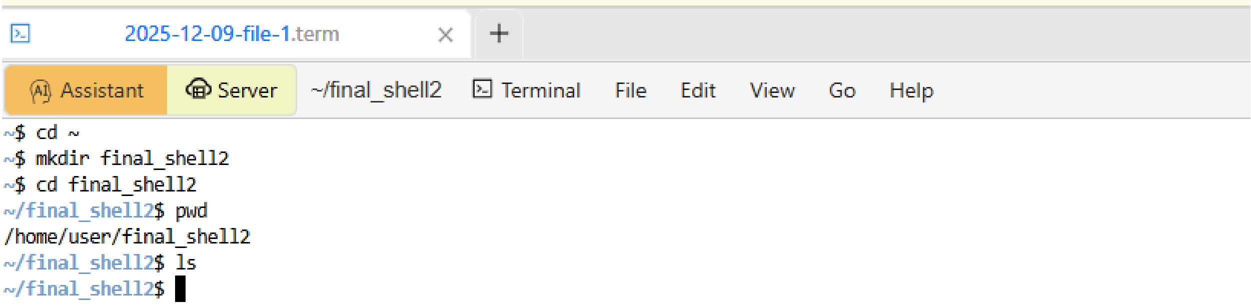
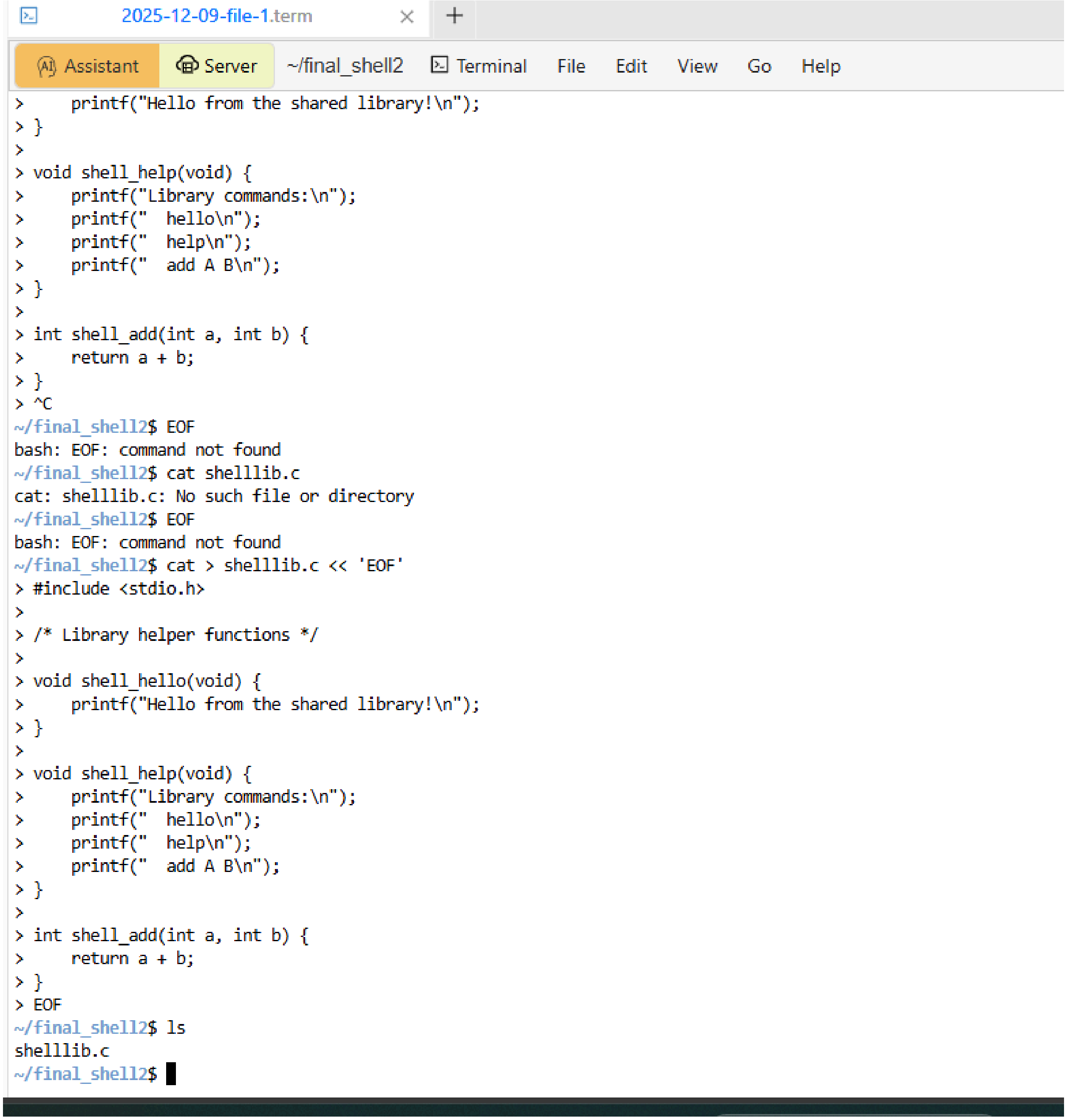


