LAB 1

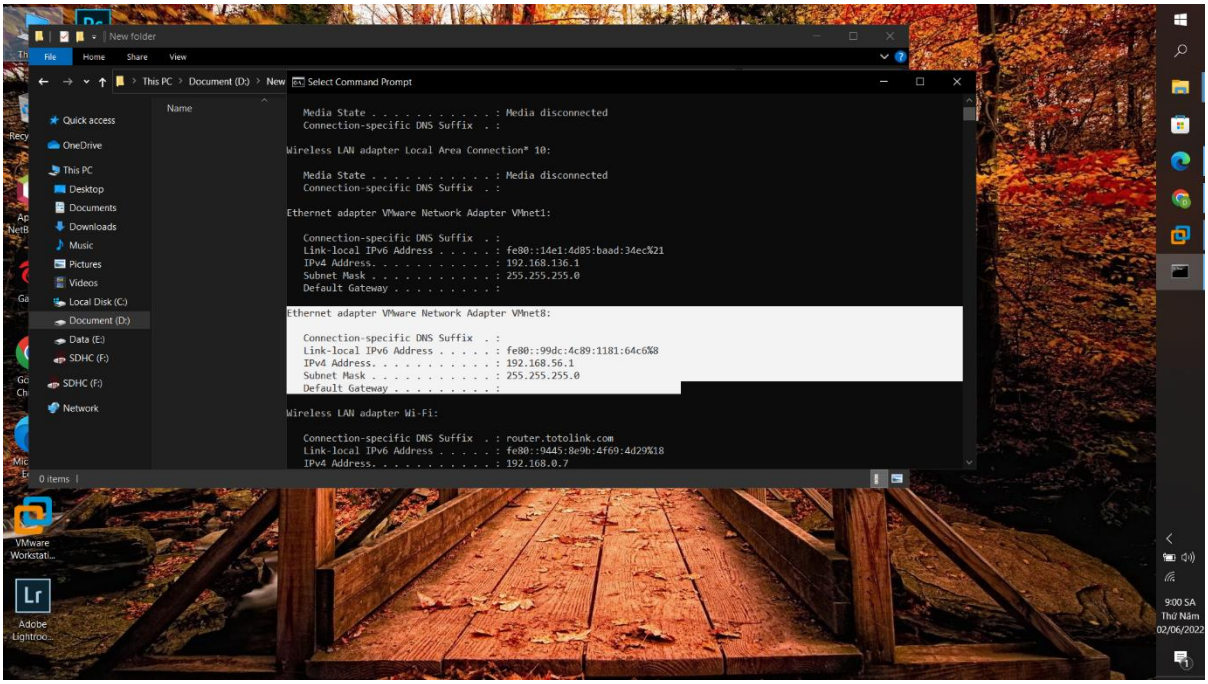
Subject: OSG202

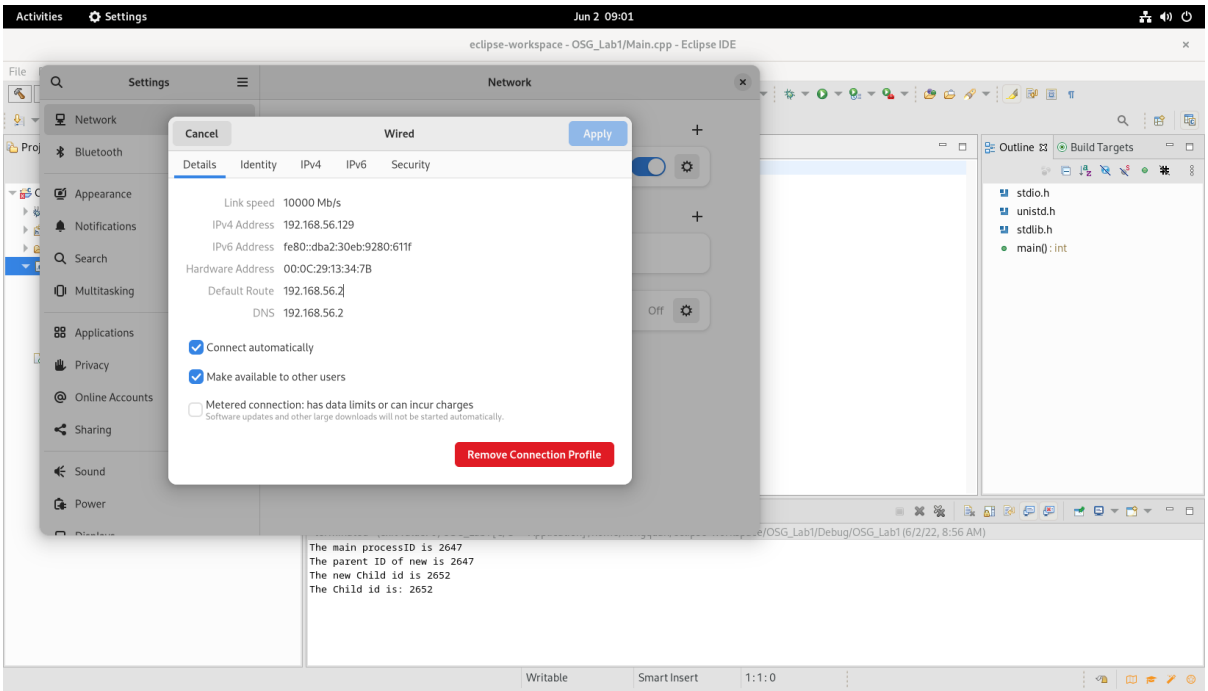
Teacher : Nguyễn Tấn Phúc

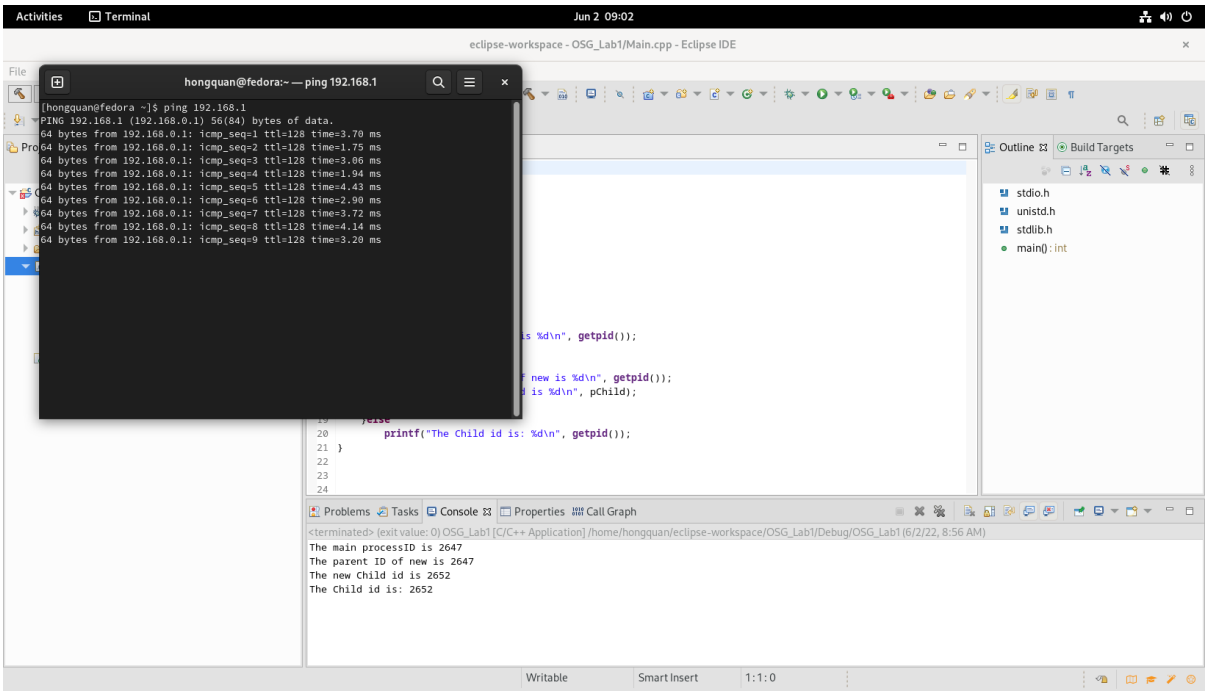
Student: Dương Hồng Quân

Student ID: SE170057

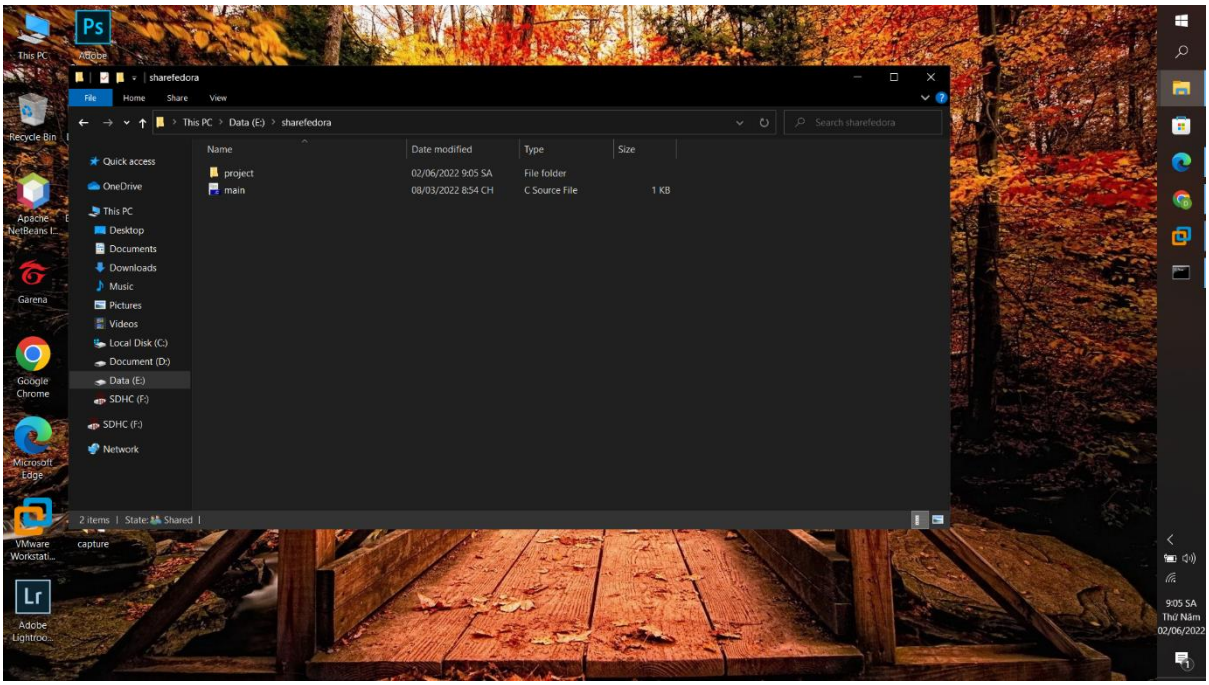
• The terminal screen when using the ifconfig command in Fedora and the ipconfig command in command prompt in Windows

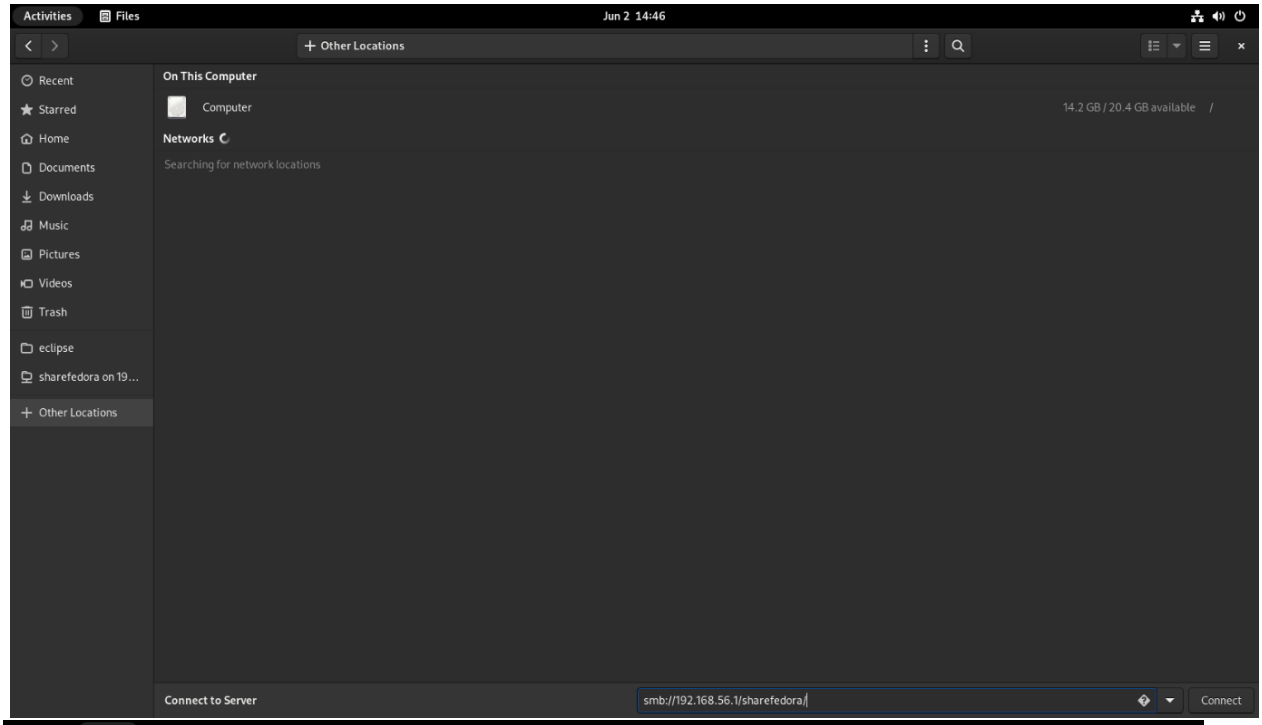


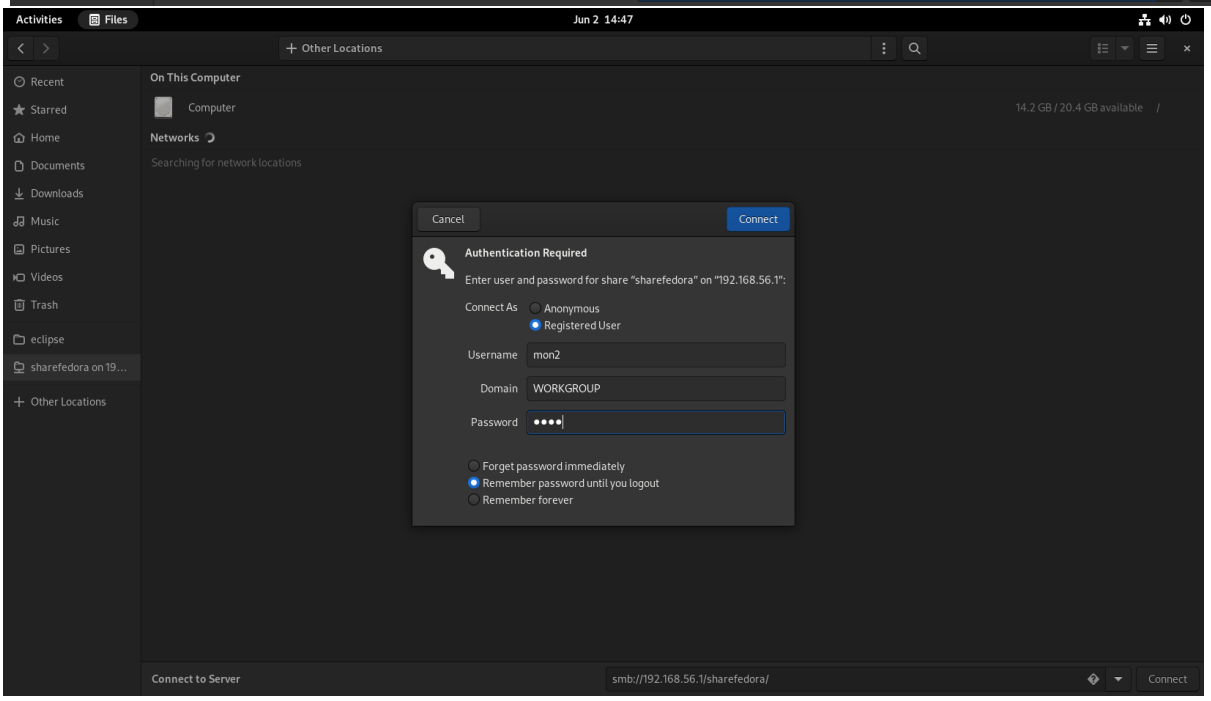


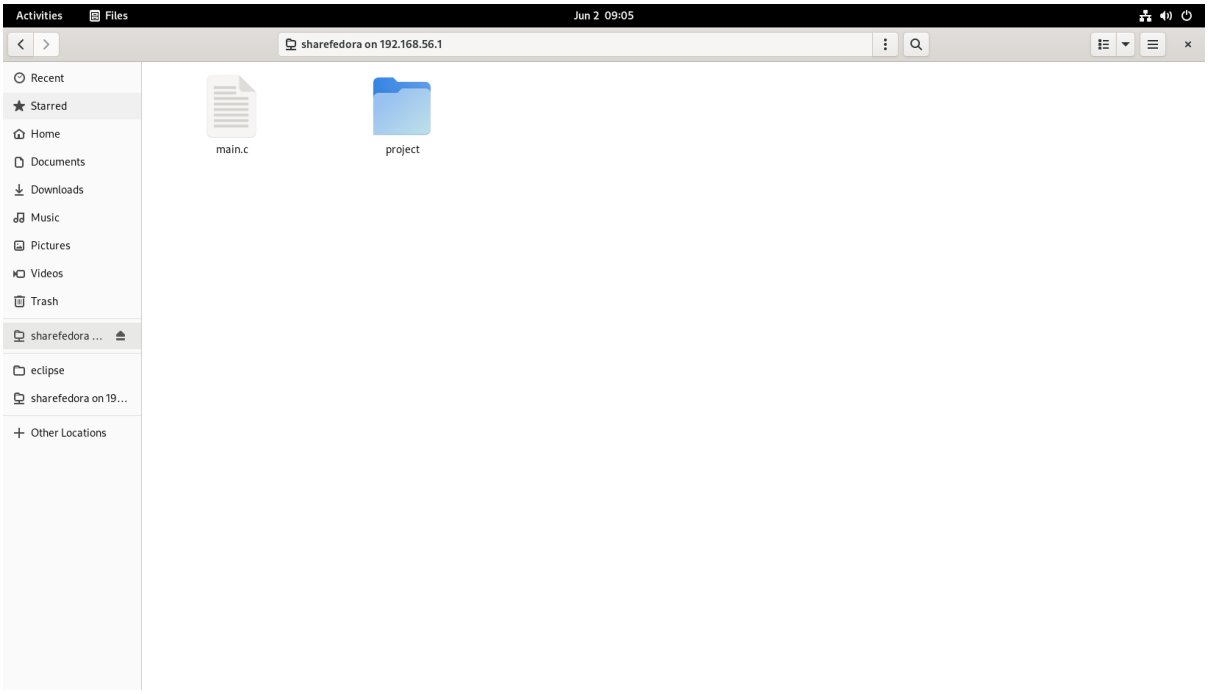


• The shared windows when connectting to the windows OS's shared folder/disk in your machine (using the File Browser with Bookmark). The shared folder must have data files and sub folder)

