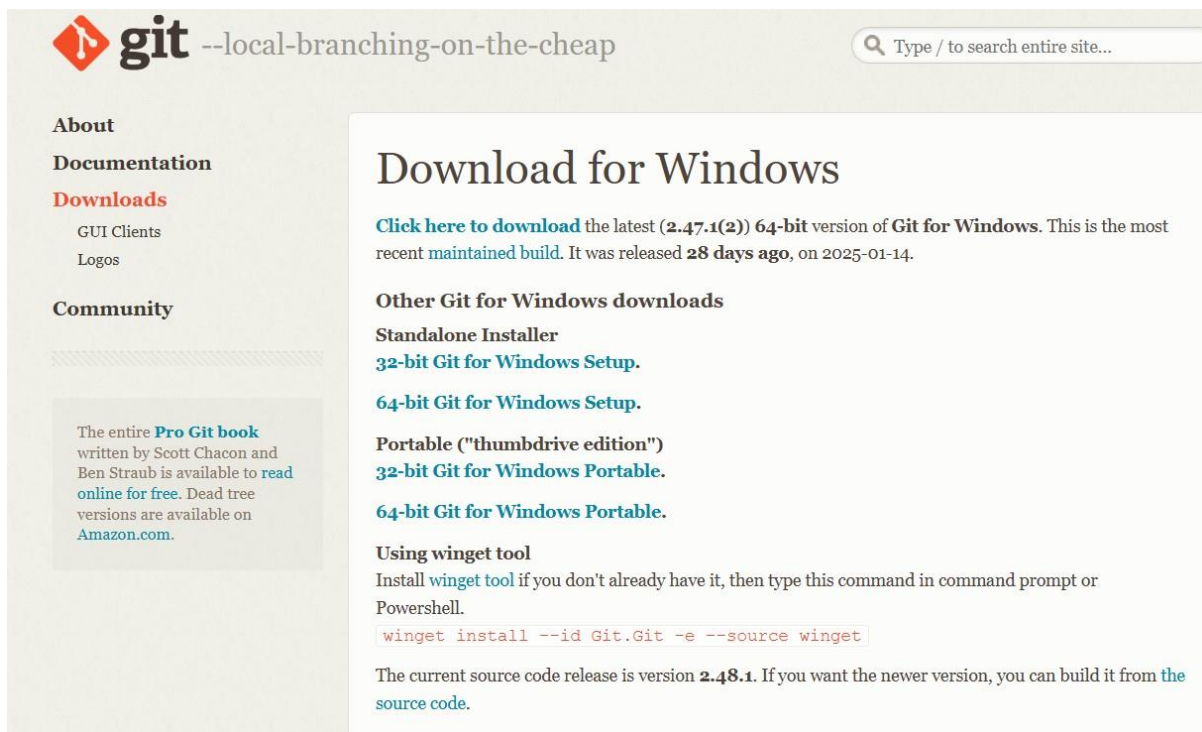


Source code Management

Lab Practical 1

1. Installing Git in Windows

Step 1: Visit section 1.5 of pro git document and navigate to Windows section



The screenshot shows the Git website's 'Download for Windows' page. The header features the Git logo and the tagline '--local-branching-on-the-cheap'. A search bar is located in the top right corner. The left sidebar contains links for 'About', 'Documentation', 'Downloads' (highlighted), 'GUI Clients', 'Logos', and 'Community'. The main content area is titled 'Download for Windows' and includes a link to download the latest (2.47.1(2)) 64-bit version of Git for Windows. It also lists other Git for Windows downloads, including standalone installers and portable versions for 32-bit and 64-bit systems. A section titled 'Using winget tool' provides instructions on how to install Git using the winget command. The footer mentions the current source code release is version 2.48.1 and provides a link to the source code.

git --local-branching-on-the-cheap

Type / to search entire site...

About

Documentation

Downloads

GUI Clients

Logos

Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Download for Windows

[Click here to download](#) the latest (**2.47.1(2)**) **64-bit** version of **Git for Windows**. This is the most recent [maintained build](#). It was released **28 days ago**, on 2025-01-14.

Other Git for Windows downloads

Standalone Installer

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

Portable ("thumbdrive edition")

[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

```
winget install --id Git.Git -e --source winget
```

The current source code release is version **2.48.1**. If you want the newer version, you can build it from [the source code](#).

Step 2: Verify Git Installation:

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ git --version
git version 2.49.0.windows.1
```

p

2. Basic CLI Commands

1) Command: pwd

Description: Prints the directory the user is working in.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ pwd
/c/source_project/kushal
```

2) Command: ls

Description: Lists all files and directories in the current directory

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ ls
'$AV_ASW'/'  AiOLog.txt          DumpStack.log      Intel/      MagniPacks/  'Program Files (x86)'/  Recovery/      Windows/  pagefile.sys
'$Recycle.Bin/'  Config.Msi/          DumpStack.log.tmp  KMPPlayer/  PerfLogs/    ProgramData/          'System Volume Information'/  hiberfil.sys  source_project/
'$WinREAgent/'  'Documents and Settings'@  'GOG Games'/      MSOCache/   'Program Files'/  'R.G. Catalyst'/      Users/          inetpub/    swapfile.sys
```

3. Command: date

Description: shows the current date and time in a standard format

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ date
Wed Jun  4 22:41:54 IST 2025
```

4. Command: clear

Description: The `clear` command in the CLI is used to clear all the current text and output displayed in the terminal window.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ date
Wed Jun  4 22:41:54 IST 2025

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ clear|
```

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$
```

5. Command: time

Description: The `time` command in the CLI is used to measure the execution time of a command or program.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ time

real    0m0.003s
user    0m0.000s
sys     0m0.000s
```

6. Command: cd 'Directory'

Description: Changes the current working directory to the desired directory.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ cd source_project

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project
$ cd kushal

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$
```

7. Command: cd ..

Description: Goes back to the previous directory.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ cd source_project

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project
$ cd kushal

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ cd ..

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project
$
```

8. Command: mkdir

Description: To create a new directory.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ mkdir labfile

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ ls
'$AV_ASW'/' Config.Msi/' 'GOG Games'/' MagiPacks/' ProgramData/' Users/' labfile/
'$Recycle.Bin'/' 'Documents and Settings'@ Intel/' PerfLogs/' 'R.G. Catalyst'/' Windows/' pagefile.sys
'$WinREAgent'/' DumpStack.log KMPlayer/' 'Program Files'/' Recovery/' hiberfil.sys source_project/
AiOLog.txt DumpStack.log.tmp MSOCache/' 'Program Files (x86)/' 'System Volume Information'/' inetpub/' swapfile.sys

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$
```

9. Command: rmdir

Description: To delete a directory

```

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ mkdir labfile

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ ls
'$AV_ASW'/'      Config.Msi/'      'GOG Games'/'    MagiPacks/'      ProgramData/'      Users/'      labfile/
'$Recycle.Bin'/'  'Documents and Settings'@ Intel/'      PerfLogs/'      'R.G. Catalyst'/'  Windows/'      pagefile.sys
'$WinREAgent'/'   DumpStack.log     KMPlayer/'      'Program Files'/'  Recovery/'         hiberfil.sys   source_project/
AiOlog.txt       DumpStack.log.tmp  MSOCache/'      'Program Files (x86)'/' 'System Volume Information'/'  inetpub/       swapfile.sys

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ rmdir labfile

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ ls
'$AV_ASW'/'      AiOlog.txt       DumpStack.log     Intel/'      MagiPacks/'      'Program Files (x86)'/'  Recovery/'      Windows/'      pagefile.sys
'$Recycle.Bin'/'  Config.Msi/'      DumpStack.log.tmp  KMPlayer/'   PerfLogs/'      ProgramData/'         'System Volume Information'/'  hiberfil.sys   source_project/
'$WinREAgent'/'   'Documents and Settings'@ 'GOG Games'/'    MSOCache/'   'Program Files'/'  'R.G. Catalyst'/'      Users/'         inetpub/       swapfile.sys

DELL@DESKTOP-95JJ6V8 MINGW64 /c
$ |

