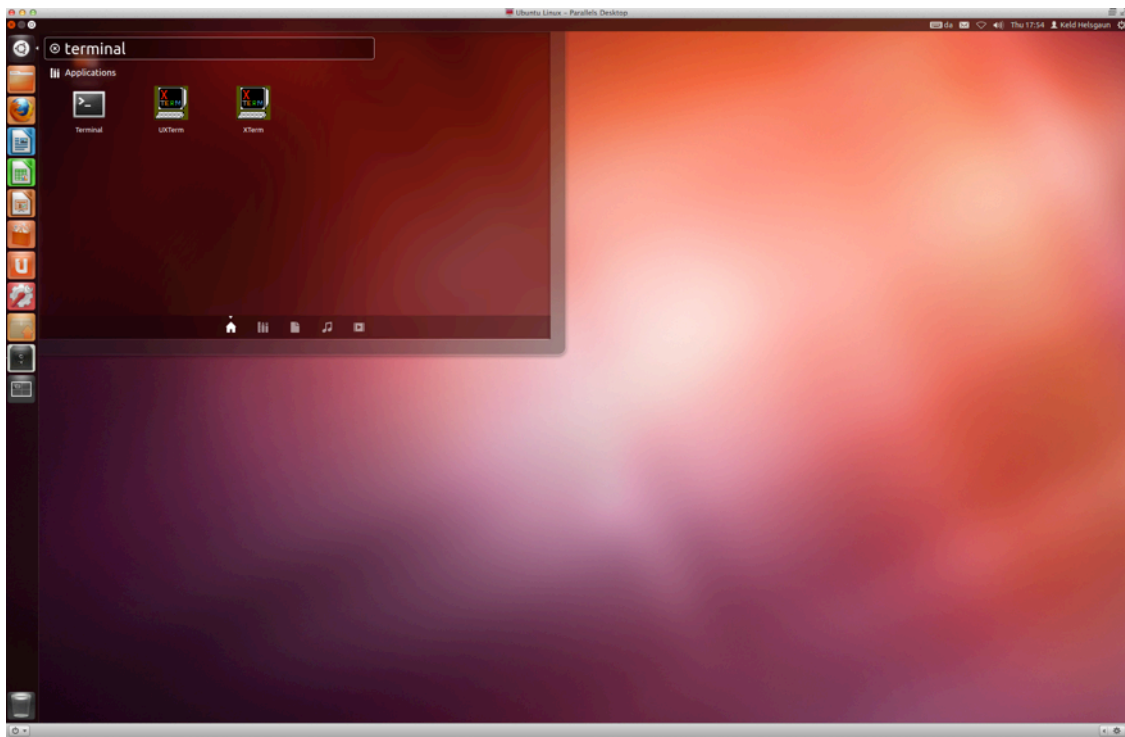


How to Compile and Run a C Program on Ubuntu Linux

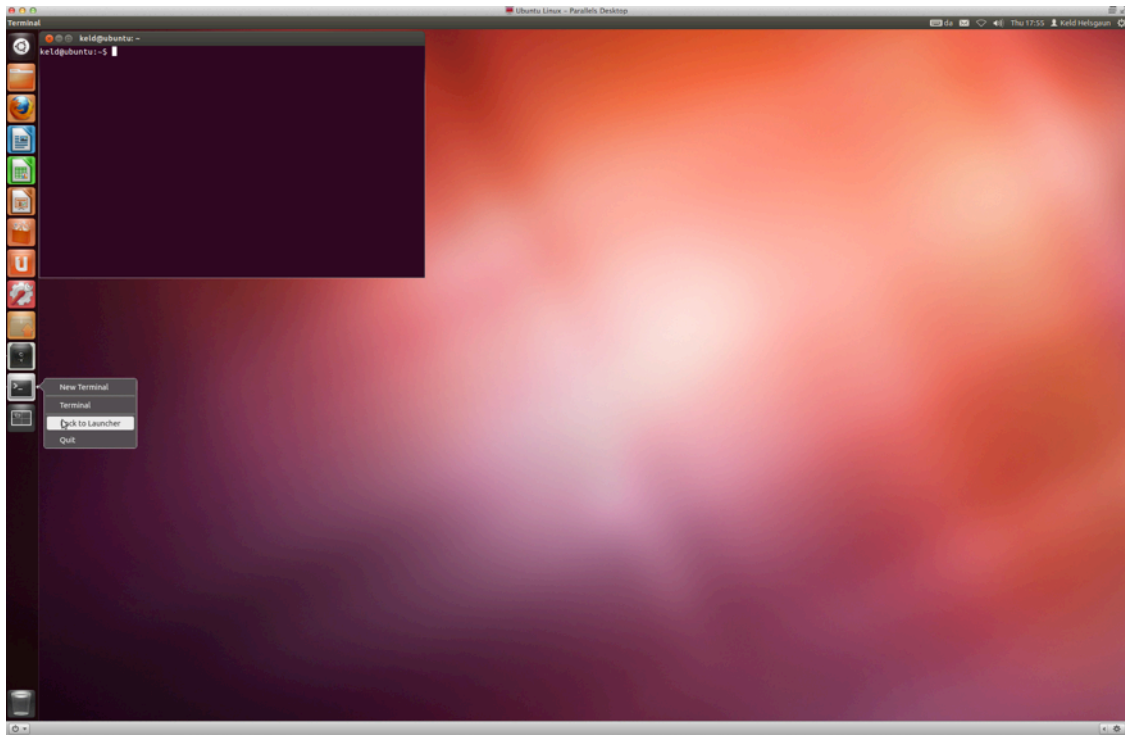
This document shows how to compile and run a C program on Ubuntu Linux using the gcc compiler.