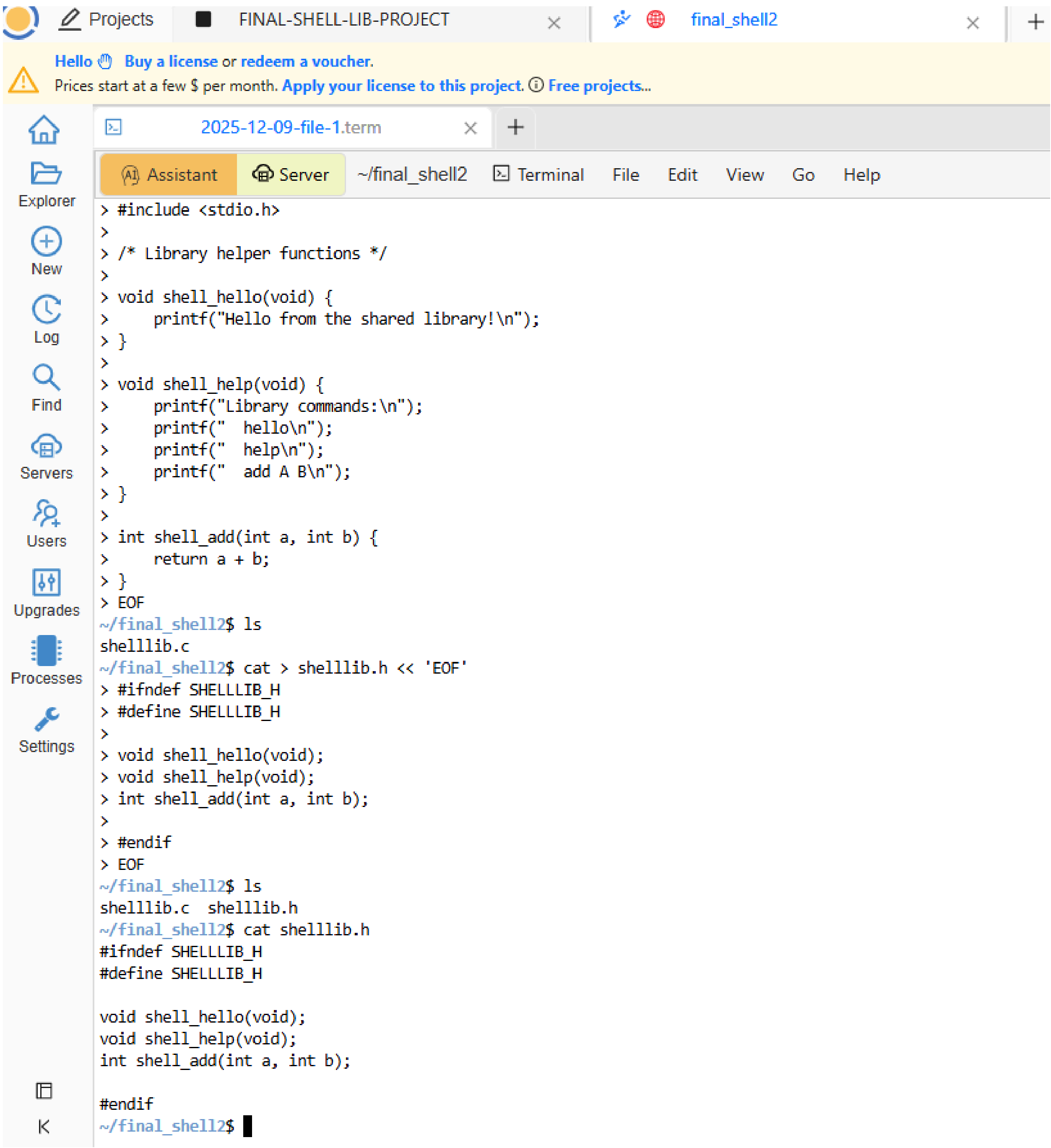
```
~$ cd ~
~$ mkdir final_shell2
~$ cd final_shell2
~/final_shell2$ pwd
/home/user/final_shell2
~/final_shell2$ ls
~/final_shell2$ 
```

New empty final_shell2 directory created.



