0 Jul 14 15:58 ../
drwxr-xr-x 1 Sarang 197611  0 Jul 14 21:18 .git/
-rw-r--r-- 1 Sarang 197611 590 Jul 14 15:58 BankDemo.class
-rw-r--r-- 1 Sarang 197611 958 Jul 14 15:58 BankDetail.class
-rw-r--r-- 1 Sarang 197611 165 Jul 14 15:58 Credit.class
-rw-r--r-- 1 Sarang 197611 1845 Jul 14 15:58 Customer.class
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ ls -la
./ ../ .git/ BankDemo.class BankDetail.class Credit.class Customer.class
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  BankDemo.class
  BankDetail.class
  Credit.class
  Customer.class

nothing added to commit but untracked files present (use "git add" to track)
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ touch file1.txt
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ cat "This is Adapter Pattern" >> file1.txt
cat: 'This is Adapter Pattern': No such file or directory
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ cat This is Adapter Pattern >> file1.txt
cat: This: No such file or directory
cat: is: No such file or directory
cat: Adapter: No such file or directory
cat: Pattern: No such file or directory
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ echo This is Adapter Pattern >> file1.txt
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ cat file1.txt
This is Adapter Pattern
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  BankDemo.class
  BankDetail.class
  Credit.class
  Customer.class
  file1.txt

nothing added to commit but untracked files present (use "git add" to track)
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git add file1.txt
warning: in the working copy of 'file1.txt', LF will be replaced by CRLF the nex
t time Git touches it
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master
```

```
MINGW64/c/Users/Sarang/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   BankDemo.class
        new file:   BankDetail.class
        new file:   Credit.class
        new file:   Customer.class
        new file:   File1.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git commit -m "Adapter Pattern"
[master (root-commit) 6d8573c] Adapter Pattern
5 files changed, 1 insertion(+)
create mode 100644 BankDemo.class
create mode 100644 BankDetail.class
create mode 100644 Credit.class
create mode 100644 Customer.class
create mode 100644 File1.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git log
commit 6d8573cd3378edad278ccb451cf7ef03251472f (HEAD -> master)
Author: Sarang <sarangeshpande7719@gmail.com>
Date:   Thu Jul 14 21:21:41 2022 +0530

    Adapter Pattern

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ touch File2.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ echo this is 2nd file >> file2.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file2.txt

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

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git add file2.txt
warning: in the working copy of 'file2.txt', LF will be replaced by CRLF the next time Git touches it

MINGW64/c/Users/Sarang/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git commit -m "Added 2 text files"
[master f7f431f] Added 2 text files
1 file changed, 1 insertion(+)
create mode 100644 file2.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git log
commit f7f431ff126d154dd299eefdb64edab640b32e (HEAD -> master)
Author: Sarang <sarangeshpande7719@gmail.com>
Date:   Thu Jul 14 21:23:16 2022 +0530

    Added 2 text files

commit 6d8573cd3378edad278ccb451cf7ef03251472f
Author: Sarang <sarangeshpande7719@gmail.com>
Date:   Thu Jul 14 21:21:41 2022 +0530

    Adapter Pattern

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git checkout ^C

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git checkout 6d8573cd3378edad278ccb451cf7ef03251472f
Note: switching to '6d8573cd3378edad278ccb451cf7ef03251472f'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Examples:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 6d8573c Adapter Pattern

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern ((6d8573c
...))
$ ^C

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern ((6d8573c
...))
$ git checkout master
Previous HEAD position was 6d8573c Adapter Pattern
Switched to branch 'master'
```

```
MINGW64/c/Users/Sarang/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ ls
MainDemo.class  Credit.class  file1.txt
MainDetail.class  Customer.class  file2.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git branch
* master

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git branch subbranch

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master
nothing to commit, working tree clean

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git checkout subbranch
Switched to branch 'subbranch'

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ git status
On branch subbranch
nothing to commit, working tree clean

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ echo committed the change >> file2.txt

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ git status
On branch subbranch
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:   file2.txt

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

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ git add file2.txt
warning: in the working copy of 'file2.txt', LF will be replaced by CRLF the next time Git touches it

