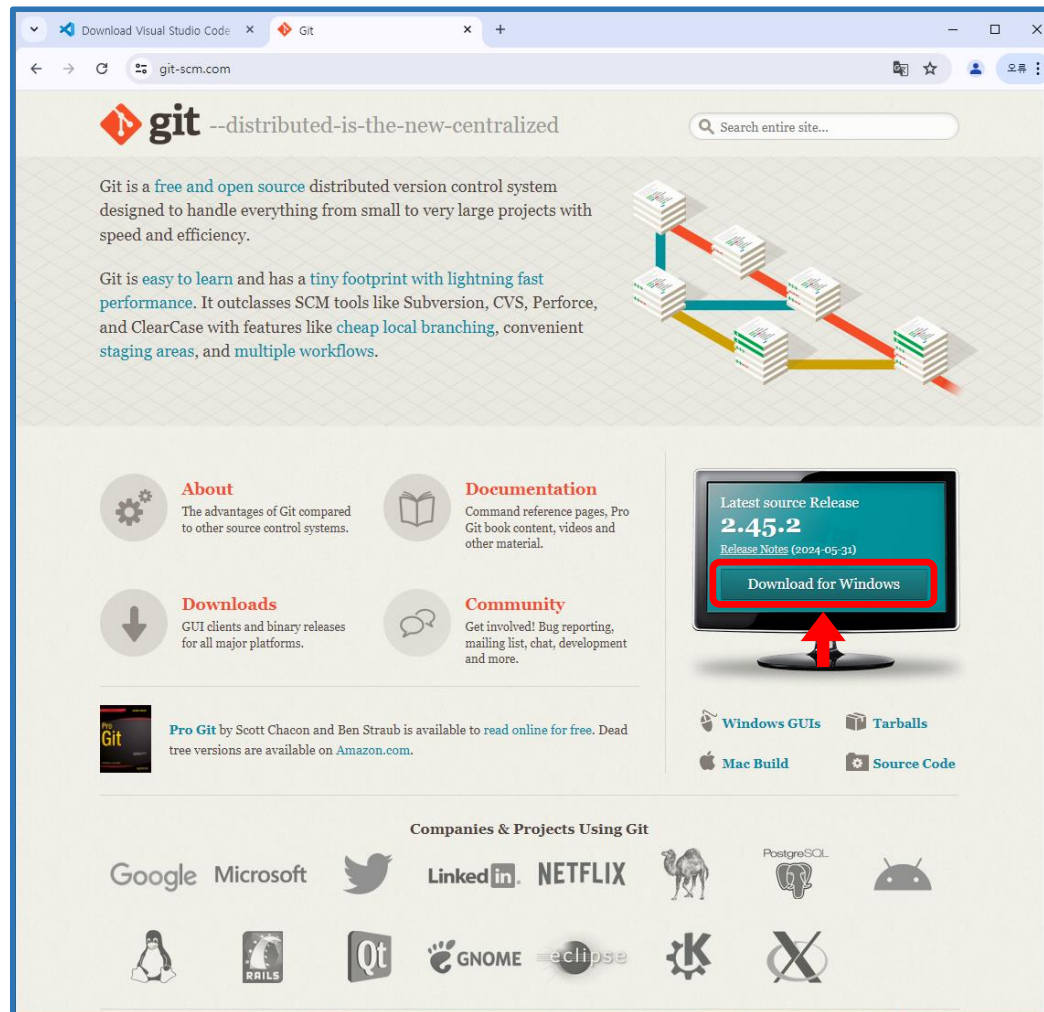


4. Installation & Setting

1. Installation

■ Download Git

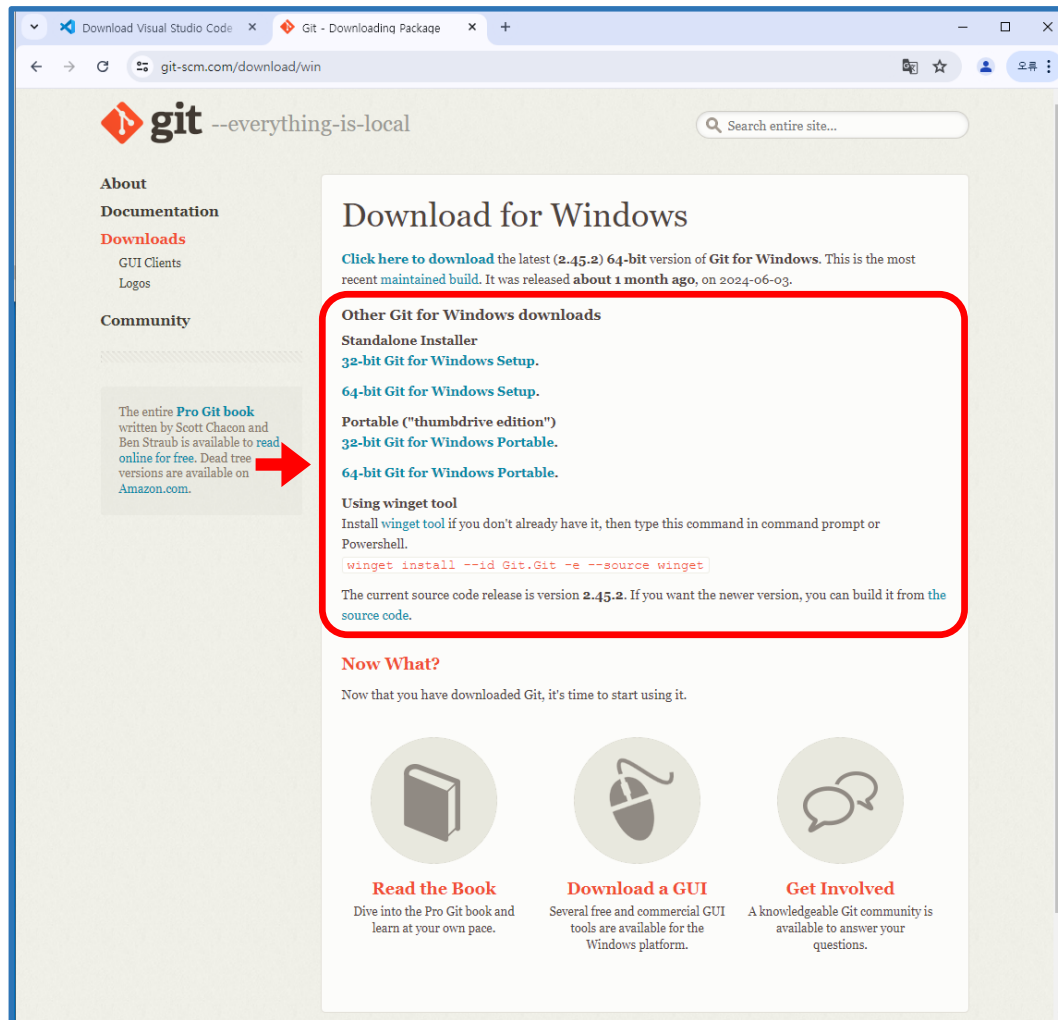
- <https://git-scm.com/>



1. Installation

■ Download Git

- <https://git-scm.com/download/win>

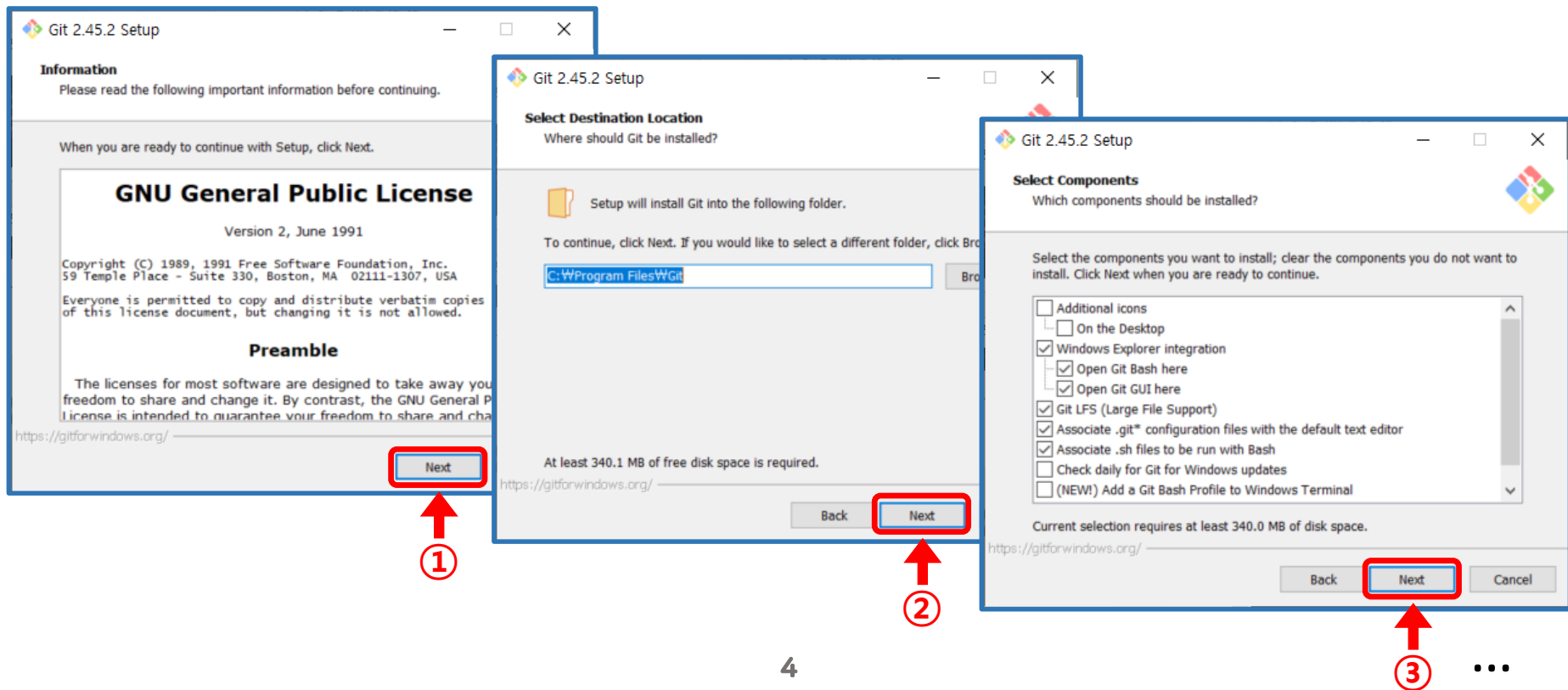


1. Installation

- Execute the downloaded .exe file



- Set it as default and move on to the “Next”