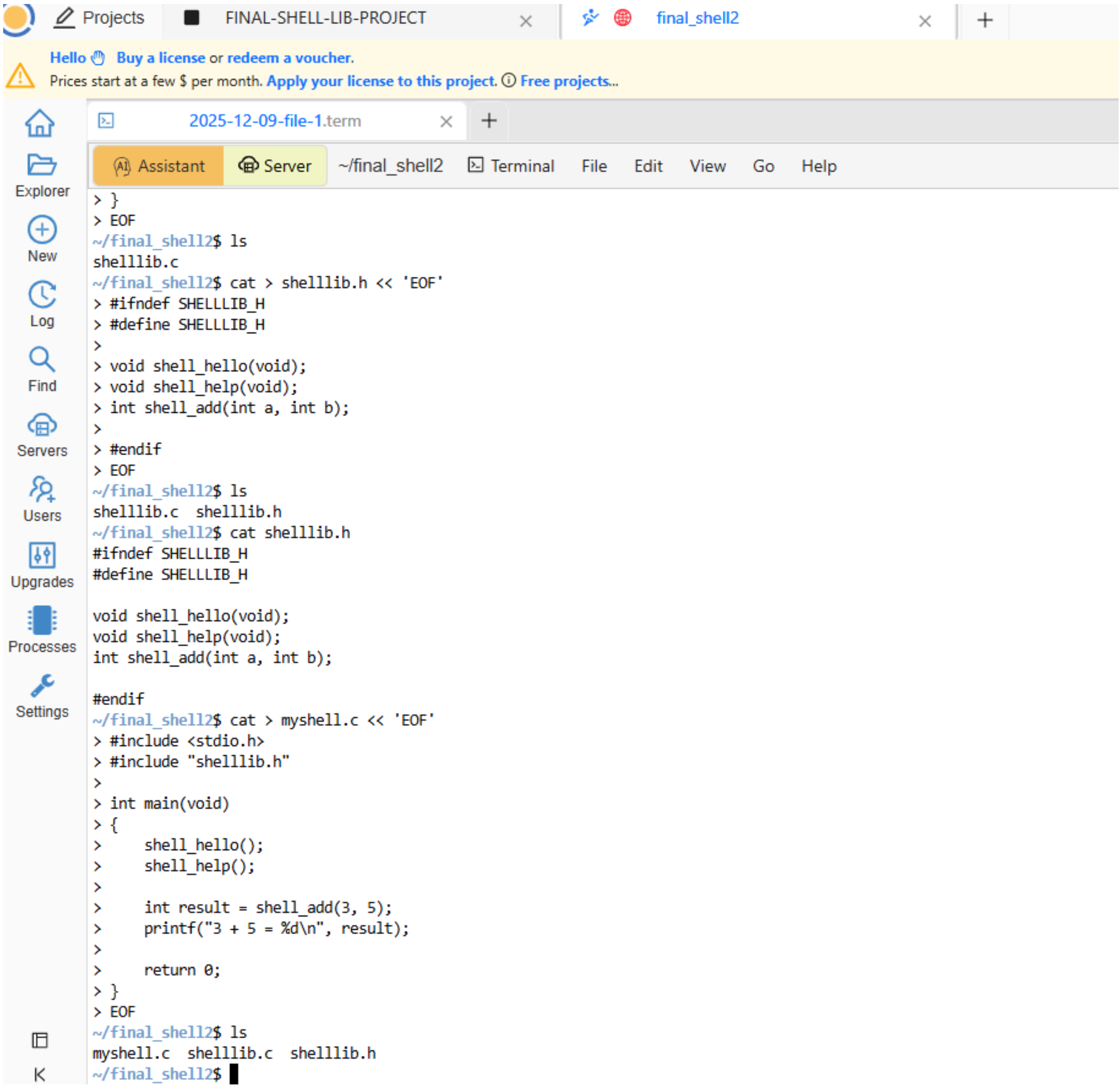
```
>     printf("Hello from the shared library!\n");
> }
>
> void shell_help(void) {
>     printf("Library commands:\n");
>     printf("  hello\n");
>     printf("  help\n");
>     printf("  add A B\n");
> }
>
> int shell_add(int a, int b) {
>     return a + b;
> }
> ^C
~/final_shell2$ EOF
bash: EOF: command not found
~/final_shell2$ cat shelllib.c
cat: shelllib.c: No such file or directory
~/final_shell2$ EOF
bash: EOF: command not found
~/final_shell2$ cat > shelllib.c << 'EOF'
> #include <stdio.h>
>
> /* Library helper functions */
>
> void shell_hello(void) {
>     printf("Hello from the shared library!\n");
> }
>
> void shell_help(void) {
>     printf("Library commands:\n");
>     printf("  hello\n");
>     printf("  help\n");
>     printf("  add A B\n");
> }
>
> int shell_add(int a, int b) {
>     return a + b;
> }
> EOF
~/final_shell2$ ls
shelllib.c
~/final_shell2$ █
```

Created shelllib.c library source file using cat << EOF.
Verified the file exists with the ls command.

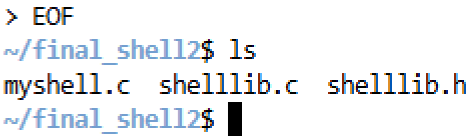


```
> #include <stdio.h>
>
> /* Library helper functions */
>
> void shell_hello(void) {
>     printf("Hello from the shared library!\n");
> }
>
> void shell_help(void) {
>     printf("Library commands:\n");
>     printf("  hello\n");
>     printf("  help\n");
>     printf("  add A B\n");
> }
>
> int shell_add(int a, int b) {
>     return a + b;
> }
> EOF
~/final_shell2$ ls
shelllib.c
~/final_shell2$ cat > shelllib.h << 'EOF'
> #ifndef SHELLLIB_H
> #define SHELLLIB_H
>
> void shell_hello(void);
> void shell_help(void);
> int shell_add(int a, int b);
>
> #endif
> EOF
~/final_shell2$ ls
shelllib.c  shelllib.h
~/final_shell2$ cat shelllib.h
#ifndef SHELLLIB_H
#define SHELLLIB_H

void shell_hello(void);
void shell_help(void);
int shell_add(int a, int b);

#endif
~/final_shell2$
```

Created the shelllib.h header file using cat << EOF.
Verified that both shelllib.c and shelllib.h exist in the final_shell2 directory.

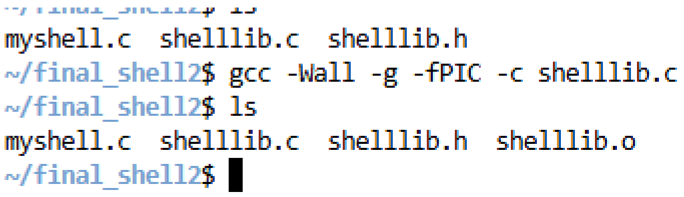


```
> }
> EOF
~/final_shell2$ ls
shelllib.c
~/final_shell2$ cat > shelllib.h << 'EOF'
> #ifndef SHELLLIB_H
> #define SHELLLIB_H
>
> void shell_hello(void);
> void shell_help(void);
> int shell_add(int a, int b);
>
> #endif
> EOF
~/final_shell2$ ls
shelllib.c  shelllib.h
~/final_shell2$ cat shelllib.h
#ifndef SHELLLIB_H
#define SHELLLIB_H

void shell_hello(void);
void shell_help(void);
int shell_add(int a, int b);

#endif
~/final_shell2$ cat > myshell.c << 'EOF'
> #include <stdio.h>
> #include "shelllib.h"
>
> int main(void)
> {
>     shell_hello();
>     shell_help();
>
>     int result = shell_add(3, 5);
>     printf("3 + 5 = %d\n", result);
>
>     return 0;
> }
> EOF
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h
~/final_shell2$ 
```

