

Name: Bhavesh Kewalramani

Roll No.-25

Section: A

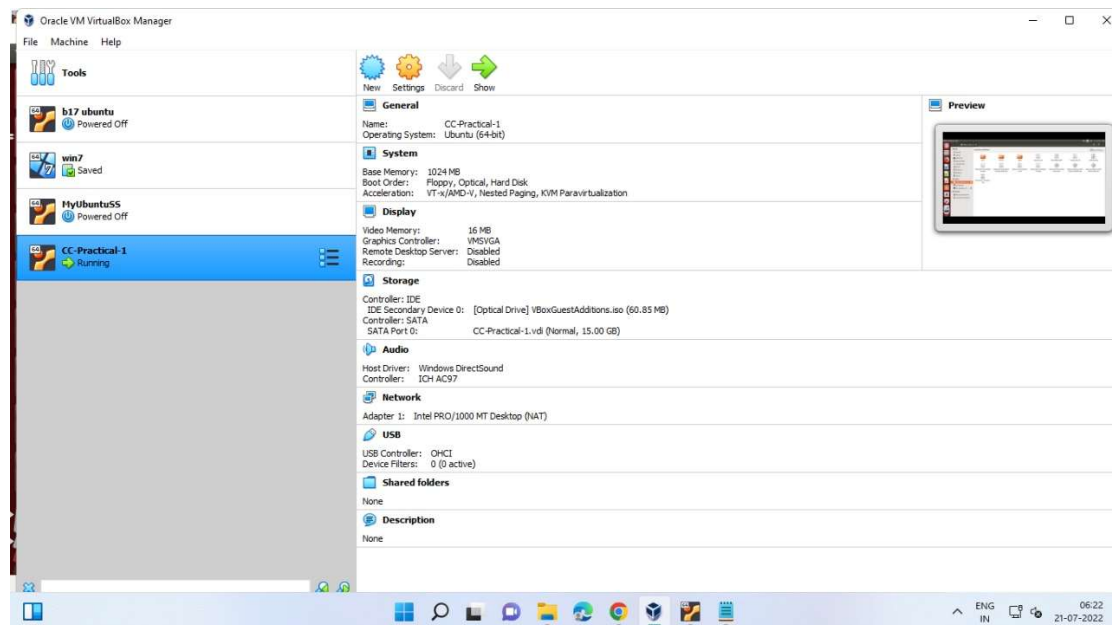
Semester: VII

Batch: A1

Aim: Implementing applications using Google App Engine (PaaS).

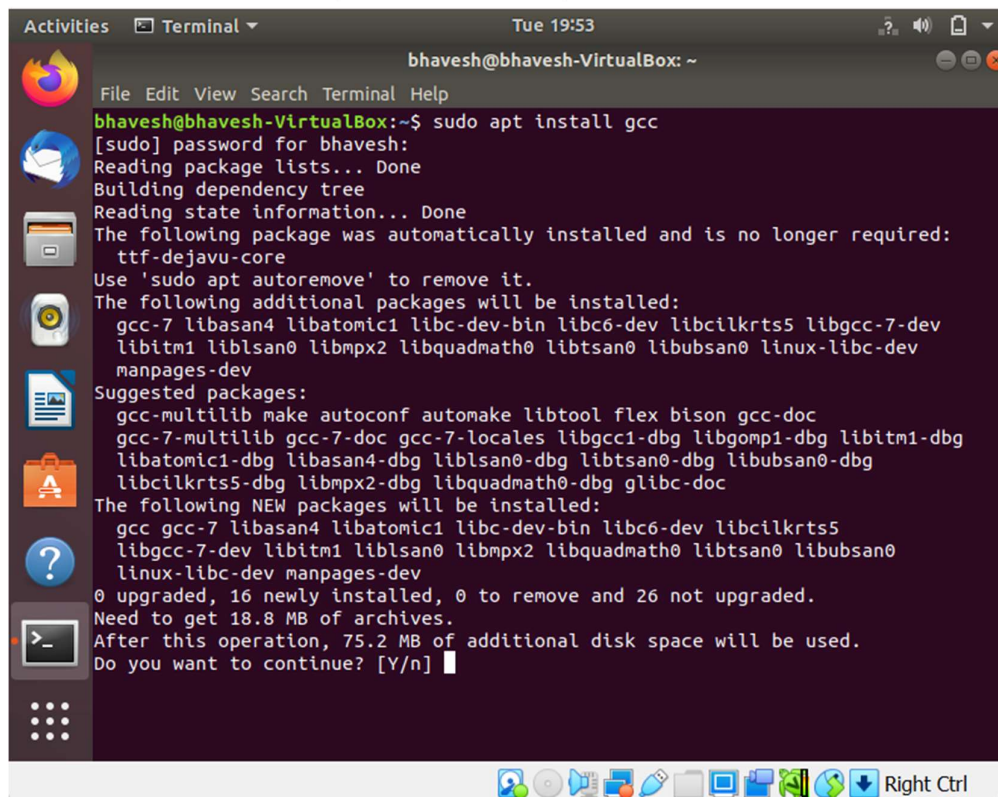
Output:

1. **Install Virtualbox/VMware Workstation with different flavours of linux or windows OS on top of windows.**



Installation of Oracle VM Virtual Box and creation of Virtual Machine named “CC Practical-1” which contains Linux OS and win 7 which contains windows 7 OS.

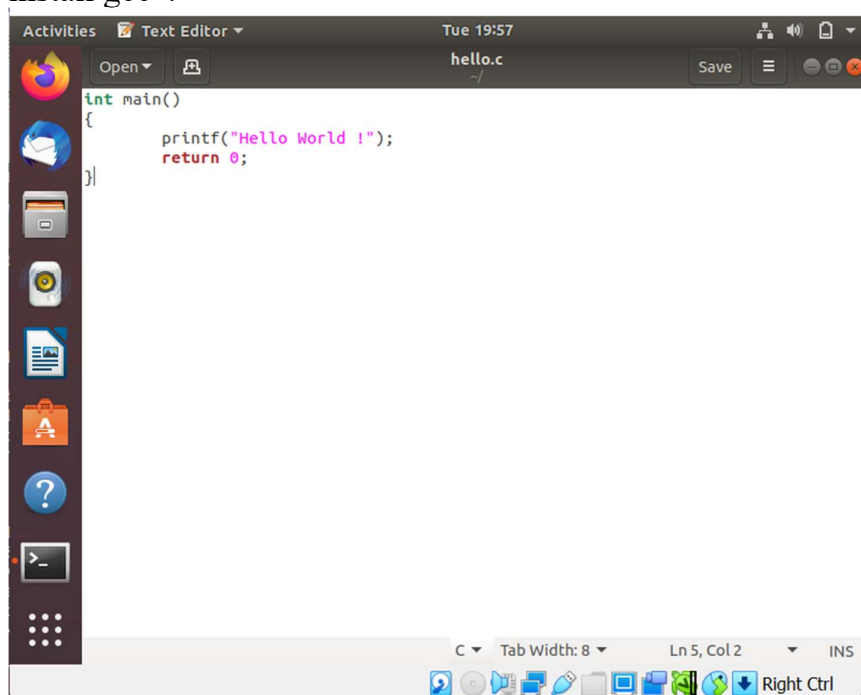
2. **Install a C compiler in the virtual machine created using virtual box and execute Simple Programs**