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ git status
On branch subbranch

Switched to branch 'master'

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git log
commit f7f431ff126d1549d2799eefdb164edab640b52e (HEAD -> master)
Author: Sarang <sarangeshpande7719@gmail.com>
Date: Thu Jul 14 21:29:16 2022 +0530

    Added 2 text files

commit 6d8573cd3378ada278ccb451cf7ef03251472f
Author: Sarang <sarangeshpande7719@gmail.com>
Date: Thu Jul 14 21:21:41 2022 +0530

    Adapter Pattern

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git checkout subbranch
Switched to branch 'subbranch'

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ git log
commit 82a328911c501748e79601806ad0b8489ec4f (HEAD -> subbranch)
Author: Sarang <sarangeshpande7719@gmail.com>
Date: Thu Jul 14 21:29:16 2022 +0530

    Changed the text file again

commit f7f431ff126d1549d2799eefdb164edab640b52e (master)
Author: Sarang <sarangeshpande7719@gmail.com>
Date: Thu Jul 14 21:23:16 2022 +0530

    Added 2 text files

commit 6d8573cd3378ada278ccb451cf7ef03251472f
Author: Sarang <sarangeshpande7719@gmail.com>
Date: Thu Jul 14 21:21:41 2022 +0530

    Adapter Pattern

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (subbranch)
$ git checkout master
Switched to branch 'master'

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git merge subbranch
Updating f7f431f..82a3289
Fast-forward
 file2.txt | 1 +
 1 file changed, 1 insertion(+)
```

## To Connect the Remote Repository with the locally created Repository

\$ git remote add origin URL

\$ git remote -v

## To push towards remote repository

\$ git push -u origin master

```
MINGW64/c/Users/Sarang/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern
$ git merge subbranch
Updating 7f431f..82a3289
Fast-forward
 file.txt | 1 +
 1 file changed, 1 insertion(+)

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git status
On branch master
nothing to commit, working tree clean

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ ^C

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (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.

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git remote add origin https://github.com/the-SARANG/new-java-fsd-praactice.git

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git remote -v
origin https://github.com/the-SARANG/new-java-fsd-praactice.git (fetch)
origin https://github.com/the-SARANG/new-java-fsd-praactice.git (push)

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git push origin master

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$ git push -u origin master
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 4 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (13/13), 2.77 KiB | 283.00 KiB/s, done.
Total 13 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/the-SARANG/new-java-fsd-praactice/pull/new/master
remote:
To https://github.com/the-SARANG/new-java-fsd-praactice.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

Sarang@SARANG MINGW64 ~/eclipse-workspace/hdfc_fsd/bin/Adapter_pattern (master)
$
```

## GUI GitHub:

Using GUI the Repository Creation, creating branch, comparing, editing can be done without the Command. Readymade Options are available. Along with the above options, Additional Option such as Code, Pull requests, issues, Actions, projects, Wiki, Insights, etc. Are available.

the-SARANG Merge pull request #1 from the-SARANG/newBranch

BankDemo.class	Adapter Pattern	1 hour ago
BankDemo.java	java files	1 hour ago
BankDetail.class	Adapter Pattern	1 hour ago
BankDetail.java	java files	1 hour ago
Credit.class	Adapter Pattern	1 hour ago
Credit.java	java files	1 hour ago
Customer.class	Adapter Pattern	1 hour ago
Customer.java	java files	1 hour ago
README.md	Initial commit	2 hours ago

README.md

# java-fsd-practice

About

No description, website, or topics provided.

Readme

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Java 100.0%

java-fsd-practice / BankDemo.java / <> Jump to +

the-SARANG java files

Latest commit 9e58a79 1 hour ago

1 contributor

12 lines (9 sloc) 224 Bytes