Step 1. Open up a terminal

Search for the terminal application in the Dash tool (located as the topmost item in the Launcher). Open up a terminal by clicking on the icon.



For ease of future access to the terminal application, right click its icon in the Launcher and select “Lock to Launcher”.



Step 2. Use a text editor to create the C source code.

Type the command

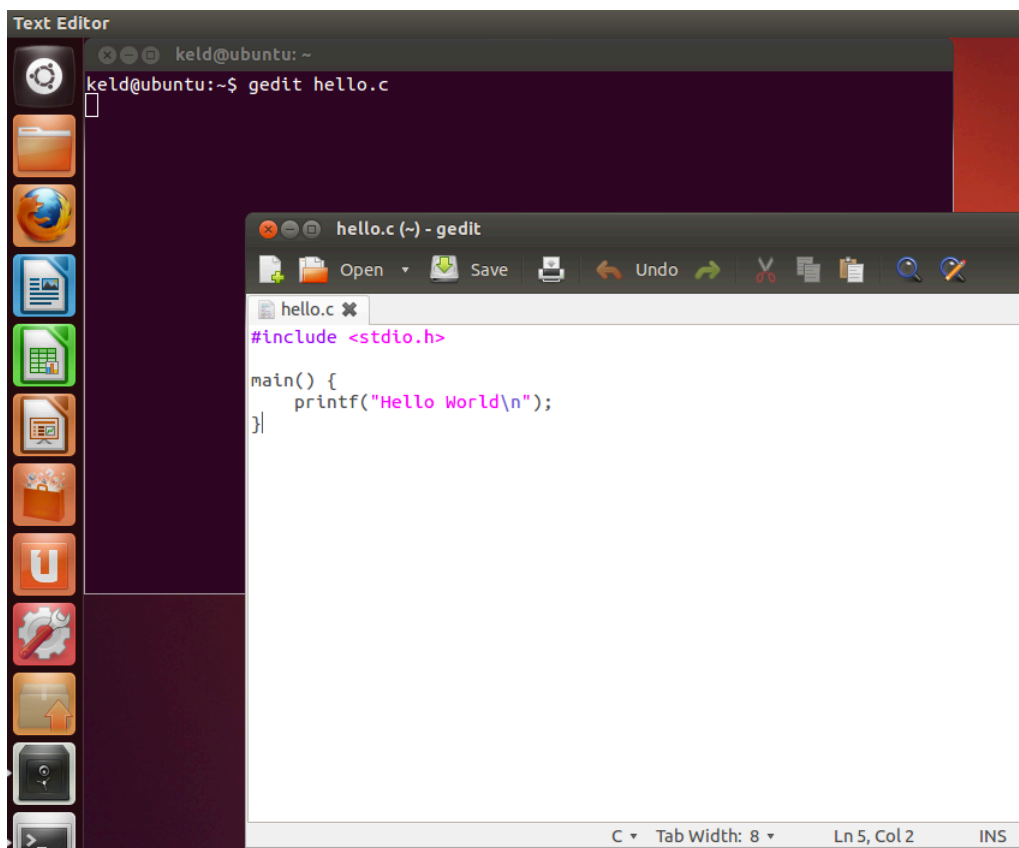
```
gedit hello.c
```

and enter the C source code below:

```
#include <stdio.h>

main() {
    printf("Hello World\n");
}
```

Close the editor window.