```

3. Vim Text Editor

1) Command: vi hi.txt

Description: Opens (or creates) the file `hi.txt` in the Vim text editor.

```

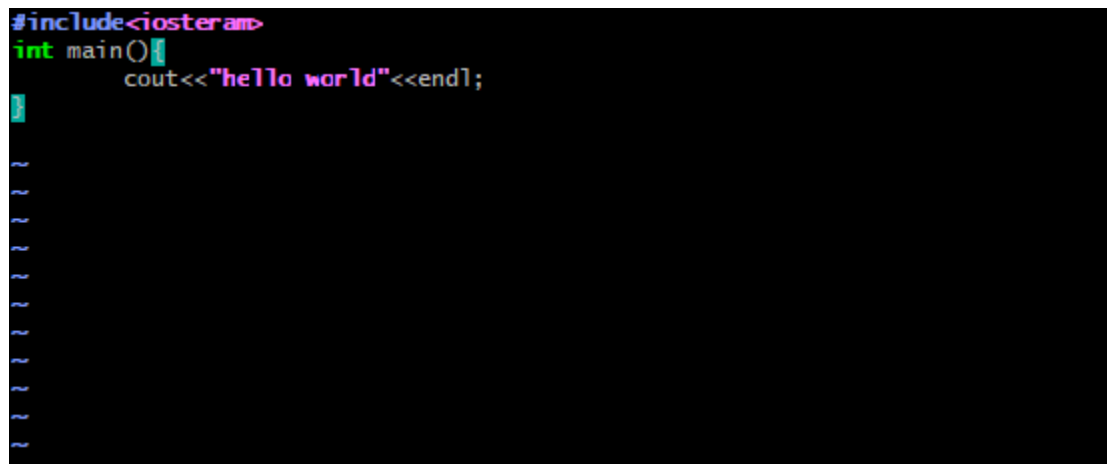
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile
$ vi hello.cpp

```



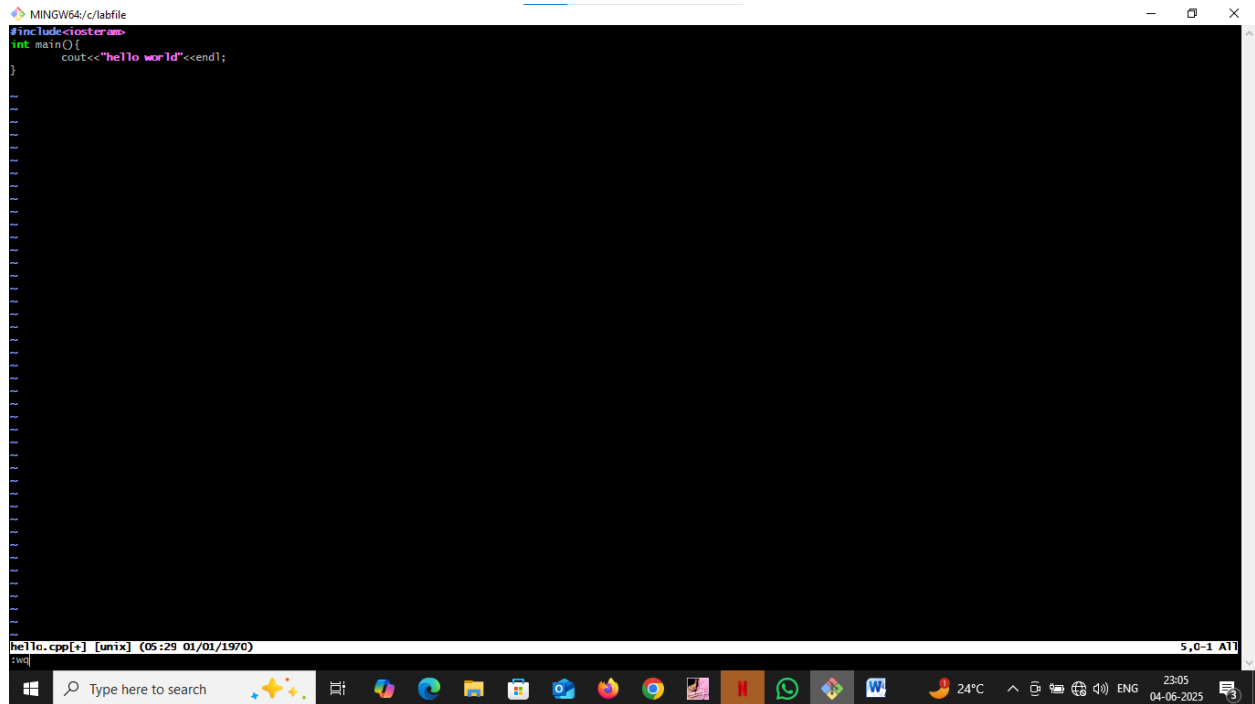
2) Command: `i` (Insert Mode)