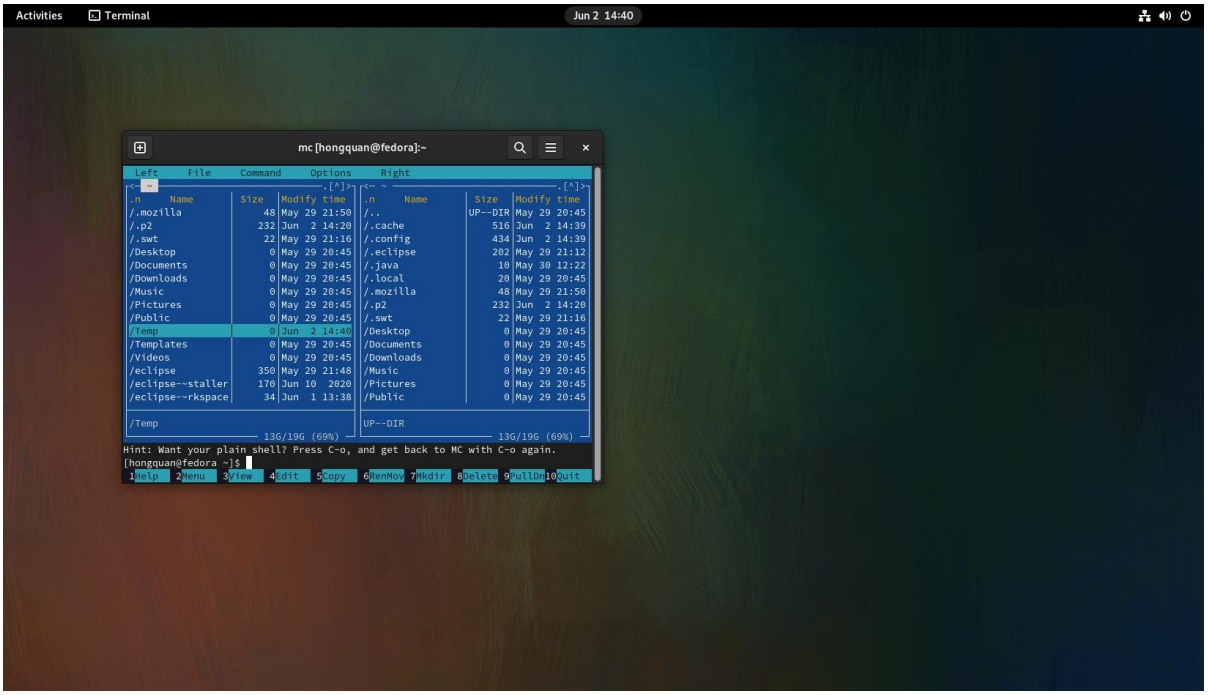



Activities
Files
Jun 2 14:46
Other Locations
Recent
Starred
Home
Documents
Downloads
Music
Pictures
Videos
Trash
eclipse
sharefedora on 19...
Other Locations
On This Computer
Computer
14.2 GB / 20.4 GB available /
Networks
Searching for network locations
Connect to Server
smb://192.168.56.1/sharefedora/
Connect