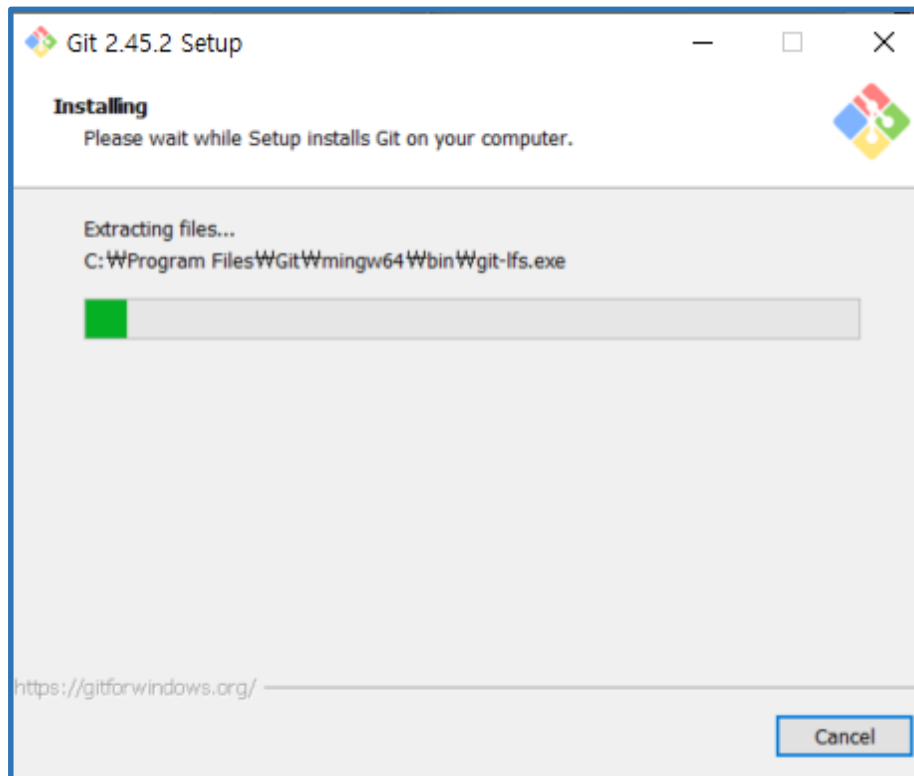


1. Installation

- Execute the downloaded .exe file

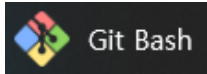


- Set it as default and install

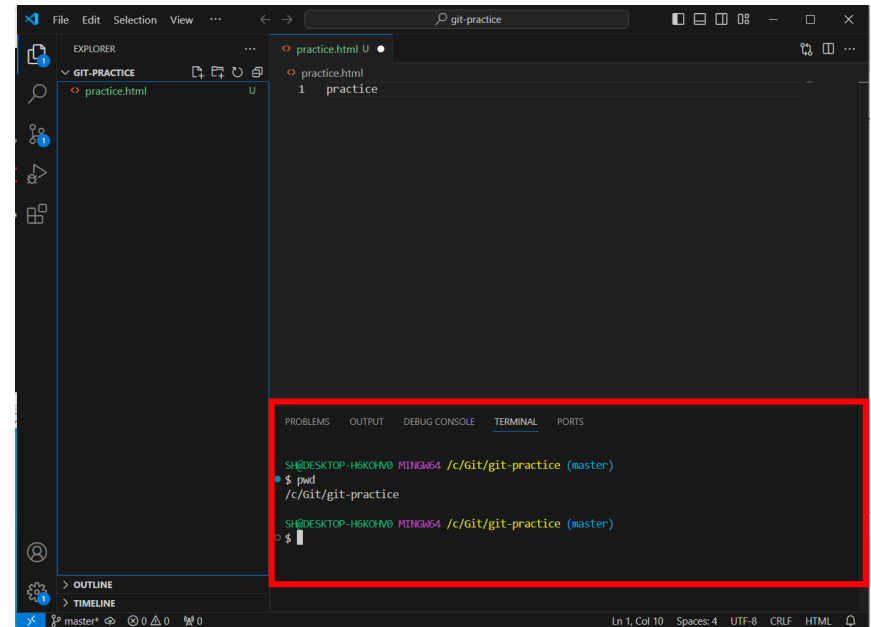


1. Installation

- Now can control Git in “Git Bash” or “VScode”



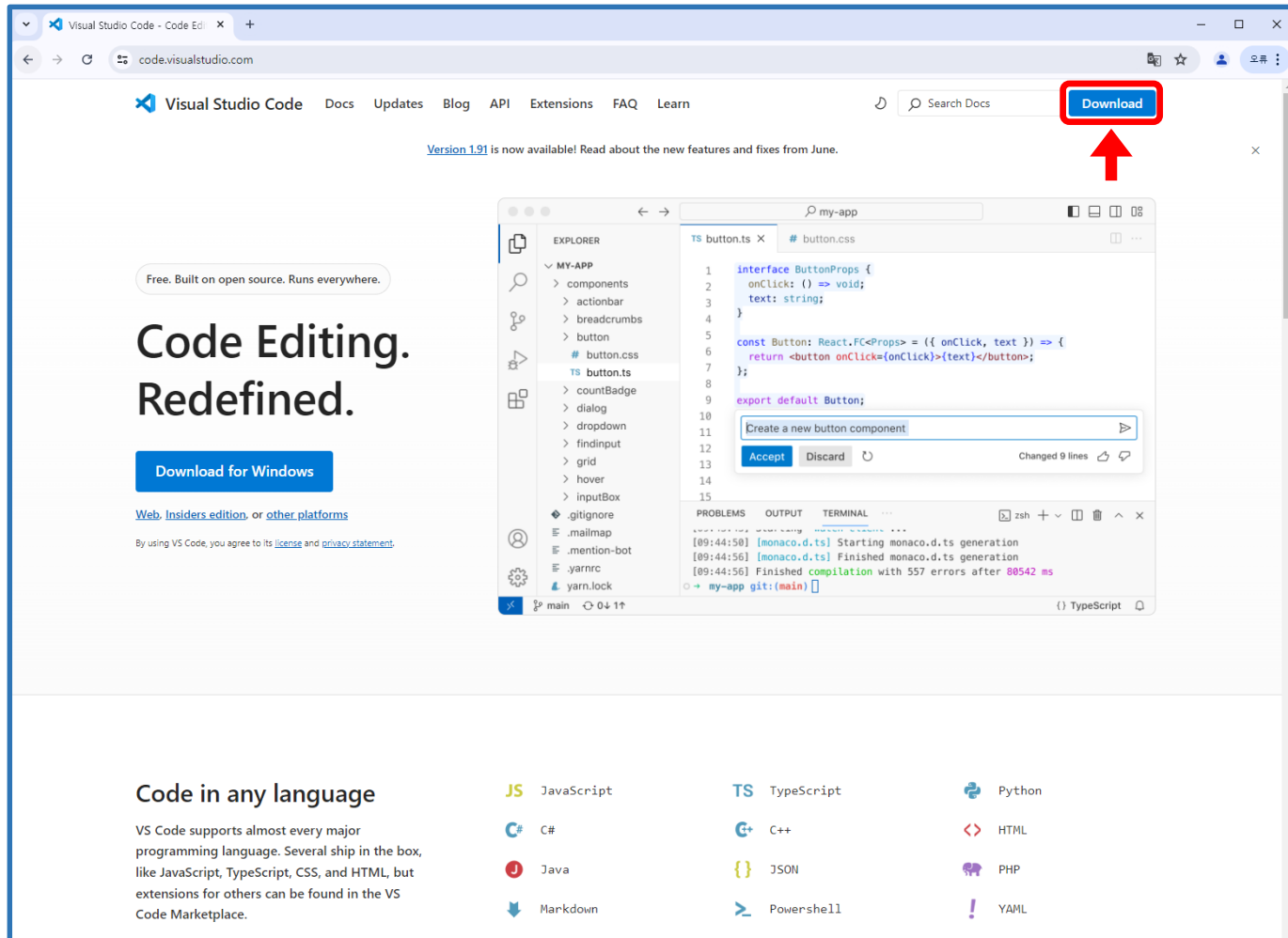
Visual Studio Code



1. Installation

■ Download Visual Studio Code (VScode)