A terminal window titled "Terminal" with the user "bhavesh@bhavesh-VirtualBox: ~". The command "sudo apt install gcc" has been executed. The output shows the password prompt, package list reading, dependency tree building, and state information reading. It lists packages to be removed (ttf-dejavu-core), additional packages to be installed (gcc-7, libasan4, libatomic1, libc-dev-bin, libc6-dev, libcilkrts5, libgcc-7-dev, libitm1, liblsan0, libmpx2, libquadmath0, libtsan0, libubsan0, linux-libc-dev, manpages-dev), and suggested packages (gcc-multilib, make, autoconf, automake, libtool, flex, bison, gcc-doc, etc.). It also shows the disk space requirements and asks for confirmation to continue.

```
bhavesh@bhavesh-VirtualBox:~$ sudo apt install gcc
[sudo] password for bhavesh:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  ttf-dejavu-core
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  gcc-7 libasan4 libatomic1 libc-dev-bin libc6-dev libcilkrts5 libgcc-7-dev
  libitm1 liblsan0 libmpx2 libquadmath0 libtsan0 libubsan0 linux-libc-dev
  manpages-dev
Suggested packages:
  gcc-multilib make autoconf automake libtool flex bison gcc-doc
  gcc-7-multilib gcc-7-doc gcc-7-locales libgcc1-dbg libgomp1-dbg libitm1-dbg
  libatomic1-dbg libasan4-dbg liblsan0-dbg libtsan0-dbg libubsan0-dbg
  libcilkrts5-dbg libmpx2-dbg libquadmath0-dbg glibc-doc
The following NEW packages will be installed:
  gcc gcc-7 libasan4 libatomic1 libc-dev-bin libc6-dev libcilkrts5
  libgcc-7-dev libitm1 liblsan0 libmpx2 libquadmath0 libtsan0 libubsan0
  linux-libc-dev manpages-dev
0 upgraded, 16 newly installed, 0 to remove and 26 not upgraded.
Need to get 18.8 MB of archives.
After this operation, 75.2 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

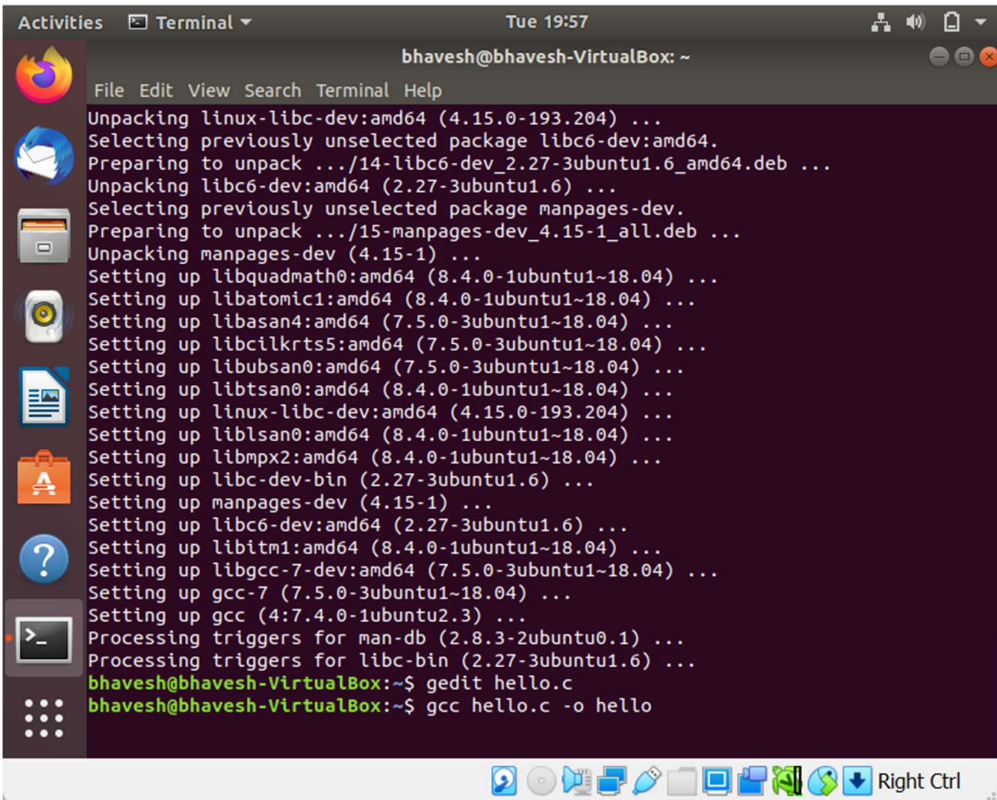
Installing gcc compiler to execute the C program using the command “sudo apt install gcc”.



A text editor window titled "Text Editor" with the file "hello.c" open. The code is a simple C program that prints "Hello World !" and returns 0.