Activities
Files
Jun 2 14:47
Other Locations
Recent
Starred
Home
Documents
Downloads
Music
Pictures
Videos
Trash
eclipse
sharefedora on 19...
Other Locations
On This Computer
Computer
14.2 GB / 20.4 GB available /
Networks
Searching for network locations
Cancel
Connect
Authentication Required
Enter user and password for share "sharefedora" on "192.168.56.1":
Connect As
Anonymous
Registered User
Username
mon2
Domain
WORKGROUP
Password
Forget password immediately
Remember password until you logout
Remember forever
Connect to Server
smb://192.168.56.1/sharefedora/
Connect



- **The mc screen about the F7 function in mc**

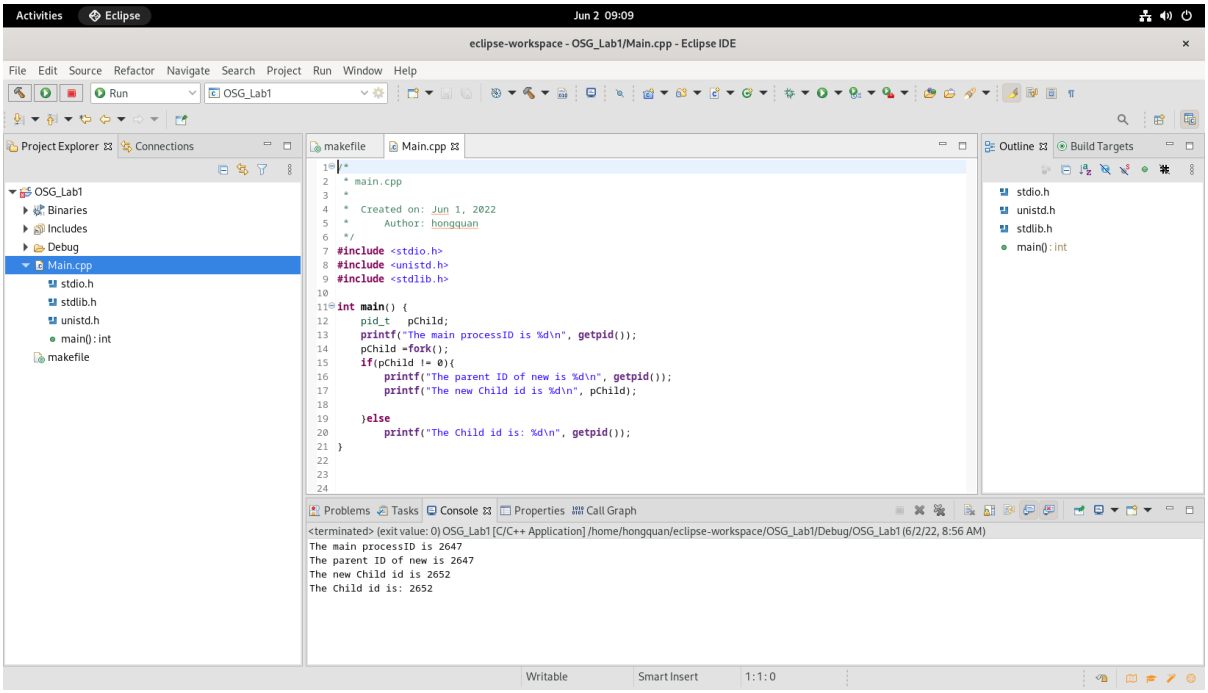


Explain:

F7 funtion in mc : makes a new directory in the directory open in the current panel. Creating new directories is simple using the mkdir command in Linux/Unix. Make Directory (mkdir) is the abbreviation for "create directory." More than just a single directory may be created at a time using mkdir, which also

allows you to manage permissions.

- **Submit the C/C++ files and objects (*.c and *.o) as the below content then capture the result of the program, give your explanation the result of this program**



Explain : Using the fork() function, the main() process is copied to a new process with pChild = 0 and a "else" decision that prints the ID of the new process and returns to the main() process (). The fork() library call with pChild = 1 continues to be executed, resulting in the "if" decision and the printing of two lines (Parent ID and new child ID)