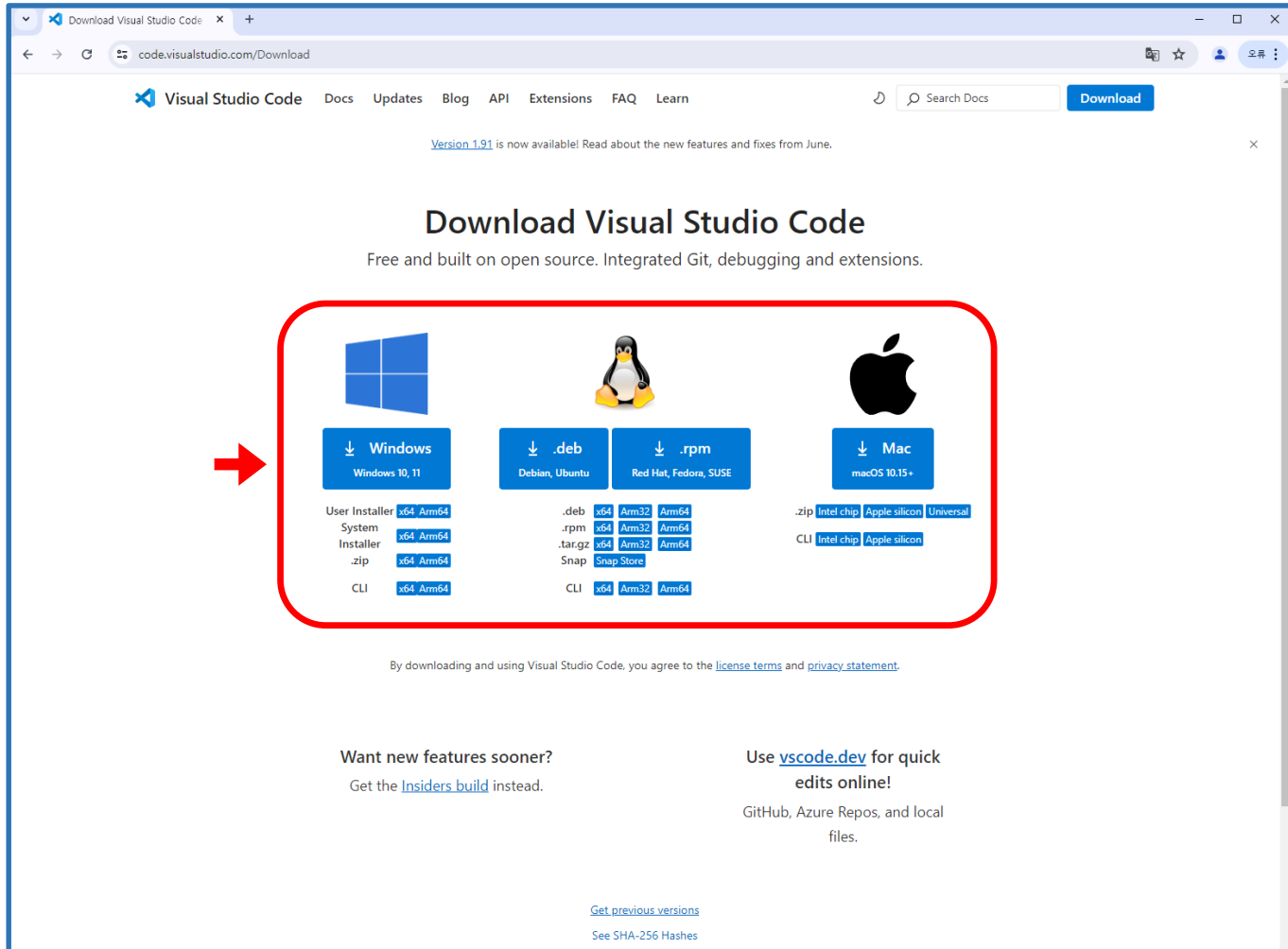
- <https://code.visualstudio.com/>



1. Installation

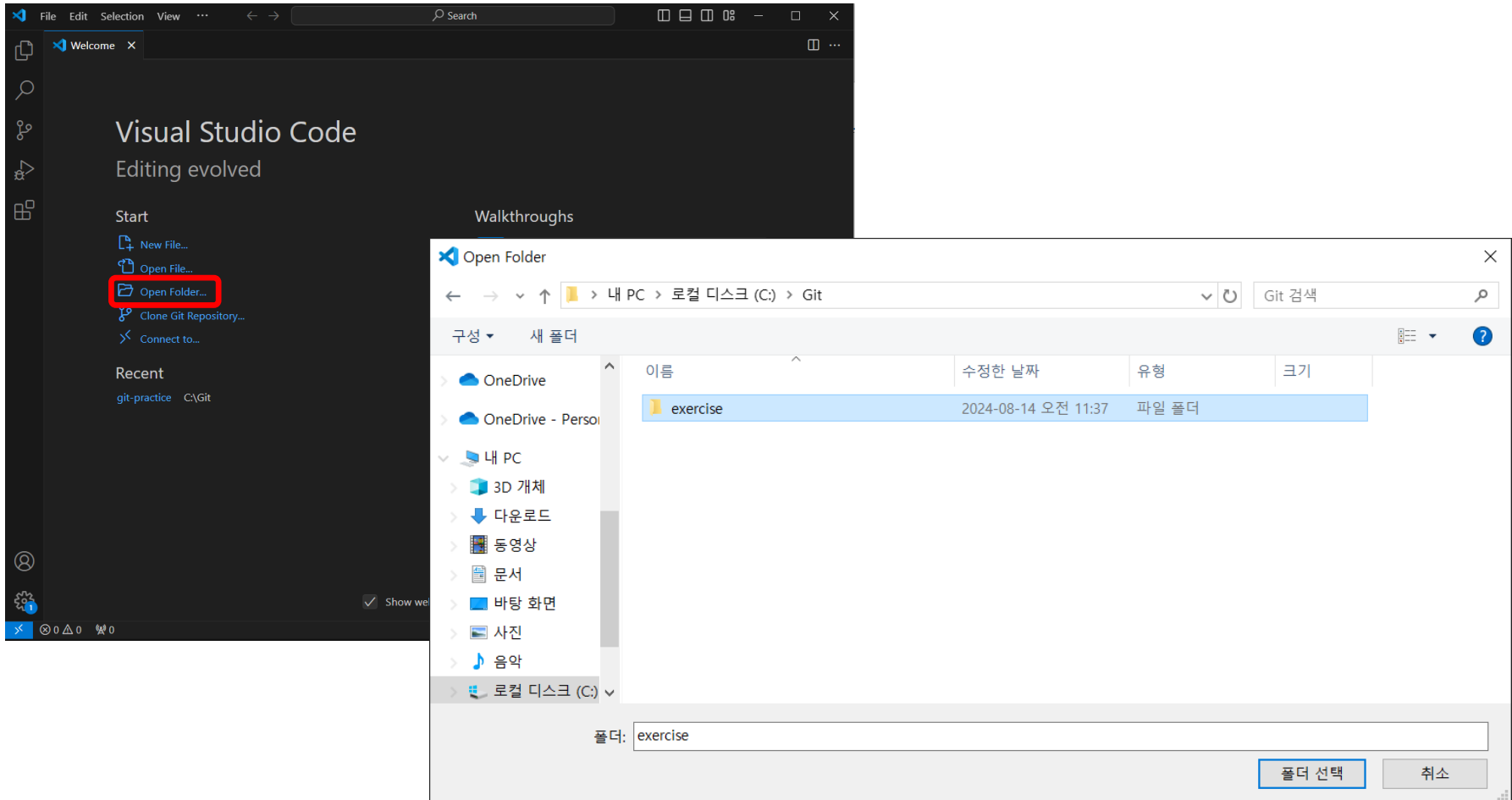
■ Download Visual Studio Code (VScode)

- <https://code.visualstudio.com/Download>



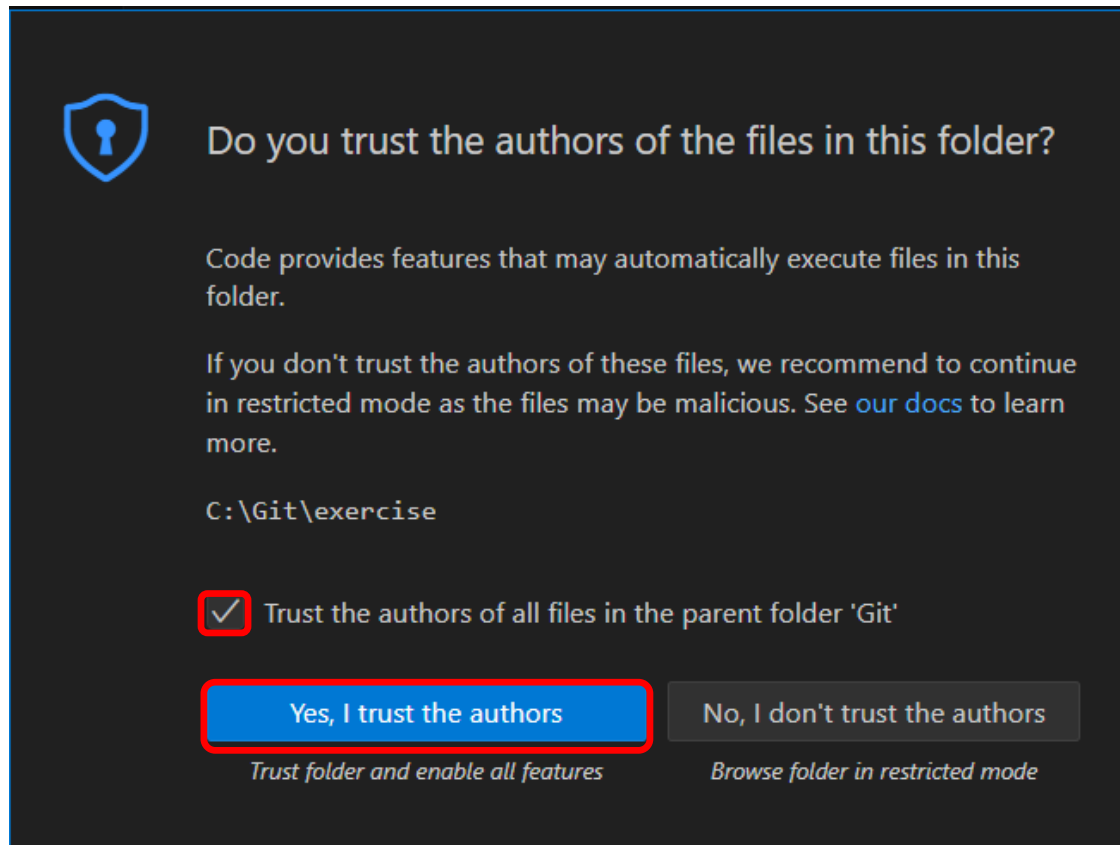
2. Setting

- Make folder “exercise” in your directory
- Open this folder in VScode



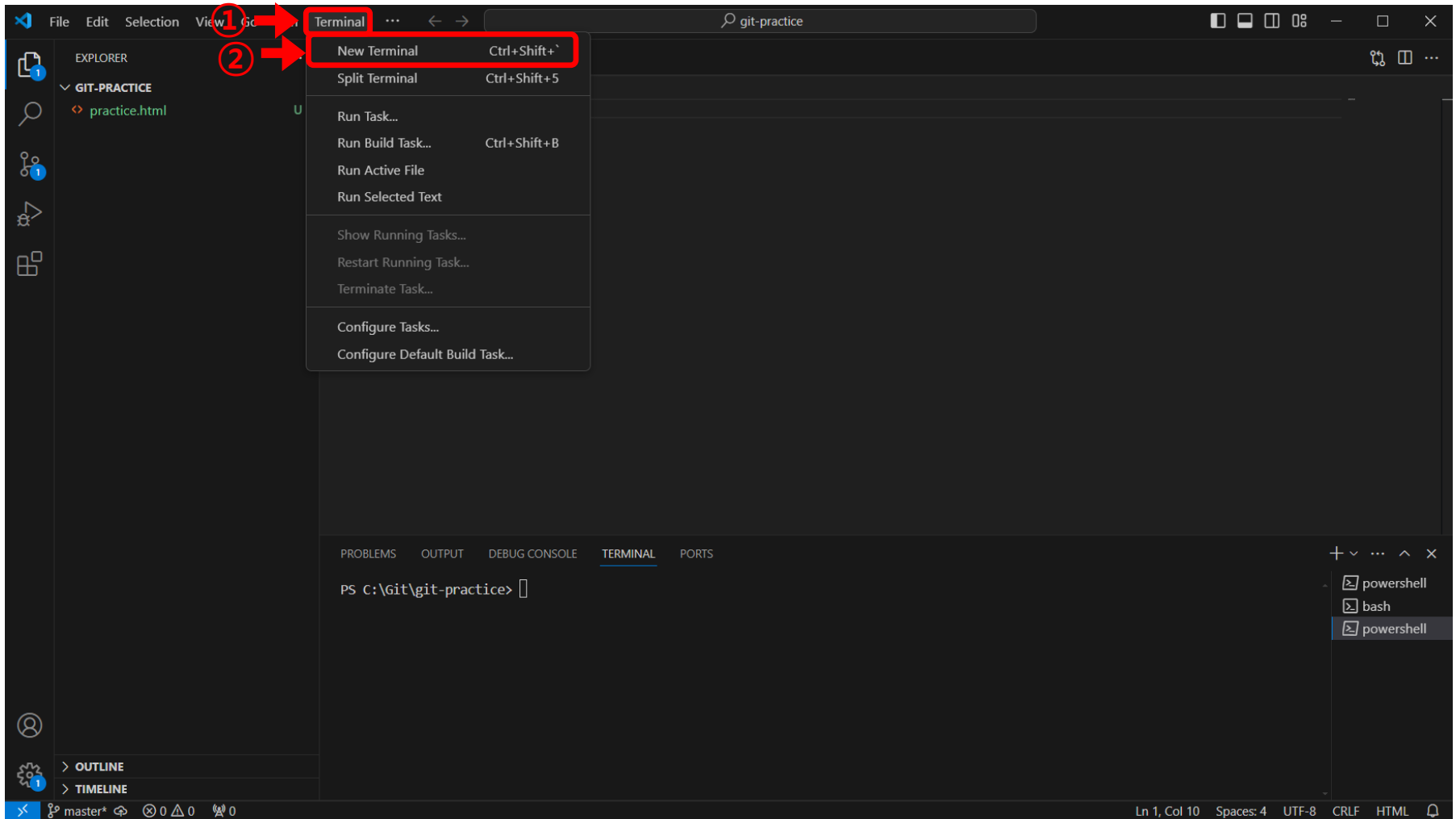
2. Setting

- Check “Trust the authors ...”
- Yes, I trust the authors



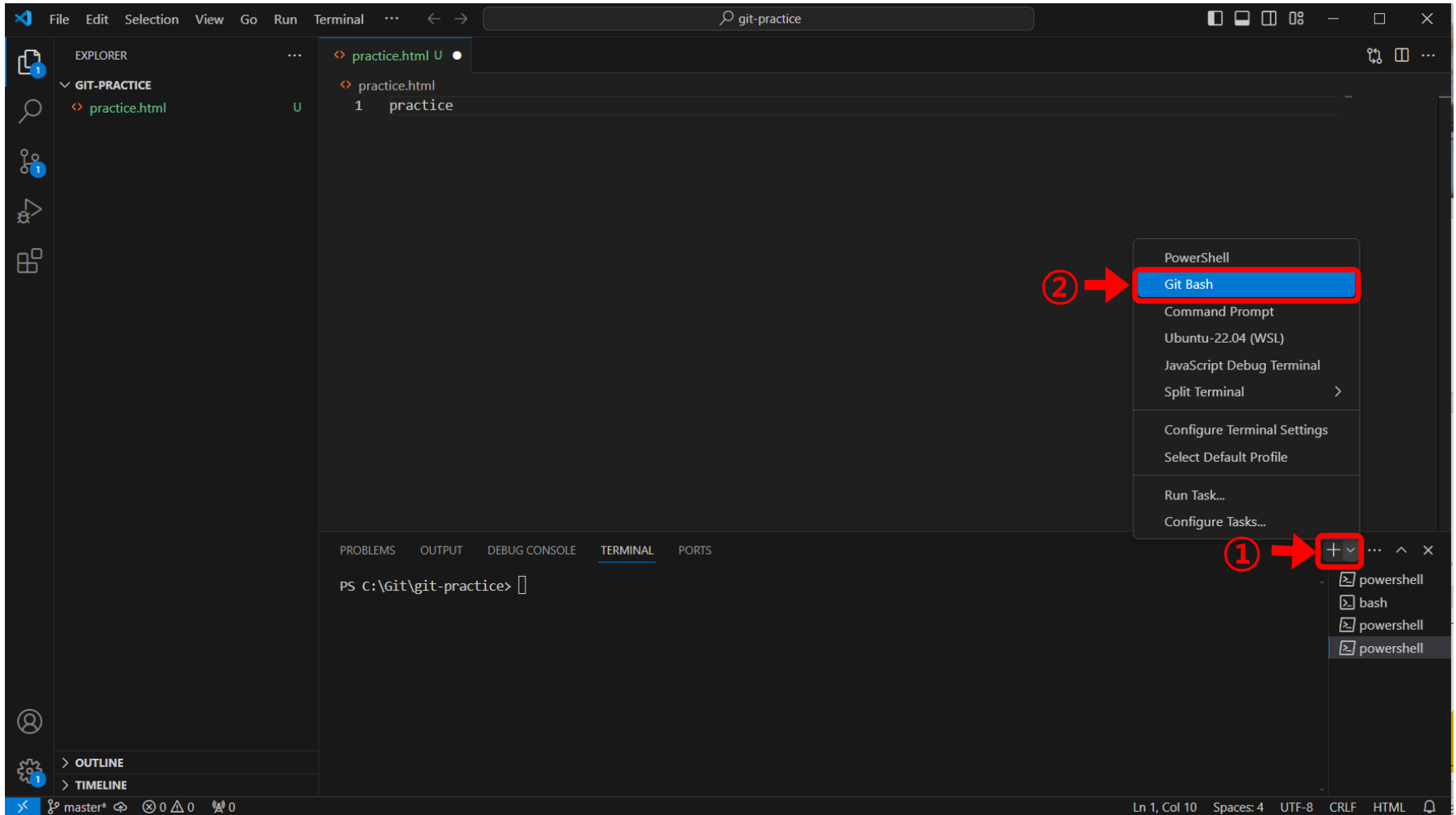
2. Setting

■ Open “New Terminal” in VScode



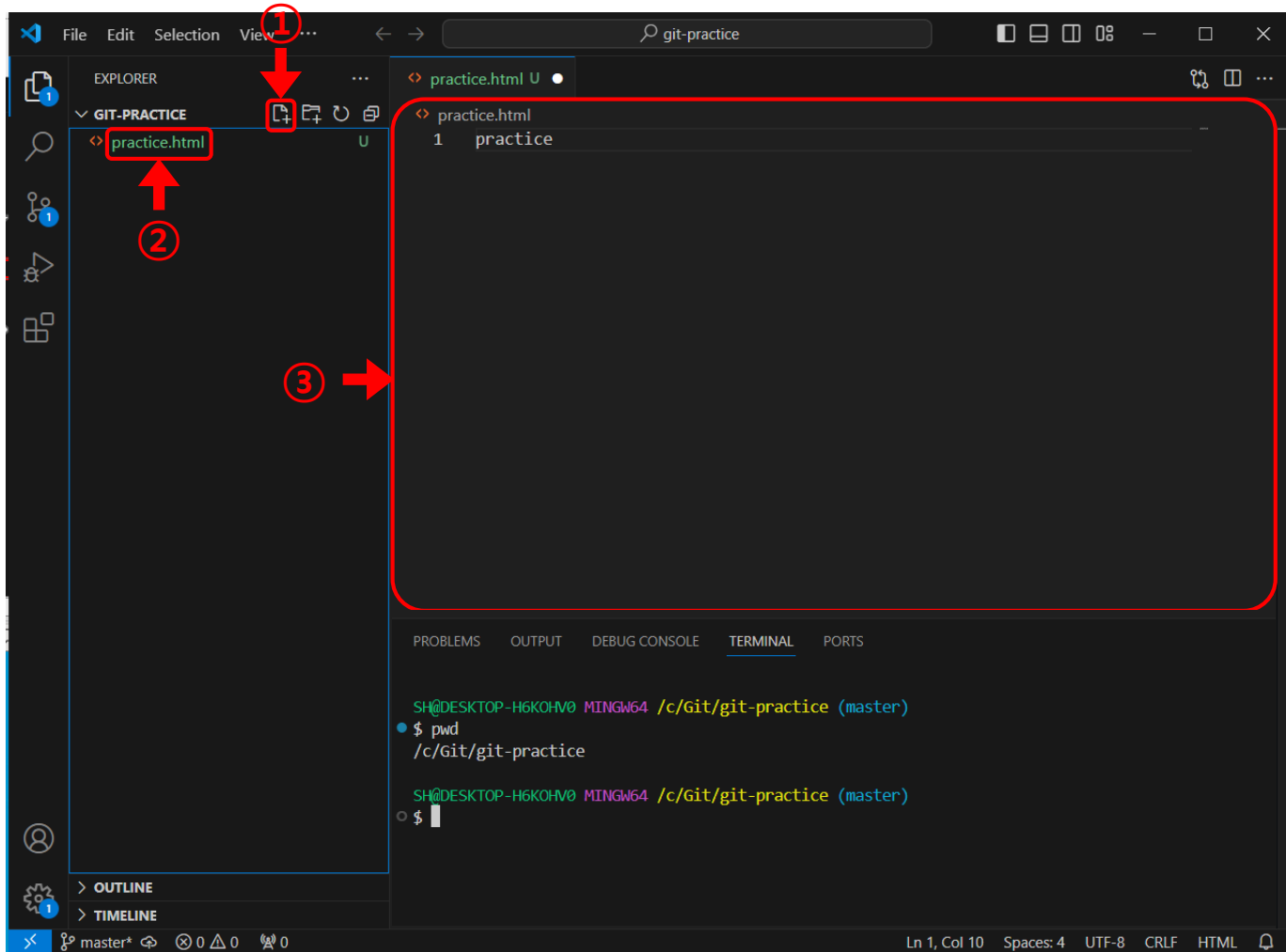
2. Setting

- Change terminal to “Git Bash” or “Bash”