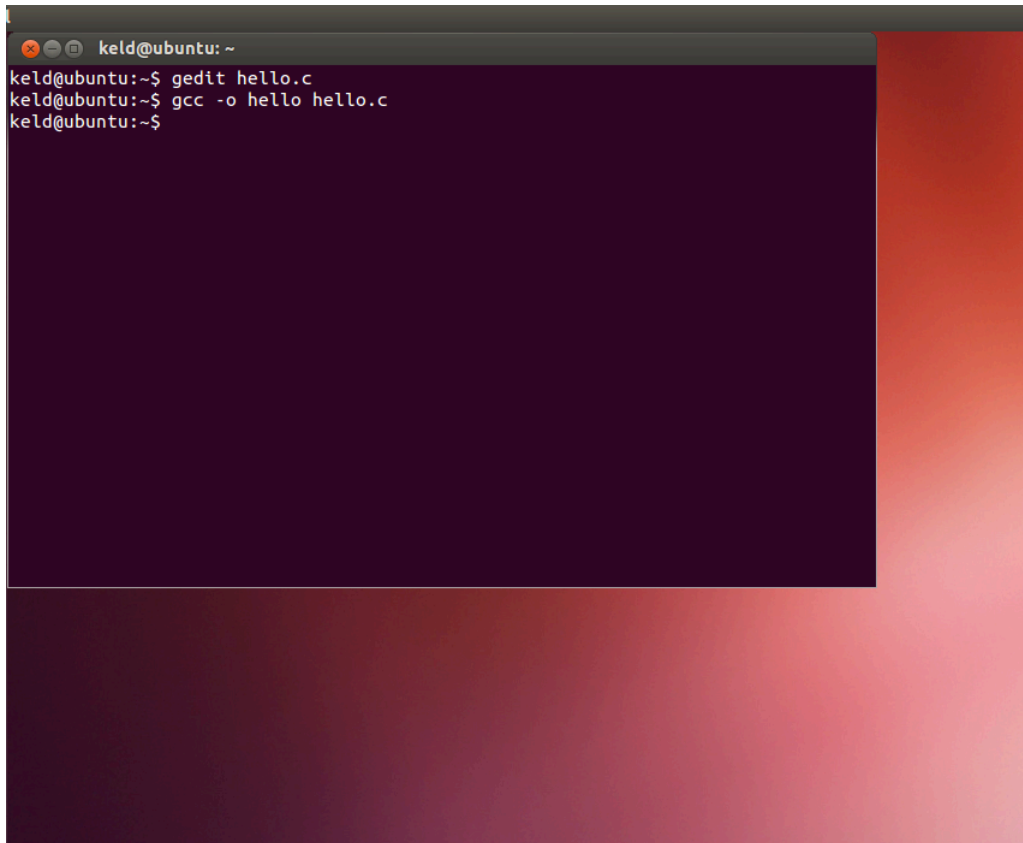


Step 3. Compile the program.

Type the command

```
gcc -o hello hello.c
```

This command will invoke the GNU C compiler to compile the file `hello.c` and output (-o) the result to an executable called `hello`.

A terminal window titled 'keld@ubuntu: ~' with a dark purple background. It shows the execution of two commands: 'gedit hello.c' and 'gcc -o hello hello.c'. The window has standard Ubuntu window controls (close, maximize, and a button with a question mark) in the top-left corner.

```
keld@ubuntu: ~  
keld@ubuntu:~$ gedit hello.c  
keld@ubuntu:~$ gcc -o hello hello.c  
keld@ubuntu:~$
```