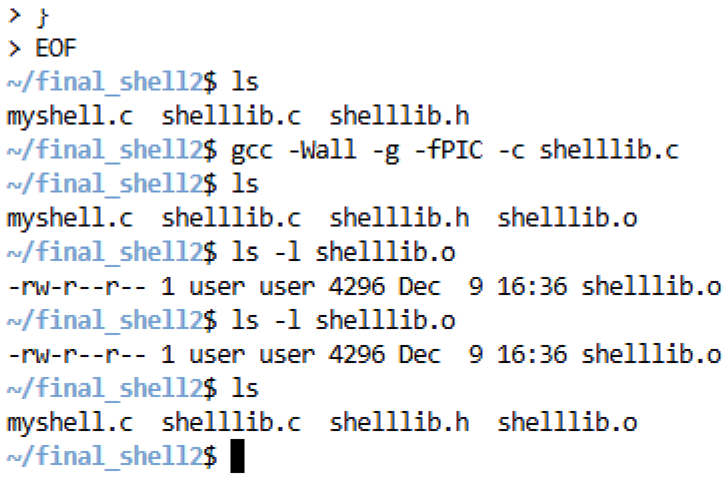


```
> EOF
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h
~/final_shell2$ 
```

Created myshell.c using cat << EOF.
Verified the file by listing the directory and printing its contents.

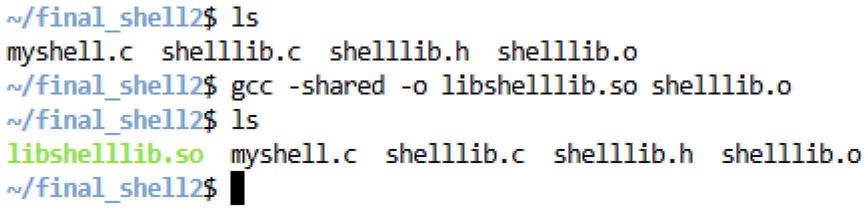


```
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h
~/final_shell2$ gcc -Wall -g -fPIC -c shelllib.c
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$
```



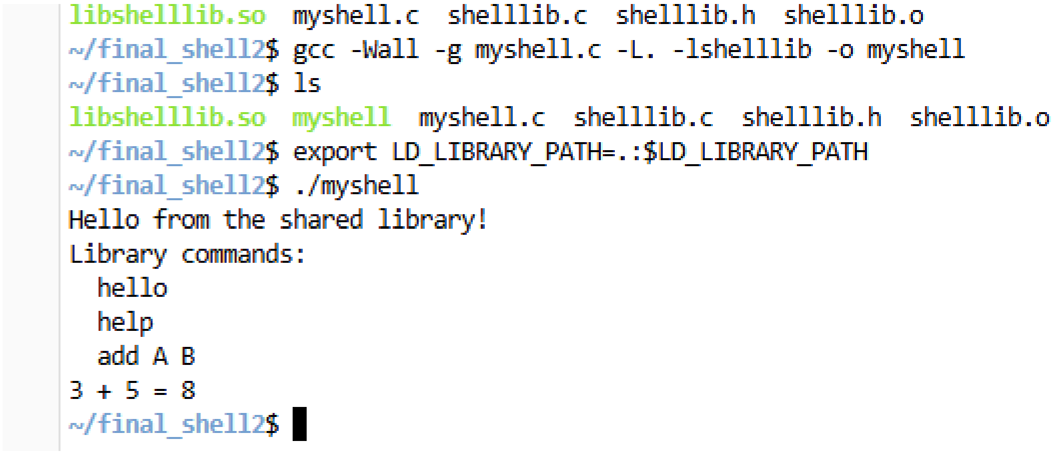
```
> }
> EOF
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h
~/final_shell2$ gcc -Wall -g -fPIC -c shelllib.c
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ ls -l shelllib.o
-rw-r--r-- 1 user user 4296 Dec  9 16:36 shelllib.o
~/final_shell2$ ls -l shelllib.o
-rw-r--r-- 1 user user 4296 Dec  9 16:36 shelllib.o
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$
```

Compiled shelllib.c into shelllib.o using gcc with -fPIC.
Verified object file creation with ls.



```
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ gcc -shared -o libshelllib.so shelllib.o
~/final_shell2$ ls
libshelllib.so  myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$
```