2. Setting

- Now you can add file (①, ②)
- Write code (③)



4. Practical Exercise

1. Practical Exercise

- **Check the path of your directory**
- **Make your directory to a Git repository**
 - Set Git configuration
- **Initialize your Git repository**
- **Check the “.git” folder to make sure it has been created**
 - Check the list of files in your working directory
- **Check the name of your branch in the repository**

1. Practical Exercise

```
File Edit Selection View Go ... exercise
EXPLORER
EXERCISE
TERMINAL
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise
$ pwd
/c/Git/exercise
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise
$ git config --global user.name NR
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise
$ git config --global user.email nrim.cas@gmail.com
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise
$ git config --global -l
user.name=NR
user.email=nrim.cas@gmail.com
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise
$ git init
Initialized empty Git repository in C:/Git/exercise/.git/
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ ls -a
./ ../ .git/
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$
```


2. Practical Exercise

- **Create a new file “ex1.txt” in your directory**
 - Can create it using VScode or a terminal command
- **Check the status of your working directory**
- **Stage your new file in the index**
- **Commit your new file to your repository with the message “add ex1.txt”**
- **Check the history of your repository**

2. Practical Exercise

The screenshot shows the Visual Studio Code interface with the following components:

- EXPLORER:** A folder named 'EXERCISE' is expanded, showing a file 'ex1.txt'. A red arrow points to this file.
- TERMINAL:** The terminal shows the following commands and output:

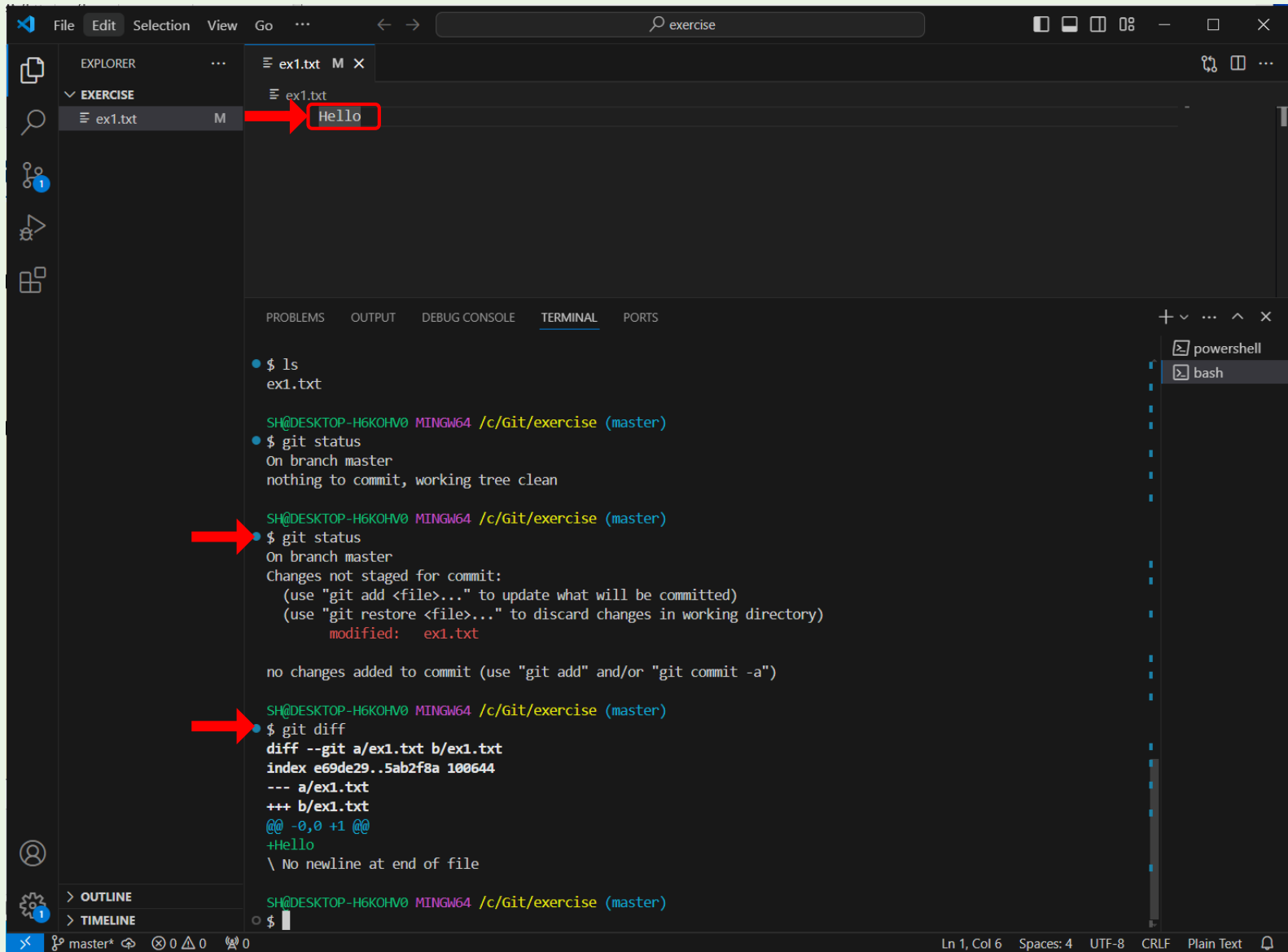
```
./ ./ .git/  
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)  
$ git status  
On branch master  
  
No commits yet  
  
Untracked files:  
  (use "git add <file>..." to include in what will be committed)  
    ex1.txt  
  
nothing added to commit but untracked files present (use "git add" to track)  
  
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)  
$ git add ex1.txt  
  
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)  
$ git commit -m "add ex1.txt"  
[master (root-commit) 7769ca4] add ex1.txt  
1 file changed, 0 insertions(+), 0 deletions(-)  
create mode 100644 ex1.txt  
  
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)  
$ git log  
commit 7769ca49da8fe5d4e8e1aaf5300c6dbd0623236e (HEAD -> master)  
Author: NR <nrim.cas@gmail.com>  
Date: Wed Aug 14 12:09:53 2024 +0900
```

Red arrows point to the 'git status', 'git add', 'git commit', and 'git log' commands.
- STATUS BAR:** The status bar at the bottom shows 'add ex1.txt' in a red box.

3. Practical Exercise

- **Write a code “Hello” in the file “ex1.txt”**
- **Check the status and difference of your working directory**
 - git status & git diff
- **Stage your new file in the index**
- **Check the status and differences of your working directory**
 - git status & git diff

3. Practical Exercise



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a folder named 'EXERCISE' containing a file 'ex1.txt'. The file 'ex1.txt' is open in the editor, and its content is 'Hello'. A red arrow points to the word 'Hello' in the editor. The Terminal panel at the bottom shows the output of several commands:

```
$ ls
ex1.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git status
On branch master
nothing to commit, working tree clean

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (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:   ex1.txt

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

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff
diff --git a/ex1.txt b/ex1.txt
index e69de29..5ab2f8a 100644
--- a/ex1.txt
+++ b/ex1.txt
@@ -0,0 +1 @@
+Hello
\ No newline at end of file

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$
```

Red arrows point to the 'git status' and 'git diff' commands in the terminal. The status bar at the bottom shows 'master*' and 'Ln 1, Col 6 Spaces: 4 UTF-8 CRLF Plain Text'.

3. Practical Exercise

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows a folder named 'EXERCISE' containing a file 'ex1.txt'. The file 'ex1.txt' is open in the editor, showing the text 'Hello'. A red arrow points from the Explorer to the file name 'ex1.txt', and another red arrow points from the file name to the text 'Hello'.

The Terminal window at the bottom shows the following commands and output:

```
modified:   ex1.txt

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

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff
diff --git a/ex1.txt b/ex1.txt
index e69de29..5ab2f8a 100644
--- a/ex1.txt
+++ b/ex1.txt
@@ -0,0 +1 @@
+Hello
\ No newline at end of file

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git add .

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   ex1.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$
```

Red arrows point to the 'git add .' command, the 'git status' command, and the 'git diff' command in the terminal.

4. Practical Exercise

- **Check the differences between the working directory and the repository**
- **Check the differences between the index and the repository**
- **Commit the previous change**
 - commit message: add first code in ex1.txt
- **Check the history of your repository**

4. Practical Exercise

The screenshot shows the Visual Studio Code interface. In the Explorer pane on the left, a folder named 'EXERCISE' is expanded, showing a file 'ex1.txt'. The file is open in the editor, and its content is 'Hello'. A red arrow points from the file name 'ex1.txt' in the Explorer to the terminal output. The terminal window shows the following commands and output:

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff
diff --git a/ex1.txt b/ex1.txt
index e69de29..5ab2f8a 100644
--- a/ex1.txt
+++ b/ex1.txt
@@ -0,0 +1 @@
+Hello
\ No newline at end of file

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff HEAD
diff --git a/ex1.txt b/ex1.txt
index e69de29..5ab2f8a 100644
--- a/ex1.txt
+++ b/ex1.txt
@@ -0,0 +1 @@
+Hello
\ No newline at end of file

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$
```

Red arrows also point to the 'Hello' text in the file editor and the 'Hello' text in the terminal output.

4. Practical Exercise

The screenshot shows the Visual Studio Code interface. In the Explorer sidebar, a folder named 'EXERCISE' is expanded, showing a file 'ex1.txt'. A red arrow points from the file name in the sidebar to the file name in the editor tab. The editor window shows the file 'ex1.txt' with the text 'Hello' on the first line, which is also highlighted by a red box. Below the editor, the TERMINAL panel is active, displaying the following commands and output:

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git commit -m "add first code in ex1.txt"
[master 07658c8] add first code in ex1.txt
1 file changed, 1 insertion(+)

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git log
commit 07658c8d20d06bb11174253631717a6db9d11765 (HEAD -> master)
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:45:19 2024 +0900

    add first code in ex1.txt

commit 7769ca49da8fe5d4e8e1aaf5300c6dbd0623236e
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:09:53 2024 +0900

    add ex1.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$
```

Red arrows point from the terminal output to the 'add first code in ex1.txt' commit message and the 'git log' command. The terminal also shows a list of commits, with the first commit highlighted by a red box. The status bar at the bottom indicates the current branch is 'master' and shows 0 changes.