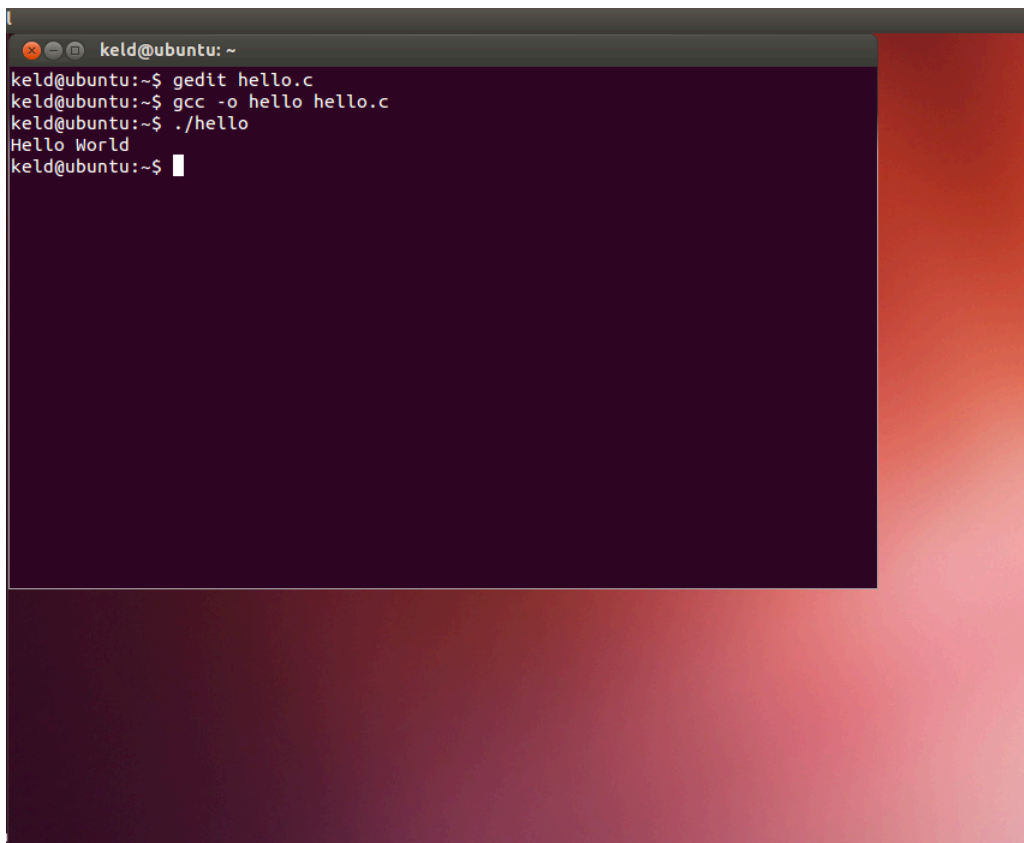
Step 4. Execute the program.

Type the command

```
./hello
```

This should result in the output

```
Hello World
```

A terminal window titled 'keld@ubuntu: ~' with a dark purple background. It shows the following commands and output:

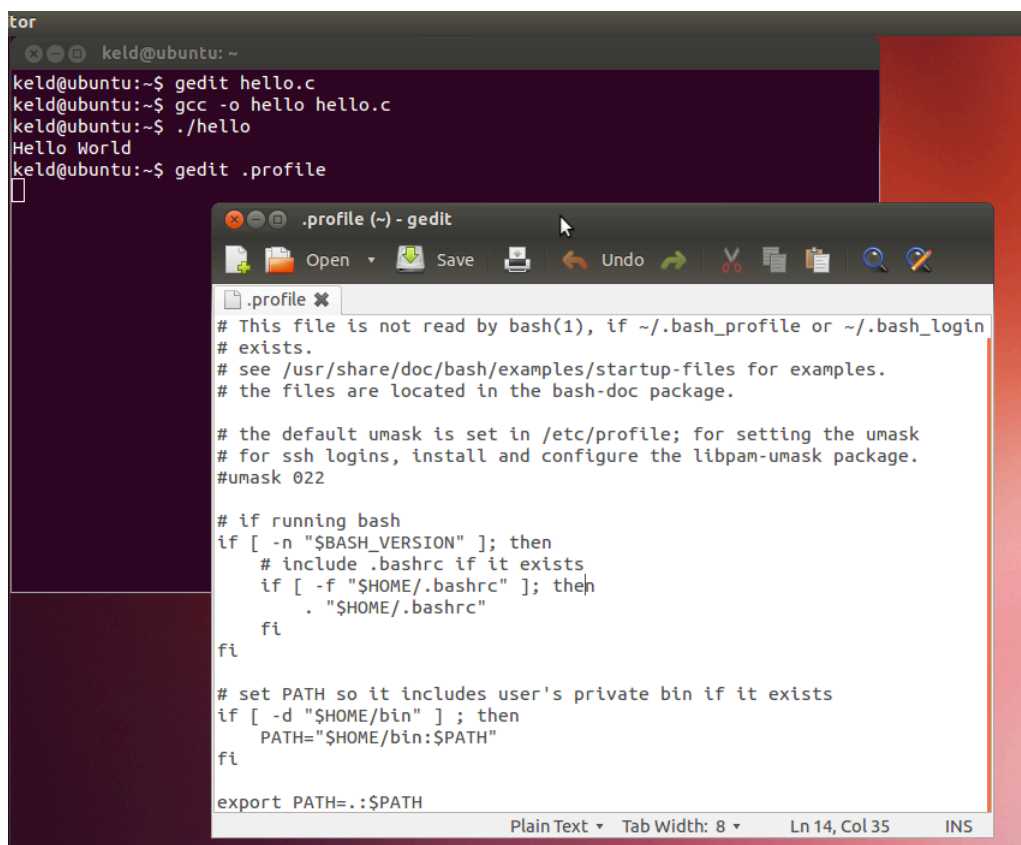
```
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$
```