Description: Enters insert mode in Vim to allow text input.



3) Command: esc

Description: Used to exit insert mode



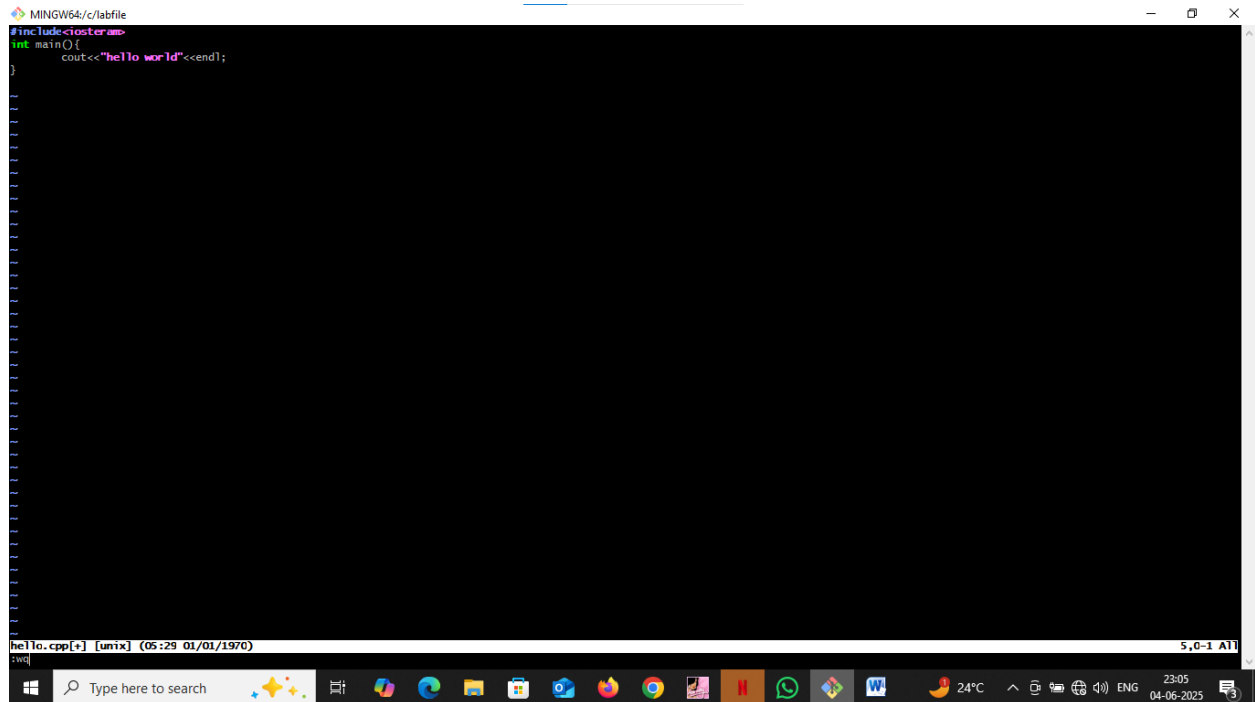
The screenshot shows a Windows terminal window titled "MINGW64/c/labfile". Inside the terminal, a C++ program is being edited in Vim. The code is as follows:

```
#include <iostream>
int main() {
    cout << "hello world" << endl;
}
```

The Vim status bar at the bottom of the terminal window displays "hello.cpp[*] [unix] (05:29 01/01/1970) 5,0-1 All". The Windows taskbar is visible at the bottom of the screen, showing various application icons and the system clock indicating 23:05 on 04-06-2025.