5. Practical Exercise

- Add a code “Welcome” in the file “ex1.txt”
- Expect the differences between the working directory, the index, and the repository and check the results
- Commit the changes

5. Practical Exercise

The screenshot shows the Visual Studio Code interface with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows a folder named 'EXERCISE' containing a file 'ex1.txt'. The code editor shows the content of 'ex1.txt' with the text 'Hello' on line 1 and 'Welcome' on line 2. The terminal shows the following commands and output:

```
$ 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:   ex1.txt

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

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff
diff --git a/ex1.txt b/ex1.txt
index 5ab2f8a..577e7a4 100644
--- a/ex1.txt
+++ b/ex1.txt
@@ -1,2 @@
-Hello
\ No newline at end of file
+Hello
+Welcome
\ No newline at end of file

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff HEAD
diff --git a/ex1.txt b/ex1.txt
index 5ab2f8a..577e7a4 100644
--- a/ex1.txt
+++ b/ex1.txt
@@ -1,2 @@
-Hello
\ No newline at end of file
+Hello
+Welcome
\ No newline at end of file

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git add .

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git commit -m "add second code in ex1.txt"
[main 68e331f] add second code in ex1.txt
1 file changed, 2 insertions(+), 1 deletion(-)

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (master)
$ git diff --cached
```

Red arrows point to the 'Welcome' text in the code editor, the first three git commands in the terminal, and the 'git diff --cached' command.

6. Practical Exercise

- **Check the history and the file “ex1.txt”**
- **Hard reset the second commit in branch_2**
 - Second commit message: add first code in ex1.txt
- **Check the history of branch_2 and the file “ex1.txt”**

6. Practical Exercise

The screenshot shows the Visual Studio Code interface with the Explorer, Search, and Run and Debug panels. The Explorer panel shows a file named `ex1.txt` under a workspace named `EXERCISE`. The Search panel shows the file `ex1.txt` with a search result at line 1: `Hello`. The Run and Debug panel shows the `TERMINAL` tab with the following output:

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_2)
$ git log
commit 68e331f181a30945c305de67433a1bb0f5a91e3a (HEAD -> branch_2, main, branch_1)
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 13:15:21 2024 +0900

    add second code in ex1.txt

commit 07658c8d20d06bb11174253631717a6db9d11765
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:45:19 2024 +0900

    add first code in ex1.txt

commit 7769ca49da8fe5d4e8e1aaf5300c6dbd0623236e
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:09:53 2024 +0900

    add ex1.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_2)
$ git reset --hard 07658c8
HEAD is now at 07658c8 add first code in ex1.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_2)
$ git log
commit 07658c8d20d06bb11174253631717a6db9d11765 (HEAD -> branch_2)
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:45:19 2024 +0900

    add first code in ex1.txt

commit 7769ca49da8fe5d4e8e1aaf5300c6dbd0623236e
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:09:53 2024 +0900

    add ex1.txt
```

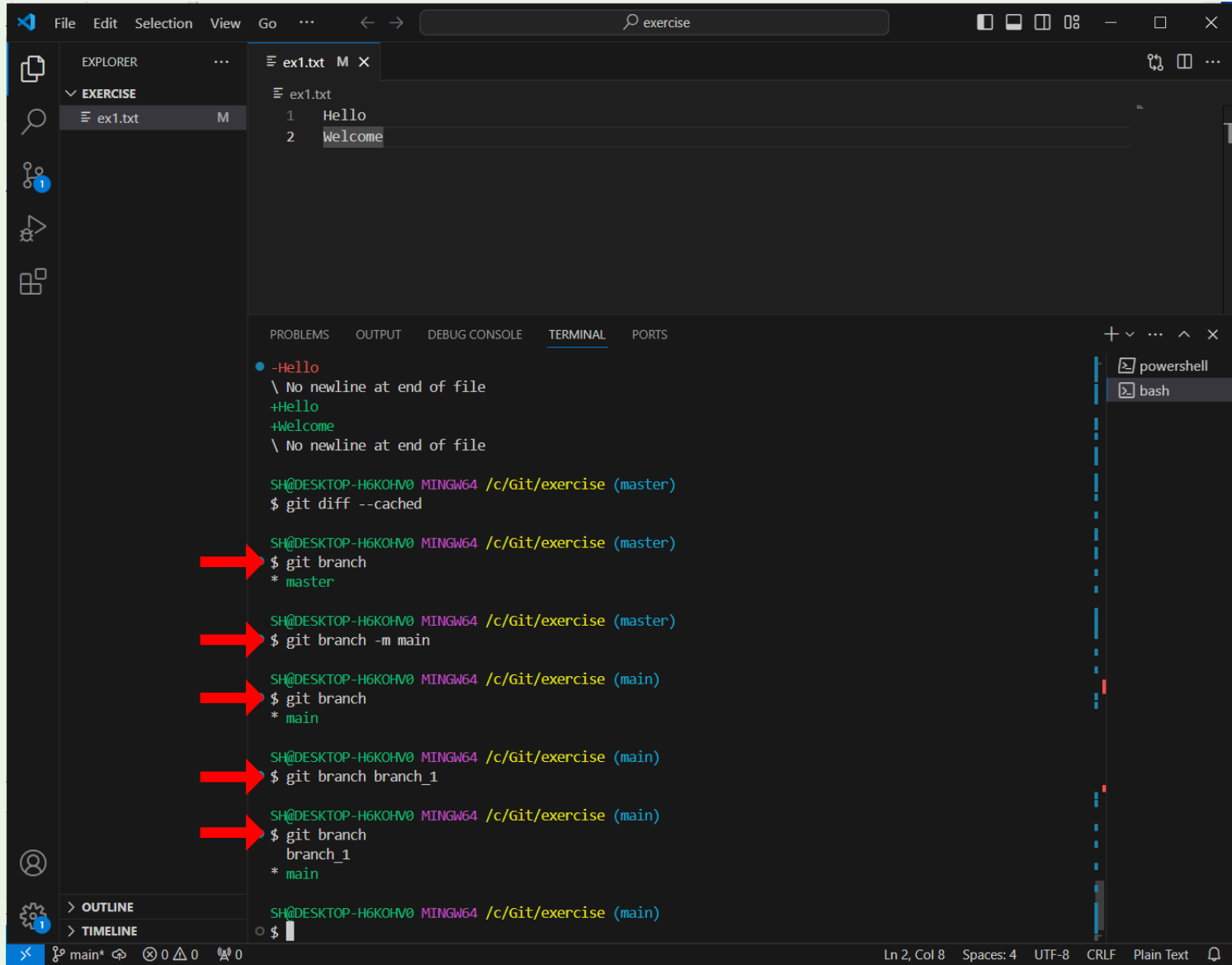
Red arrows and boxes highlight the following elements:

- A red arrow points to the `ex1.txt` file in the Explorer panel.
- A red box highlights the commit `07658c8d20d06bb11174253631717a6db9d11765` in the `git log` output, which corresponds to the commit `add first code in ex1.txt`.
- A red arrow points to the `$ git reset --hard 07658c8` command in the terminal output.
- A red arrow points to the commit `07658c8d20d06bb11174253631717a6db9d11765` in the `git log` output, which corresponds to the commit `add first code in ex1.txt`.

7. Practical Exercise

- **Check the branches in your repository**
- **Change your current branch name to “main”**
 - master -> main
- **Create a new branch named “branch_1”**

7. Practical Exercise



8. Practical Exercise

- **Switch the current branch to “branch_1”**
 - Use switch command
- **Check the history of the branch_1**
 - Check if it has the same history as its ancestor branch
- **Create and switch to a new branch named “branch_2”**
 - Use checkout command

8. Practical Exercise

```
File Edit Selection View Go ... exercise
```

EXPLORER

- EXERCISE
 - ex1.txt

ex1.txt

```
1 Hello
2 Welcome
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git switch branch_1
Switched to branch 'branch_1'

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git log
commit 68e331f181a30945c305de67433a1bb0f5a91e3a (HEAD -> branch_1, main)
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 13:15:21 2024 +0900

    add second code in ex1.txt

commit 07658c8d20d06bb11174253631717a6db9d11765
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:45:19 2024 +0900

    add first code in ex1.txt

commit 7769ca49da8fe5d4e8e1aaf5300c6dbd0623236e
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 12:09:53 2024 +0900

    add ex1.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git checkout -b branch_2
Switched to a new branch 'branch_2'

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_2)
$
```

branch_2 0 0 0

Ln 2, Col 8 Spaces: 4 UTF-8 CRLF Plain Text

9. Practical Exercise

- **Create a new file “ex2.txt” in branch “branch_1”**
- **Add a code “Hello” in the file “ex2.txt”**
- **Stage and commit the changes**
 - Commit message: “add ex2.txt”
- **Merge branch_1 into main**
 - Merge message: “merge branch_1 into main”
- **check the history of the main branch**

9. Practical Exercise

```
File Edit Selection View Go ... exercise
```

EXPLORER

- EXERCISE
 - ex1.txt
 - ex2.txt

EXERCISE

- ex1.txt
- ex2.txt

ex2.txt

```
1 Hello
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ touch ex2.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git add .