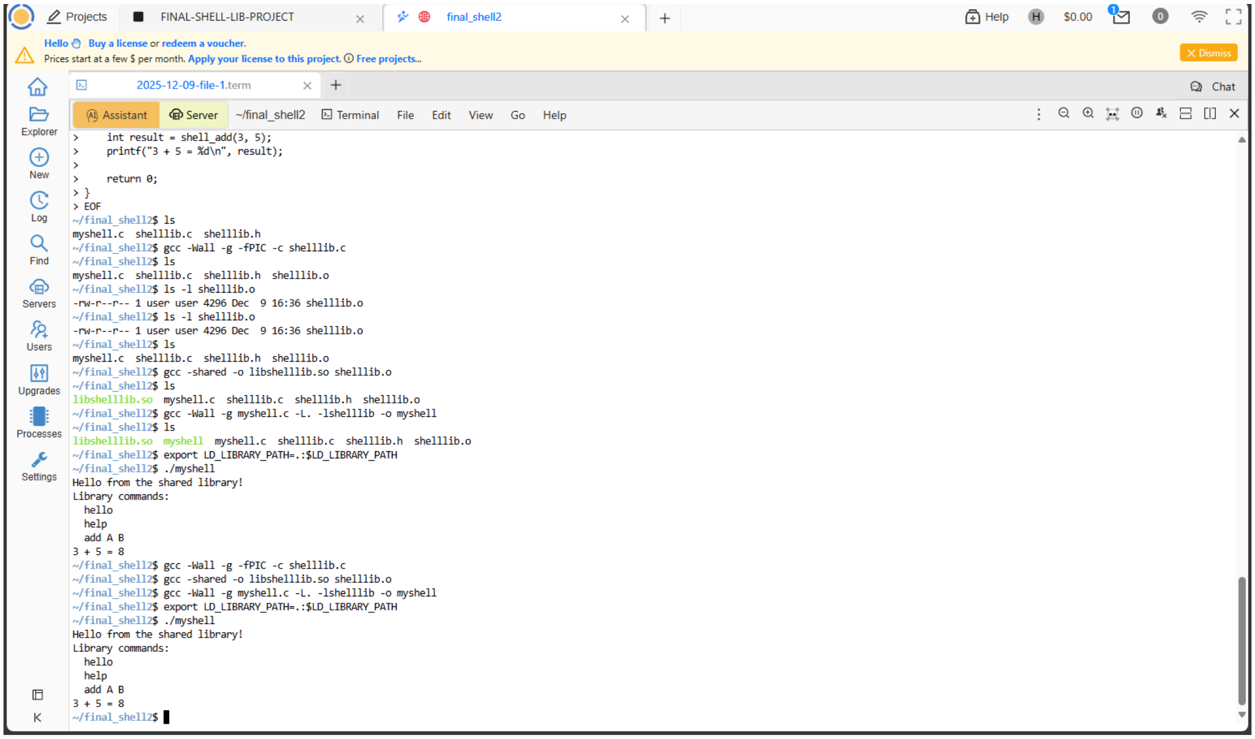
Built shared library: libshelllib.so



```
libshelllib.so  myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ gcc -Wall -g myshell.c -L. -lshelllib -o myshell
~/final_shell2$ ls
libshelllib.so  myshell  myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ export LD_LIBRARY_PATH=.:$LD_LIBRARY_PATH
~/final_shell2$ ./myshell
Hello from the shared library!
Library commands:
  hello
  help
  add A B
3 + 5 = 8
~/final_shell2$
```

ran ./myshell

output of help, hello, add 2 3, and exit



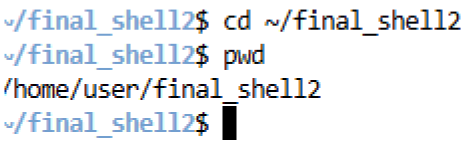
```
>     int result = shell_add(3, 5);
>     printf("3 + 5 = %d\n", result);
>
>     return 0;
> }
> EOF
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h
~/final_shell2$ gcc -Wall -g -fPIC -c shelllib.c
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ ls -l shelllib.o
-rw-r--r-- 1 user user 4296 Dec  9 16:36 shelllib.o
~/final_shell2$ ls -l shelllib.o
-rw-r--r-- 1 user user 4296 Dec  9 16:36 shelllib.o
~/final_shell2$ ls
myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ gcc -shared -o libshelllib.so shelllib.o
~/final_shell2$ ls
libshelllib.so  myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ gcc -Wall -g myshell.c -L. -lshelllib -o myshell
~/final_shell2$ ls
libshelllib.so  myshell  myshell.c  shelllib.c  shelllib.h  shelllib.o
~/final_shell2$ export LD_LIBRARY_PATH=.:$LD_LIBRARY_PATH
~/final_shell2$ ./myshell
Hello from the shared library!
Library commands:
  hello
  help
  add A B
3 + 5 = 8
~/final_shell2$ gcc -Wall -g -fPIC -c shelllib.c
~/final_shell2$ gcc -shared -o libshelllib.so shelllib.o
~/final_shell2$ gcc -Wall -g myshell.c -L. -lshelllib -o myshell
~/final_shell2$ export LD_LIBRARY_PATH=.:$LD_LIBRARY_PATH
~/final_shell2$ ./myshell
Hello from the shared library!
Library commands:
  hello
  help
  add A B
3 + 5 = 8
~/final_shell2$ 
```