Optional step

In order to avoid the `./` prefix each time a program is to be executed, insert the following as the last line in the file `.profile` (located in your home folder):

```
export PATH=.:$PATH
```

This step needs only to be done once.



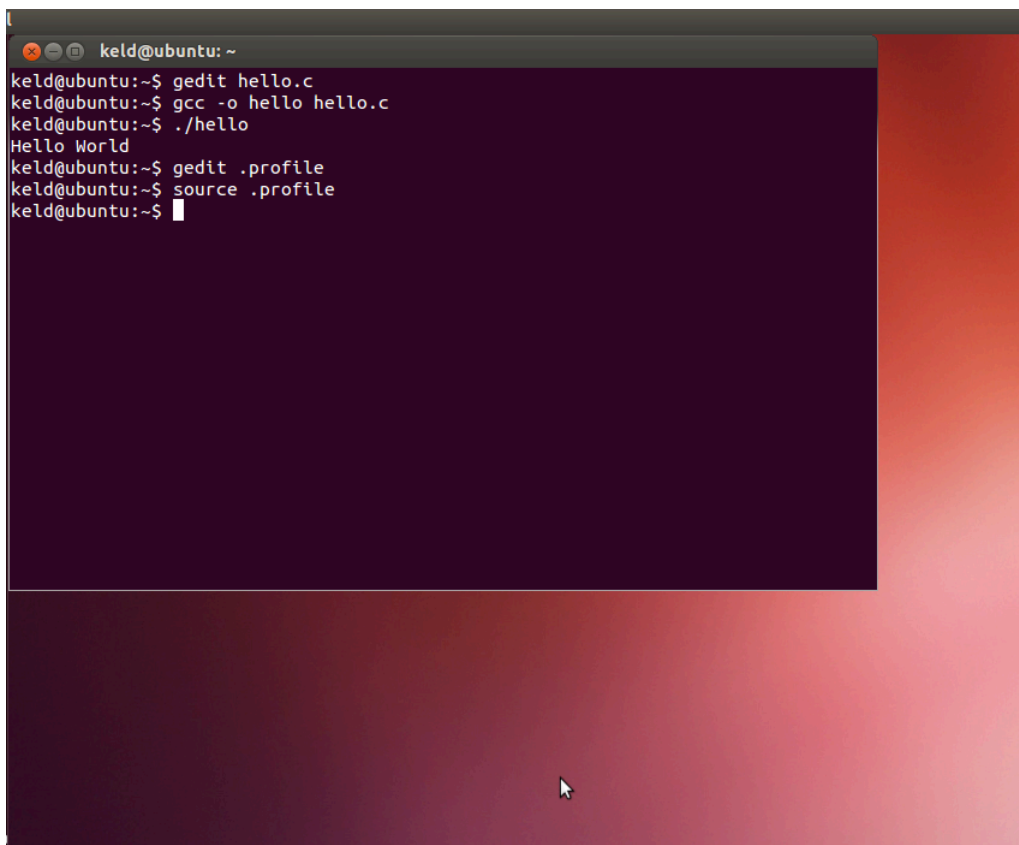
The screenshot shows a terminal window with the following commands and output:

```
keld@ubuntu: ~  
keld@ubuntu:~$ gedit hello.c  
keld@ubuntu:~$ gcc -o hello hello.c  
keld@ubuntu:~$ ./hello  
Hello World  
keld@ubuntu:~$ gedit .profile
```

