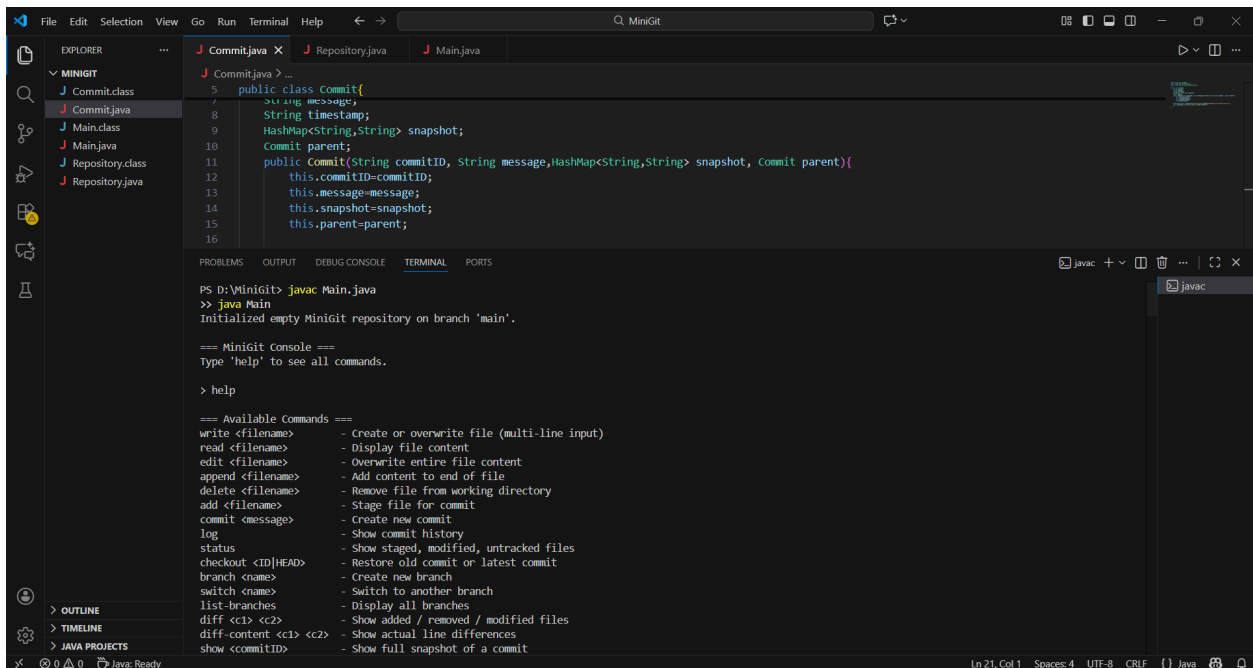


## Screenshots of MiniGit



The screenshot shows an IDE with the following components:

- EXPLORER:** A tree view showing the project structure: MINIGIT, Commit.class, Commit.java, Main.class, Main.java, Repository.class, and Repository.java.
- EDITOR:** The `Commit.java` file is open, showing the following code:

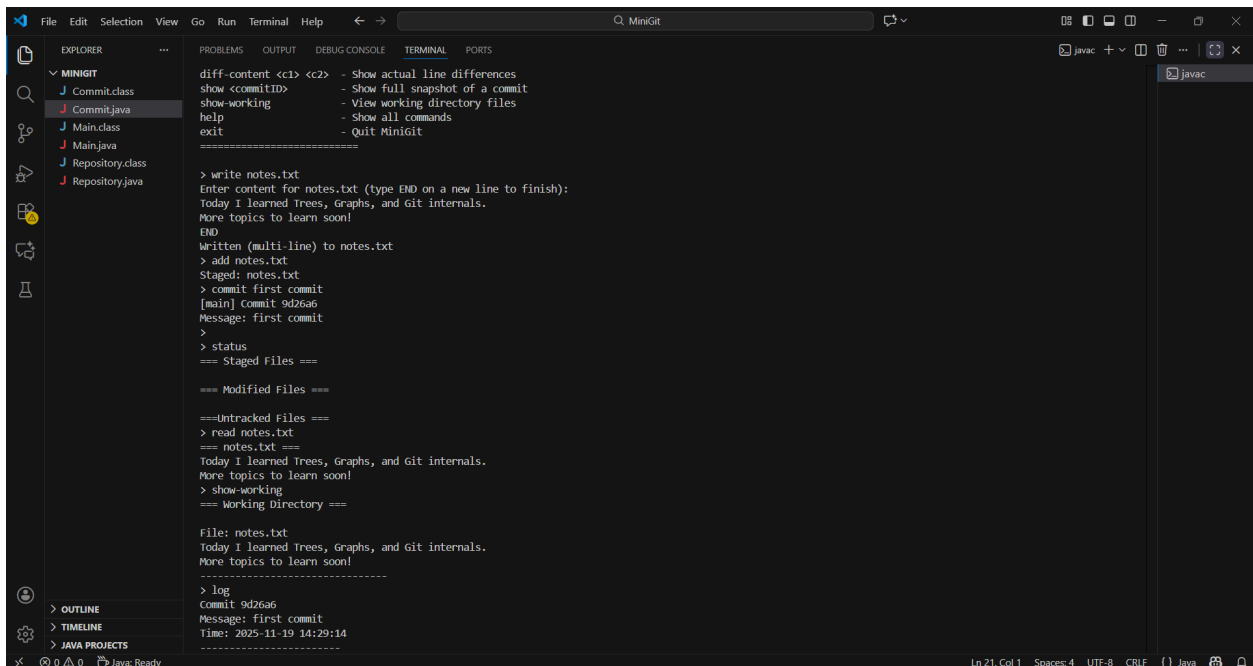
```
1 public class Commit{
2     String message;
3     String timestamp;
4     HashMap<String,String> snapshot;
5     Commit parent;
6     public Commit(String commitID, String message,HashMap<String,String> snapshot, Commit parent){
7         this.commitID=commitID;
8         this.message=message;
9         this.snapshot=snapshot;
10        this.parent=parent;
11    }
12 }
```
- TERMINAL:** The terminal window shows the execution of `java Main`. The output is:

```
PS D:\MiniGit> java Main
>> java Main
Initialized empty Minigit repository on branch 'main'.

=== Minigit Console ===
Type 'help' to see all commands.

> help

=== Available Commands ===
write <filename> - Create or overwrite file (multi-line input)
read <filename> - Display file content
edit <filename> - Overwrite entire file content
append <filename> - Add content to end of file
delete <filename> - Remove file from working directory
add <filename> - Stage file for commit
commit <message> - Create new commit
log - Show commit history
status - Show staged, modified, untracked files
checkout <ID|HEAD> - Restore old commit or latest commit
branch <name> - Create new branch
switch <name> - Switch to another branch
list-branches - Display all branches
diff <c1> <c2> - Show added / removed / modified files
diff-content <c1> <c2> - Show actual line differences
show <commitID> - Show full snapshot of a commit
```



The screenshot shows the same IDE as the first screenshot, but the terminal window now displays the output of the `diff-content` command:

```
diff-content <c1> <c2> - Show actual line differences
show <commitID> - Show full snapshot of a commit
show-working - View working directory files
help - Show all commands
exit - Quit Minigit

=====

> write notes.txt
Enter content for notes.txt (type END on a new line to finish):
Today I learned Trees, Graphs, and Git internals.
More topics to learn soon!
END
Written (multi-line) to notes.txt
> add notes.txt
Staged: notes.txt
> commit first commit
[main] Commit 9d26a6
Message: first commit
>
> status
=== Staged Files ===

=== Modified Files ===

=== Untracked Files ===
> read notes.txt
=== notes.txt ===
Today I learned Trees, Graphs, and Git internals.
More topics to learn soon!
> show-working
=== Working Directory ===

File: notes.txt
Today I learned Trees, Graphs, and Git internals.
More topics to learn soon!

> log
Commit 9d26a6
Message: first commit
Time: 2025-11-19 14:29:14
```

The screenshot shows the VS Code interface with the Explorer sidebar on the left displaying a project named 'MINIGIT' containing files: Commit.class, Commit.java, Main.class, Main.java, Repository.class, and Repository.java. The Terminal window is active, showing the following commands and their outputs:

```
> edit notes.txt
Existing content:
Today I learned Trees, Graphs, and Git internals.
More topics to learn soon!

Enter new content (type END on a new line to finish):
Updated notes after learning diff.
END
Updated content in notes.txt
> status
=== Staged Files ===

=== Modified Files ===
notes.txt

===Untracked Files ===
> commit updated notes
Nothing to commit. Staging area is empty.
> write intro.txt
Enter content for intro.txt (type END on a new line to finish):
Hello, this is an intro file.
END
Written (multi-line) to intro.txt
> status
=== Staged Files ===

=== Modified Files ===
notes.txt

===Untracked Files ===
intro.txt
> delete intro.txt
status
Deleted file: intro.txt
> == Staged Files ==

=== Modified Files ===
notes.txt

===Untracked Files ===
> status
```