This screenshot shows the final working implementation of the custom shell program linked against the libshelllib.so shared library.

Steps demonstrated:

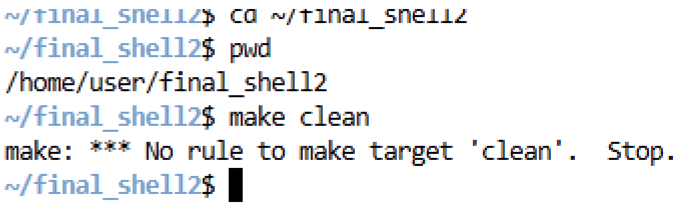
- Compilation of myshell with gcc using -L. and -lshelllib
- Setting LD_LIBRARY_PATH to locate the shared library at runtime
- Running the shell program (./myshell)
- Execution of all supported shell commands:
    - hello → displays output from the shared library
    - help → lists available commands
    - add 3 5 → successfully performs arithmetic calculation

This confirms correct dynamic linking and functional integration of the shared C library within the shell application.



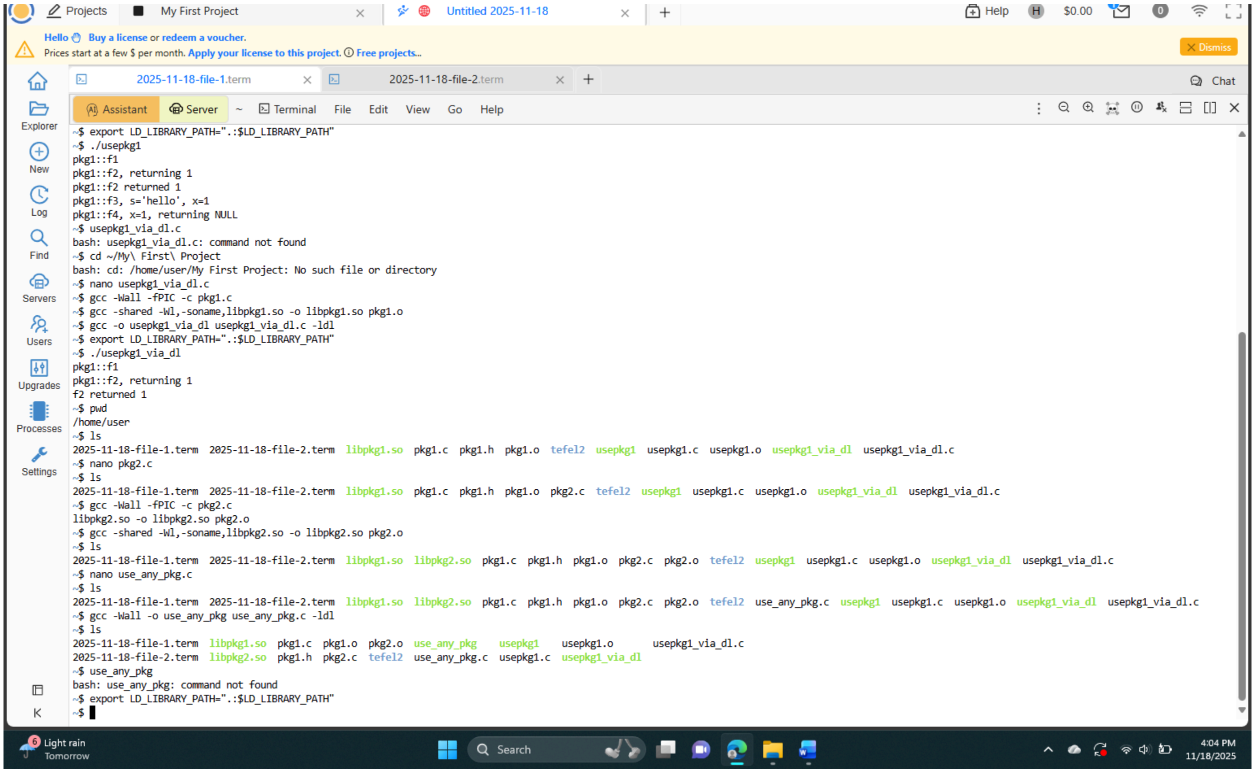
```
~/final_shell2$ cd ~/final_shell2
~/final_shell2$ pwd
/home/user/final_shell2
~/final_shell2$ █
```