The `.profile` file is open in the gedit editor. The content of the file is:

```
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login  
# exists.  
# see /usr/share/doc/bash/examples/startup-files for examples.  
# the files are located in the bash-doc package.  
  
# the default umask is set in /etc/profile; for setting the umask  
# for ssh logins, install and configure the libpam-umask package.  
#umask 022  
  
# if running bash  
if [ -n "$BASH_VERSION" ]; then  
    # include .bashrc if it exists  
    if [ -f "$HOME/.bashrc" ]; then  
        . "$HOME/.bashrc"  
    fi  
fi  
  
# set PATH so it includes user's private bin if it exists  
if [ -d "$HOME/bin" ] ; then  
    PATH="$HOME/bin:$PATH"  
fi  
  
export PATH=.:$PATH
```

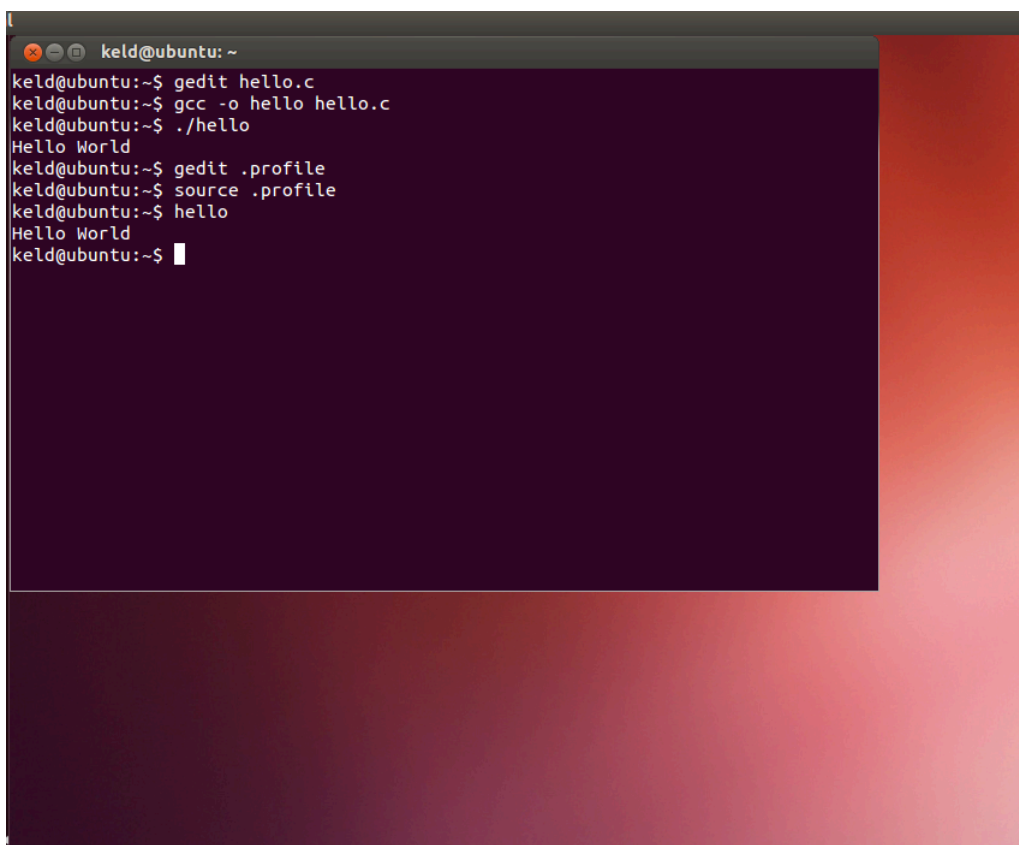
The status bar at the bottom of the gedit window indicates "Plain Text", "Tab Width: 8", "Ln 14, Col 35", and "INS".



A terminal window titled 'keld@ubuntu: ~' with a dark purple background. The window shows the following commands and output:

```
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$ gedit .profile
keld@ubuntu:~$ source .profile
keld@ubuntu:~$
```

The cursor is at the end of the last line.



A terminal window titled 'keld@ubuntu: ~' with a dark purple background. The window shows the following commands and output:

```
keld@ubuntu:~$ gedit hello.c
keld@ubuntu:~$ gcc -o hello hello.c
keld@ubuntu:~$ ./hello
Hello World
keld@ubuntu:~$ gedit .profile
keld@ubuntu:~$ source .profile
keld@ubuntu:~$ hello
Hello World
keld@ubuntu:~$
```

The cursor is at the end of the last line.