The screenshot shows the VS Code interface with the same Explorer sidebar. The Terminal window continues with the following commands and outputs:

```
===Untracked Files ===
> status
=== Staged Files ===

=== Modified Files ===
notes.txt

===Untracked Files ===
> append notes.txt
Existing content:
Updated notes after learning diff.

Enter content to append (type END on a new line to finish):
This line was added for branch testing.
END
Appended content to notes.txt
> add notes.txt
Staged: notes.txt
> commit added append line
[main] Commit b2426d
Message: added append line
> log
Commit b2426d
Message: added append line
Time: 2025-11-19 14:32:12
-----
Commit 9d26a6
Message: first commit
Time: 2025-11-19 14:29:14
-----
> diff b2426d 9d26a6

=== Added Files ===

=== Removed Files ===

=== Modified Files ===
notes.txt
> diff-content b2426d 9d26a6
```

The status bar at the bottom indicates 'Ln 21, Col 1', 'Spaces: 4', 'UTF-8', 'CRLF', and the active file is 'Java'.

```
> diff-content b2426d 9d26a6

=== Content Differences ===

File: notes.txt
--- Old Content ---
Updated notes after learning diff.
This line was added for branch testing.
--- New Content ---
Today I learned Trees, Graphs, and Git internals.
More topics to learn soon!
> checkout b2426d
Checked out to commit b2426d
> show-working
=== Working Directory ===

File: notes.txt
Updated notes after learning diff.
This line was added for branch testing.
-----
> checkout HEAD
Checked out to HEAD (latest commit).
> branch new
Branch 'new' created at commit b2426d
> list-branches
=== Branches ===
new
main *
> edit notes.txt
Existing content:
Updated notes after learning diff.
This line was added for branch testing.

Enter new content (type END on a new line to finish):
Sath new on main.
END
Updated content in notes.txt
> switch new
Error: You have unstaged modifications. Commit or discard changes before switching.
> checkout HEAD
Checked out to HEAD (latest commit).
```

```
Error: You have unstaged modifications. Commit or discard changes before switching.
> checkout HEAD
Checked out to HEAD (latest commit).
> switch new
Switched to branch 'new'.
> list-branches
Unknown command. Type 'help' to see list of commands.
> list-branches
=== Branches ===
new
main *
> write featureA.txt
Enter content for featureA.txt (type END on a new line to finish):
This file exists only on featureA branch.
END
Written (multi-line) to featureA.txt
> add featureA.txt
Staged: featureA.txt
> commit
added branch file
[new] Commit 3e7cf7
Message: added branch file
> switch main
Switched to branch 'main'.
> show-working
=== Working Directory ===

File: notes.txt
Updated notes after learning diff.
This line was added for branch testing.
-----
> log
Commit b2426d
Message: added append line
Time: 2025-11-19 14:32:12
-----
Commit 9d26a6
Message: first commit
Time: 2025-11-19 14:29:14
-----
> switch main
Switched to branch 'main'.
```

The screenshot shows an IDE interface with a dark theme. The Explorer sidebar on the left shows a project named 'MINIGIT' containing files: 'Commit.class', 'Commit.java', 'Main.class', 'Main.java', 'Repository.class', and 'Repository.java'. The 'Commit.java' file is selected. The main editor area displays the output of a terminal session, which includes git commands and their corresponding messages and commit hashes. The terminal output is as follows:

```
Switched to branch 'main'.
> log
Commit b2426d
Message: added append line
Time: 2025-11-19 14:32:12
-----
Commit 9d26a6
Message: first commit
Time: 2025-11-19 14:29:14
-----
> switch new
Switched to branch 'new'.
> log
Commit 3e7cf7
Message: added branch file
Time: 2025-11-19 14:37:22
-----
Commit b2426d
Message: added append line
Time: 2025-11-19 14:32:12
-----
Commit 9d26a6
Message: first commit
Time: 2025-11-19 14:29:14
-----
> diff b2426d 3e7cf7

=== Added Files ===
featureA.txt

=== Removed Files ===

=== Modified Files ===
> diff-content b2426d 3e7cf7

=== Content Differences ===
No content differences found.
> exit
Exiting Minigit...
PS D:\Minigit>
```

The status bar at the bottom indicates the current cursor position is 'Ln 21, Col 1', the encoding is 'UTF-8', and the line ending is 'CRLF'. The language mode is set to 'Java'.