```
1 package Adapter_pattern;
2
3 public class BankDemo {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         Credit cd = new Customer();
8         cd.getBankDetail();
9         cd.getCreditDetail();
10    }
11
12 }
```

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

the-SARANG java files

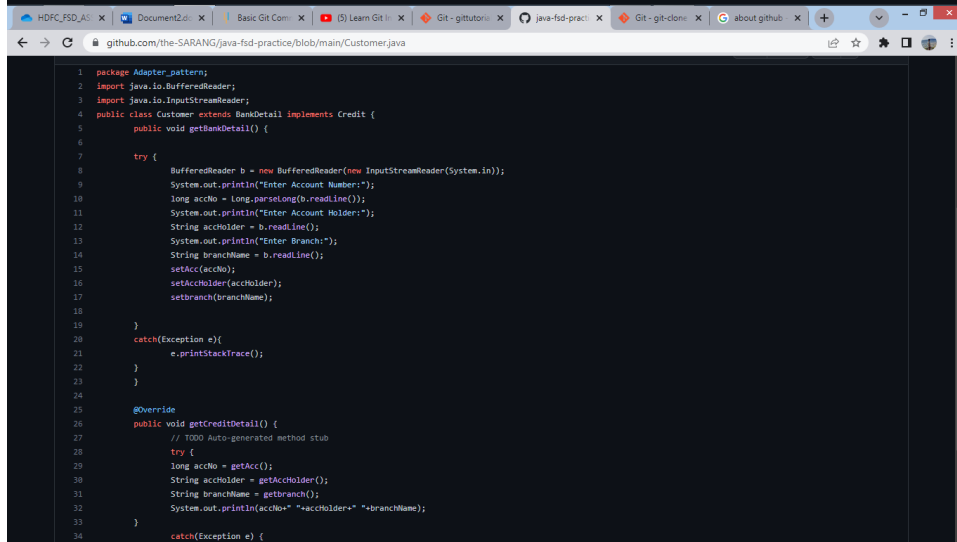
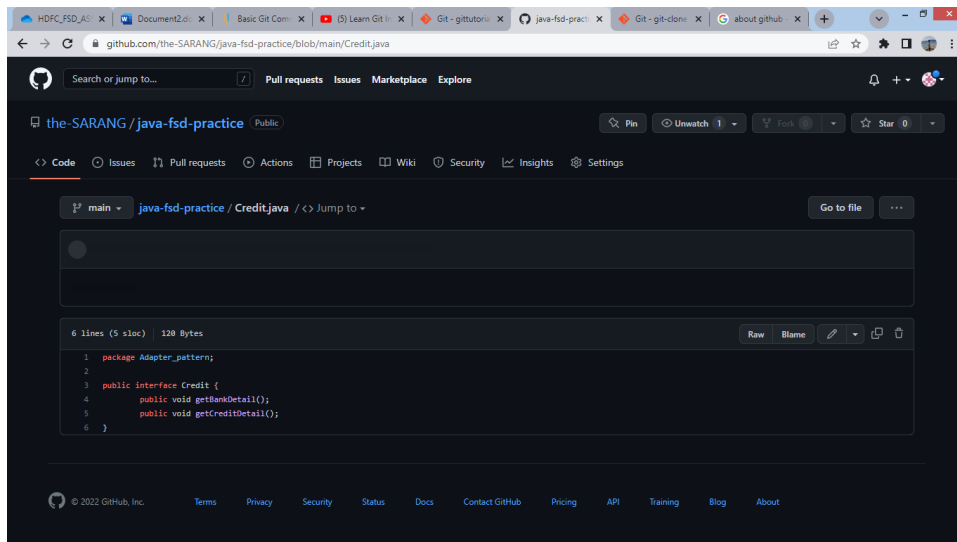
Latest commit 9e58a79 1 hour ago

1 contributor

25 lines (23 sloc) 475 Bytes

```
1 package Adapter_pattern;
2
3 public class BankDetail {
4     long accho;
5     String accholder, branchName;
6
7     public long getAcch() {
8         return accho;
9     }
10    public String getAccholder() {
11        return accholder;
12    }
13    public String getbranch() {
14        return branchName;
15    }
16    public void setAcch(long accho) {
17        this.accho = accho;
18    }
19    public void setAccholder(String name) {
20        this.accholder = name;
21    }
22    public void setbranch(String branch) {
23        this.branchName = branch;
24    }
25 }
```

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About



## Output of the Program:

