**Library Module – Shared Library\_Assignment01**

1. **Create 3 files as below.**
   1. **libapplication.c – will contain main() and will invoke functions in cal\_utility.c:**
2. Created a libapplication.c file that contains main() and invoking add(), sub() functions in cal\_utility.c**.**

Output: The code in libapplication.c

A computer screen with text

Description automatically generated

* 1. **cal\_utility.c – will contain atleast 2 or more functions [ You may add definitions of the functions in this file ]:**

1. Created a cal\_utility.c file containing definitions of add() and sub() functions.

Output: The code in cal\_utility.c

A screen shot of a computer code

Description automatically generated

* 1. **cal\_utility.h – will contain the extern declarations/prototypes of the functions in cal\_utility.c:**

1. Created a cal\_utility.h file containing prototypes of the functions in cal\_utility.

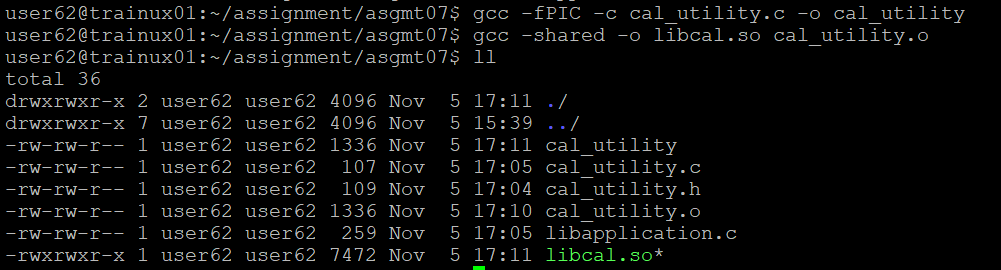
Output: The code in cal\_utility.h

A black screen with white text

Description automatically generated

1. **Refer the steps for shared library and create a shared library comprising of cal\_utility.c,.h files?**
2. To create a shared library, we need to first compile the cal\_utility.c file using -fPIC option to create a .o file. This .o file will form the shared library, to do that use gcc command with option -shared.(Ensure that library name is prefixed with “lib” and extension as “.so”) as $ gcc -shared -o libcal.so cal\_utility.o.

Output:



1. **Create an executable using shared library?**
2. To create an executable file, link the library with the .o file containing the main() using command below. [Exclude the “lib” prefix and extension in library name].

$ gcc libapplication.o -L -lcal -o sample

Output:

A black screen with white text

Description automatically generated