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git commit -m "add ex2.txt"
[branch_1 b9ce397] add ex2.txt
1 file changed, 1 insertion(+)
create mode 100644 ex2.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git switch main
Switched to branch 'main'

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git merge branch_1 -m "merge branch_1 into main"
Updating 68e331f..b9ce397
Fast-forward (no commit created; -m option ignored)
ex2.txt | 1 +
1 file changed, 1 insertion(+)
create mode 100644 ex2.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git log
commit b9ce39721c0ef1b0f168cdfad1e1ef792fbf8f6f (HEAD -> main, branch_1)
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 13:56:06 2024 +0900

    add ex2.txt

commit 68e331f181a30945c305de67433a1bb0f5a91e3a
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 13:15:21 2024 +0900

    add second code in ex1.txt
```

main 0 0 0

Ln 1, Col 6 Spaces: 4 UTF-8 CRLF Plain Text

10.1 Practical Exercise

- **Create a new file “main1.txt” in branch “main”**
- **Stage and commit the changes**
 - Commit message: “add main1.txt in main”
- **Create new files “br1.txt” in branch “branch_1”**
- **Stage and commit the changes**
 - Commit message: “add br1.txt in branch_1”
- **Create new files “br2.txt” in branch “branch_1”**
- **Stage and commit the changes**
 - Commit message: “add br2.txt in branch_1”

10.2 Practical Exercise

- **Expect and check the differences in the history of the main branch and the branch_1**
- **Rebase branch_1 into main, and merge it**
 - Merge message: “merge branch_1 into main”
- **Check the history of the main branch**

10.1 Practical Exercise

```
File Edit Selection View Go ... exercise
EXPLORER
EXERCISE
  br1.txt
  br2.txt
  ex1.txt
  ex2.txt
  main1.txt
  ex2.txt
    1 Hello
  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
  $ touch main1.txt
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
  $ git add .
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
  $ git commit -m "add main1.txt in main"
  [main 5459aa9] add main1.txt in main
  1 file changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 main1.txt
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
  $ git switch branch_1
  Switched to branch 'branch_1'
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
  $ touch br1.txt
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
  $ git add .
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
  $ git commit -m "add br1.txt in branch_1"
  [branch_1 b18d671] add br1.txt in branch_1
  1 file changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 br1.txt
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
  $ touch br2.txt
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
  $ git add .
  SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
  $ git commit -m "add br2.txt in branch_1"
  [branch_1 051f719] add br2.txt in branch_1
  1 file changed, 0 insertions(+), 0 deletions(-)
  > OUTLINE
  > TIMELINE
  main 0
```

10.2 Practical Exercise

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git log
commit 5459aa9860b887fcb955486d40d1cfff1a0e0cc6d (HEAD -> main)
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 14:04:46 2024 +0900

    add main1.txt in main

commit b9ce39721c0ef1b0f168cdfad1e1ef792fbf8f6f
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 13:56:06 2024 +0900

    add ex2.txt

commit 68e331f181a30945c305de67433a1bb0f5a91e3a
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 13:15:21 2024 +0900

    add second code in ex1.txt
```

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git log
commit 051f7191eadd62d1f142e69bbc7018af5bce364a (HEAD -> branch_1)
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 14:06:23 2024 +0900

    add br2.txt in branch_1

commit b18d671d2b57ac764e274091564cd923b7437181
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 14:06:05 2024 +0900

    add br1.txt in branch_1

commit b9ce39721c0ef1b0f168cdfad1e1ef792fbf8f6f
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 13:56:06 2024 +0900

    add ex2.txt

commit 68e331f181a30945c305de67433a1bb0f5a91e3a
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 13:15:21 2024 +0900

    add second code in ex1.txt
```

10.2 Practical Exercise

```
File Edit Selection View Go ... exercise
```

EXPLORER

- EXERCISE
 - br1.txt
 - br2.txt
 - ex1.txt
 - ex2.txt
 - main1.txt

ex2.txt

```
1 Hello
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git switch branch_1
Switched to branch 'branch_1'

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git rebase main
Successfully rebased and updated refs/heads/branch_1.

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (branch_1)
$ git switch main
Switched to branch 'main'

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git merge branch_1
Updating 5459aa9..dd1904f
Fast-forward
 br1.txt | 0
 br2.txt | 0
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 br1.txt
 create mode 100644 br2.txt

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$
```

powershell
bash

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git log
commit dd1904f899752d9e211bde8515c07d062313e49f (HEAD -> main, branch_1)
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 14:06:23 2024 +0900

    add br2.txt in branch_1

commit e0a6d4ee714e4f2078bc761e26ee870d249e2cd5
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 14:06:05 2024 +0900

    add br1.txt in branch_1

commit 5459aa9860b887fcb955486d40d1cfff1a0e0cc6d
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 14:04:46 2024 +0900

    add main1.txt in main
```

Ln 1, Col 6 Spaces: 4 UTF-8 CRLF Plain Text

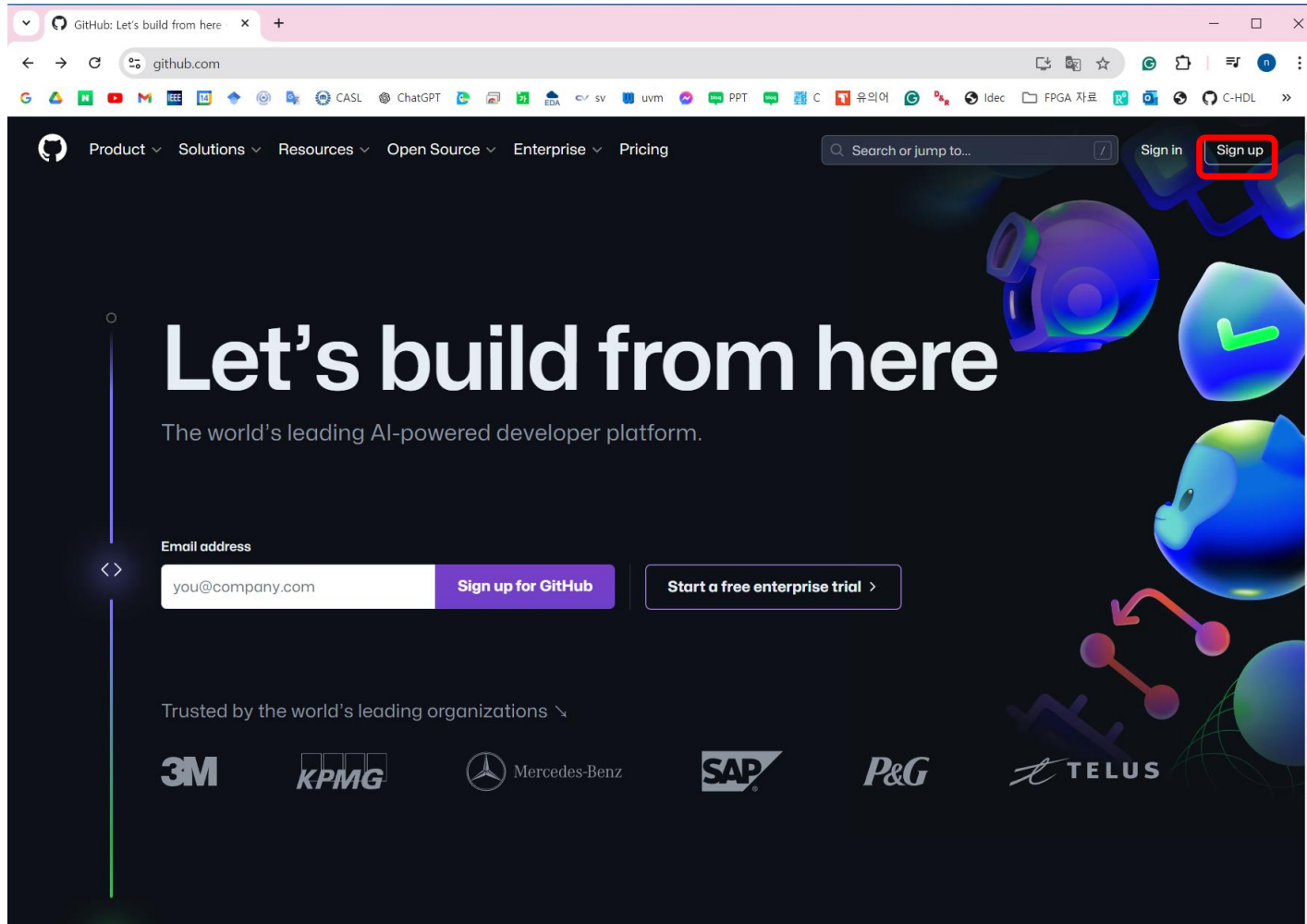
Git & Github

Github

1. Setting

■ Login your ID in GitHub

- <https://github.com/<ID>>



Setting

■ Create a new repository

The screenshot shows the GitHub homepage in a web browser. The browser's address bar displays 'github.com'. The page layout includes a top navigation bar with the GitHub logo and a search bar. The main content area is divided into several sections. On the left, there's a sidebar with the text 'Create your first project' and a red button labeled 'Create repository'. The central 'Home' section features a 'Start writing code' link and a 'Start a new repository for nrim42' card. This card contains a text input for the repository name, radio buttons for 'Public' and 'Private' visibility, and a red 'Create a new repository' button. To the right of this card is a 'Introduce yourself with a profile README' section showing a preview of a README file. Further right is an 'Explore repositories' section listing various repositories like 'jotego / jtcores' and 'kubernetes-sigs / contributor-playground'. At the bottom, there are two more cards: 'Simplify your development workflow with a GUI' and 'Get AI-based coding suggestions'.

GitHub

Dashboard

Home

Send feedback Filter

<> Start writing code

Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Create repository Import repository

or

Start a new repository for nrim42

A repository contains all of your project's files, revision history, and collaborator discussion.

Repository name *

name your new repository...

☐ Public

Anyone on the internet can see this repository

☒ Private

You choose who can see and commit to this repository

Create a new repository

Introduce yourself with a profile README

Share information about yourself by creating a profile README, which appears at the top of your profile page.

nrim42 / README.md Create

```
1 - 🙋 Hi, I'm @nrim42
2 - 🤖 I'm interested in ...
3 - 📖 I'm currently learning ...
4 - 🤝 I'm looking to collaborate on ...
5 - 📧 How to reach me ...
6 - 😊 Pronouns: ...
7 - ⚡ Fun fact: ...
8
```

Explore repositories

jotego / jtcores

FPGA cores compatible with multiple arcade game machines and KiCAD schematics of arcade games. Working on MiSTer FPGA/Analogue Pocket

☆ 209 Verilog

kubernetes-sigs / contributor-playground

☆ 193 Dockerfile

home-assistant / companion.home-assistant

Home Assistant Companion docs

☆ 83 JavaScript

Explore more →

Use tools of the trade

Simplify your development workflow with a GUI

Install GitHub Desktop to visualize, commit, and push changes without ever touching the command line.

Get AI-based coding suggestions

Try GitHub Copilot free for 30 days, which suggests entire functions in real time, right from your editor.