Project working directory set to final_shell2



```
~/final_shell2$ cd ~/final_shell2
~/final_shell2$ pwd
/home/user/final_shell2
~/final_shell2$ make clean
make: *** No rule to make target 'clean'.  Stop.
~/final_shell2$ █
```

Cleaning old compiled files



```
~$ export LD_LIBRARY_PATH=".:$LD_LIBRARY_PATH"
~$ ./usepkg1
pkg1::f1
pkg1::f2, returning 1
pkg1::f2 returned 1
pkg1::f3, s='hello', x=1
pkg1::f4, x=1, returning NULL
~$ usepkg1_via_dl.c
bash: usepkg1_via_dl.c: command not found
~$ cd ~/My\ First\ Project
bash: cd: /home/user/My First Project: No such file or directory
~$ nano usepkg1_via_dl.c
~$ gcc -Wall -fPIC -c pkg1.c
~$ gcc -shared -Wl,-soname,libpkg1.so -o libpkg1.so pkg1.o
~$ gcc -o usepkg1_via_dl usepkg1_via_dl.c -ldl
~$ export LD_LIBRARY_PATH=".:$LD_LIBRARY_PATH"
~$ ./usepkg1_via_dl
pkg1::f1
pkg1::f2, returning 1
f2 returned 1
~$ pwd
/home/user
~$ ls
2025-11-18-file-1.term  2025-11-18-file-2.term  libpkg1.so  pkg1.c  pkg1.h  pkg1.o  tefel2  usepkg1  usepkg1.c  usepkg1.o  usepkg1_via_dl  usepkg1_via_dl.c
~$ nano pkg2.c
~$ ls
2025-11-18-file-1.term  2025-11-18-file-2.term  libpkg1.so  pkg1.c  pkg1.h  pkg1.o  pkg2.c  tefel2  usepkg1  usepkg1.c  usepkg1.o  usepkg1_via_dl  usepkg1_via_dl.c
~$ gcc -Wall -fPIC -c pkg2.c
libpkg2.so -o libpkg2.so pkg2.o
~$ gcc -shared -Wl,-soname,libpkg2.so -o libpkg2.so pkg2.o
~$ ls
2025-11-18-file-1.term  2025-11-18-file-2.term  libpkg1.so  libpkg2.so  pkg1.c  pkg1.h  pkg1.o  pkg2.c  pkg2.o  tefel2  usepkg1  usepkg1.c  usepkg1.o  usepkg1_via_dl  usepkg1_via_dl.c
~$ nano use_any_pkg.c
~$ ls
2025-11-18-file-1.term  2025-11-18-file-2.term  libpkg1.so  libpkg2.so  pkg1.c  pkg1.h  pkg1.o  pkg2.c  pkg2.o  tefel2  use_any_pkg.c  usepkg1  usepkg1.c  usepkg1.o  usepkg1_via_dl  usepkg1_via_dl.c
~$ gcc -Wall -o use_any_pkg use_any_pkg.c -ldl
~$ ls
2025-11-18-file-1.term  libpkg1.so  pkg1.c  pkg1.o  pkg2.o  use_any_pkg    usepkg1    usepkg1.o       usepkg1_via_dl.c
2025-11-18-file-2.term  libpkg2.so  pkg1.h  pkg2.c  tefel2  use_any_pkg.c  usepkg1.c  usepkg1_via_dl
~$ use_any_pkg
bash: use_any_pkg: command not found
~$ export LD_LIBRARY_PATH=".:$LD_LIBRARY_PATH"
~$ █
```

Terminal showing shared library creation, program linking, and dynamic execution.