4) Command: :wq

Description: Saves the changes and exits the Vim editor.

A screenshot of a MinGW64 terminal window. The title bar reads 'MINGW64/c/labfile'. The terminal shows a C++ program with the following code:

```
#include <iostream>
int main() {
    cout << "hello world" << endl;
}
```

The status bar at the bottom of the terminal window indicates 'hello.cpp[*] [unix] (05:29 01/01/1970)' and '5,0-1 All'. Below the terminal window is a Windows taskbar with a search bar and various application icons. The system tray on the right shows the temperature as 24°C, the time as 23:05, and the date as 04-06-2025.

5. Command: git init

Description: Initializes a new Git repository in the current directory.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile
$ vi hello.cpp

DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile
$ git init
Initialized empty Git repository in C:/labfile/.git/

DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$
```

3. Command: git status

Description: Displays the current status of the working directory and staging area.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

DELL@DESKTOP-95JJ6V8 MINGW64 /c/source_project/kushal (master)
$
```

4. Command: git add Test.c

Description: Add Test.c to the staging area in preparation for a commit.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git add .
warning: in the working copy of 'hello.cpp', LF will be replaced by CRLF the next time Git touches it
```

5. Command: git commit -m "add file one"

Description: Commits the stage changes with the message "add file one".

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git add .
warning: in the working copy of 'hello.cpp', LF will be replaced by CRLF the next time Git touches it

DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git commit -m "initial commit"
[master (root-commit) dc6f155] initial commit
1 file changed, 5 insertions(+)
create mode 100644 hello.cpp
```

6. Command: git log

Description: Display the commit history of the repository.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git log
commit dc6f15588f55a769fa6e7eee8d41180fc3d1eed8 (HEAD -> master)
Author: kushal-reddy19 <kushalreddy.ks@es.amity.edu>
Date:   Wed Jun 4 23:15:59 2025 +0530

    initial commit
```

7. Command: git clone

Description: To obtain a copy of an existing Git repository.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git clone https://github.com/kushal-reddy19/hello
Cloning into 'hello'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 13 (delta 3), reused 10 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (13/13), done.
Resolving deltas: 100% (3/3), done.
```

8. Command: git log --oneline

Description: For generating shorter commit ID.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git log --oneline
dc6f155 (HEAD -> master) initial commit
```

9. Command: git diff

Description: To compare two files.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git log --oneline
16a0963 (HEAD -> master) second commit
dc6f155 initial commit

DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git diff 16a0963 dc6f155
diff --git a/he.cpp b/he.cpp
deleted file mode 100644
index 8c9380f..0000000
--- a/he.cpp
+++ /dev/null
@@ -1,2 +0,0 @@
-hi
-iam kushal
diff --git a/hello b/hello
deleted file mode 160000
index 7521200..0000000
--- a/hello
+++ /dev/null
@@ -1 +0,0 @@
-Subproject commit 7521200c815cf726ef3b9e649bc7f6bf74a27c2
```

10. Command: git remote add "Variable"

Description: To connect with the Users GitHub account.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git remote add origin https://github.com/kushal-reddy19/sample
```

11. Command: git remote

Description: To check the status of the repositories connected with the Users account.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git remote
origin
```

12. Command: git push -u "Variable" master

Description: To push all the files to the Users account.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 565 bytes | 188.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/kushal-reddy19/sample
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

13. Command: git merge "File_Name" -m "comment"

Description: To merge a branch with main branch.

```
DELL@DESKTOP-95JJ6V8 MINGW64 /c/labfile (master)
$ git merge test|
```