Setting

■ Set the repository name to “practice”

github.com/new

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * nrim42 / Repository name * practice
✔ practice is available.

Great repository names are short and memorable. Need inspiration? How about curly-meme?

Description (optional)

☐ Public
Anyone on the internet can see this repository. You choose who can commit.

☒ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore
.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license
License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

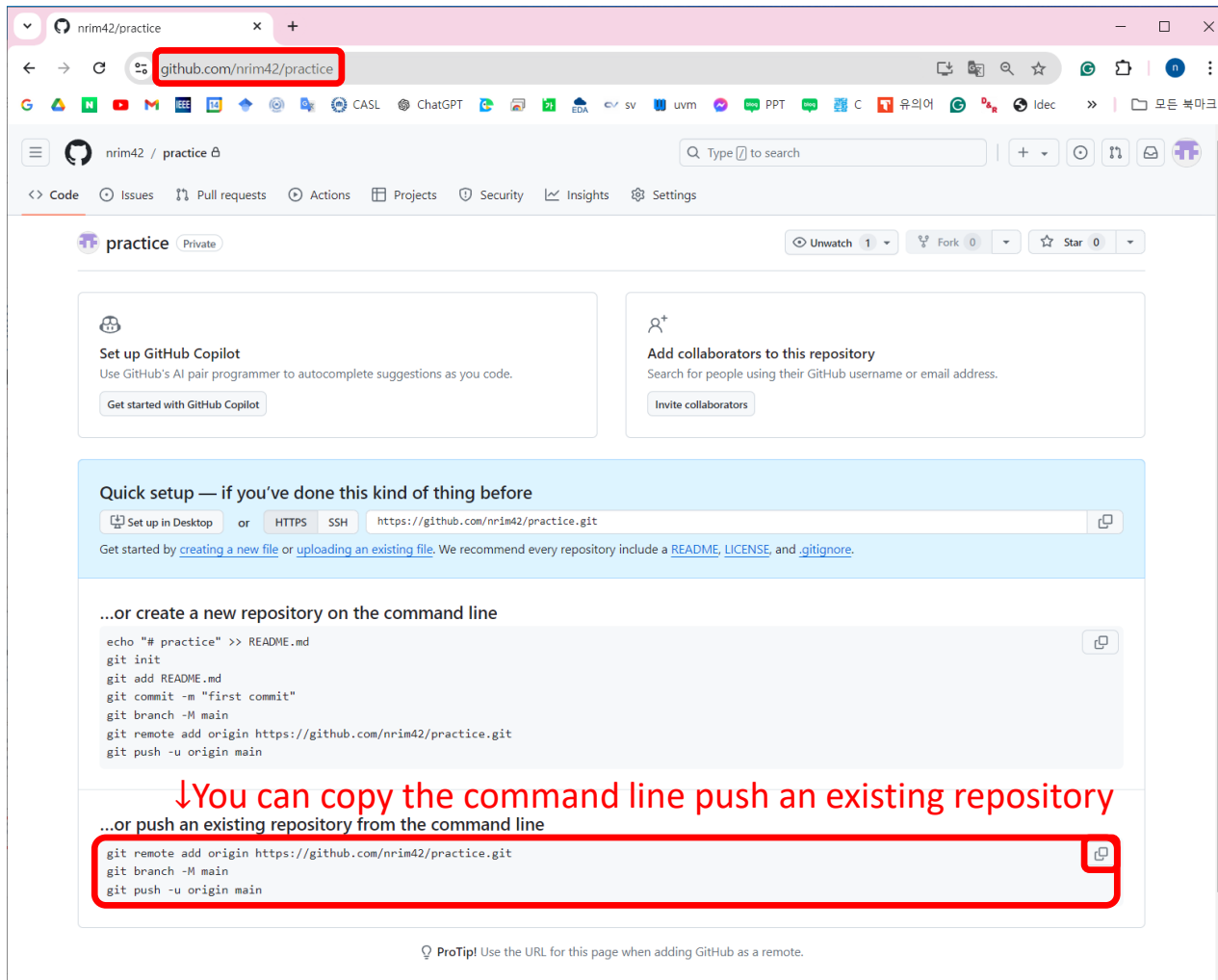
① You are creating a private repository in your personal account.

Create repository

Setting

■ Check your repository

- `https://github.com/<ID>/<practice>.git`



11. Practical Exercise

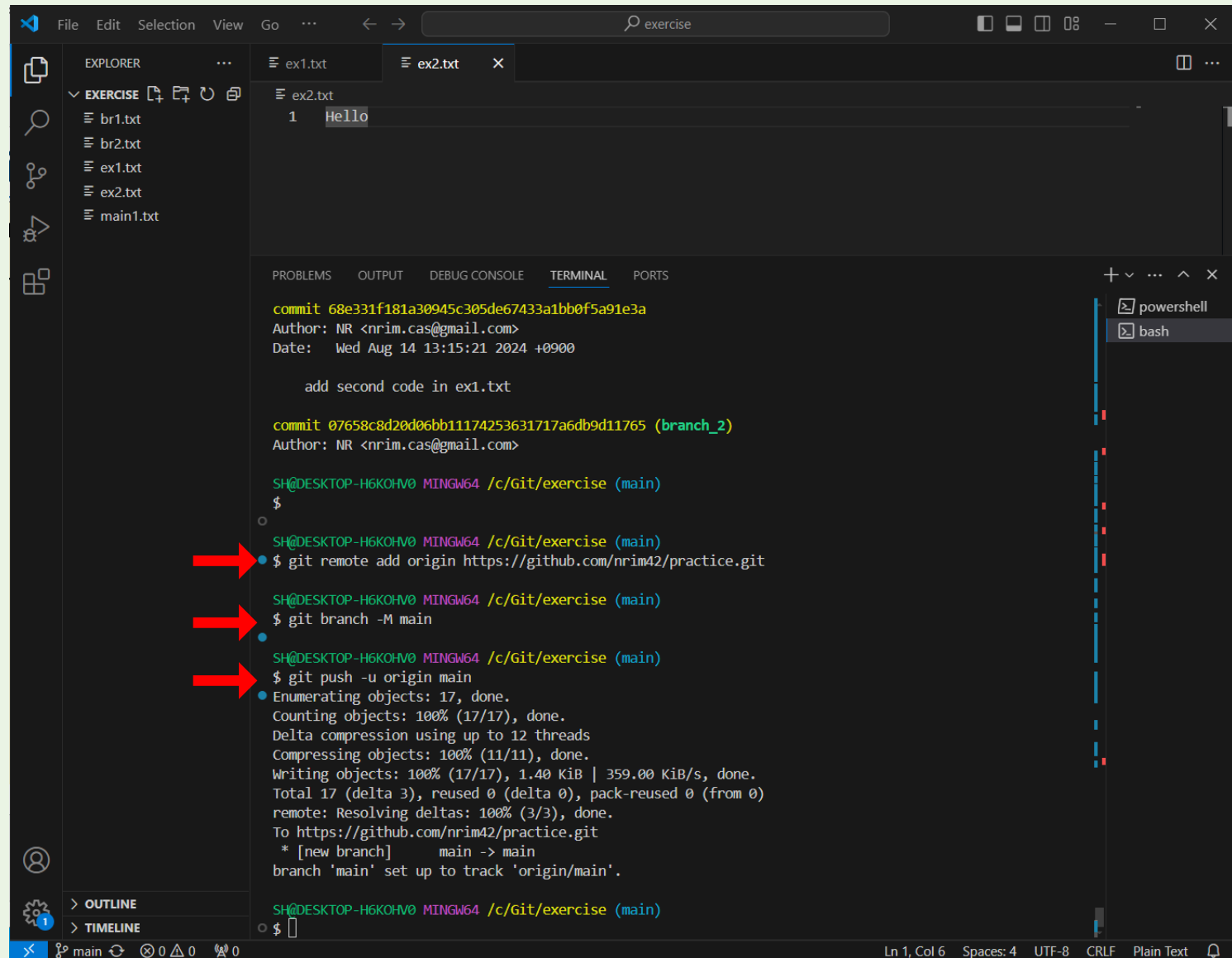
■ What do the following commands do?

- `git remote add origin https://github.com/nrim42/practice.git`
- `git branch -M main`
- `git push -u origin main`

11. Practical Exercise

- **git remote add origin <https://github.com/nrim42/practice.git>**
= Add a new remote repository (GitHub) to your local Git repository
- **git branch -M main**
= Force to change the current branch name to “main”
- **git push -u origin main**
= Push your local changes to a remote repository (GitHub) and set up tracking for the branch
* You can use “git push/pull” in the future without having to specify the branch name

11. Practical Exercise



```
commit 68e331f181a30945c305de67433a1bb0f5a91e3a
Author: NR <nrim.cas@gmail.com>
Date: Wed Aug 14 13:15:21 2024 +0900

    add second code in ex1.txt

commit 07658c8d20d06bb11174253631717a6db9d11765 (branch_2)
Author: NR <nrim.cas@gmail.com>

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$
$ git remote add origin https://github.com/nrim42/practice.git
$ git branch -M main
$ git push -u origin main
Enumerating objects: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 12 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (17/17), 1.40 KiB | 359.00 KiB/s, done.
Total 17 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/nrim42/practice.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$
```

12. Practical Exercise

- **Edit code in your main branch**
- **Upload the changes to the GitHub**

12. Practical Exercise

```
File Edit Selection View Go ... exercise
```

EXPLORER

- EXERCISE
 - br1.txt
 - br2.txt
 - ex1.txt
 - ex2.txt
 - main1.txt

main1.txt

```
1 This is a Git & Github class.
2 Thank you for attending.
3 Have a nice day.
```

TERMINAL

```
SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

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:   main1.txt

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

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git add .

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git commit -m "edit main1.txt"
[main 2b82f3a] edit main1.txt
1 file changed, 3 insertions(+)

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 317 bytes | 317.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/nrim42/practice.git
   dd1904f..2b82f3a  main -> main

SH@DESKTOP-H6KOHV0 MINGW64 /c/Git/exercise (main)
$ git log
commit 2b82f3af2d7b89af3ce87078304a06a7c77dbf53 (HEAD -> main, origin/main)
Author: NR <nrim.cas@gmail.com>
Date:   Wed Aug 14 14:54:00 2024 +0900

    edit main1.txt
```

↓GitHub

origin/main

main

Thank You