```
int main()
{
    printf("Hello World !");
    return 0;
}
```

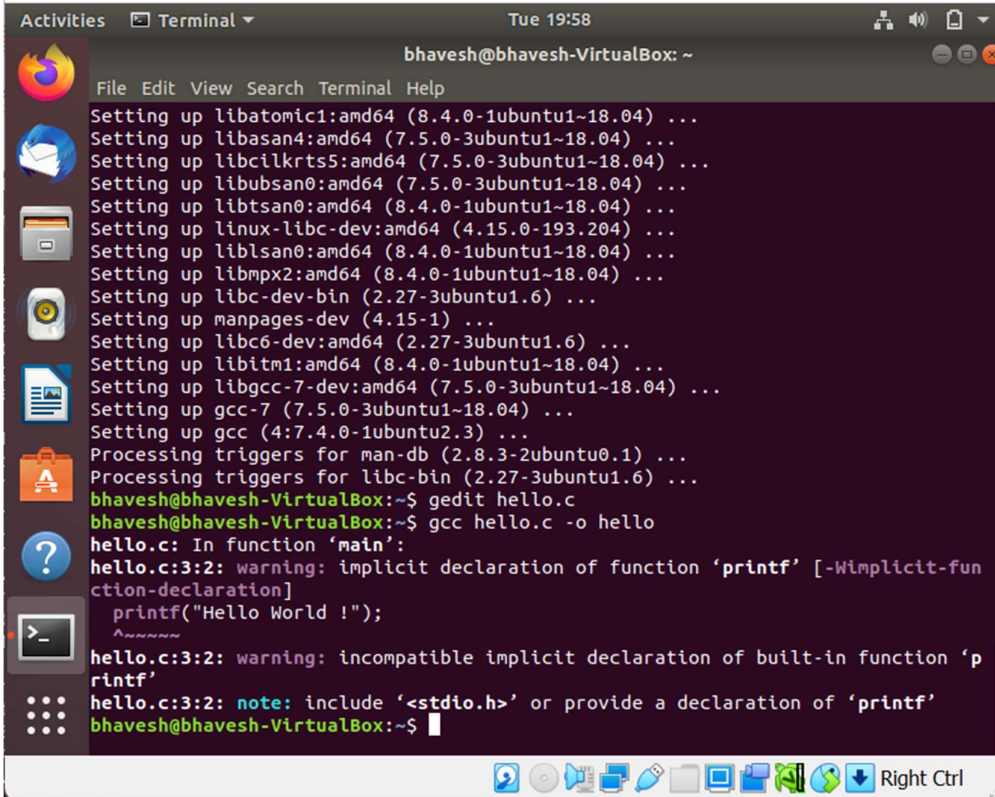
Sample Hello World program written in C language and the file is saved with .c extension.



The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox: ~" with a menu bar (File, Edit, View, Search, Terminal, Help) and a sidebar with application icons. The terminal output shows the installation of several packages: linux-libc-dev, libc6-dev, manpages-dev, libquadmath0, libatomic1, libasan4, libcilkrts5, libubsan0, libtsan0, liblsan0, libmpx2, libc-dev-bin, manpages-dev, libc6-dev, libitm1, libgcc-7-dev, gcc-7, and gcc. After the installation, the user runs `gedit hello.c` and then `gcc hello.c -o hello`.

```
bhavesh@bhavesh-VirtualBox: ~  
File Edit View Search Terminal Help  
Unpacking linux-libc-dev:amd64 (4.15.0-193.204) ...  
Selecting previously unselected package libc6-dev:amd64.  
Preparing to unpack .../14-libc6-dev_2.27-3ubuntu1.6_amd64.deb ...  
Unpacking libc6-dev:amd64 (2.27-3ubuntu1.6) ...  
Selecting previously unselected package manpages-dev.  
Preparing to unpack .../15-manpages-dev_4.15-1_all.deb ...  
Unpacking manpages-dev (4.15-1) ...  
Setting up libquadmath0:amd64 (8.4.0-1ubuntu1~18.04) ...  
Setting up libatomic1:amd64 (8.4.0-1ubuntu1~18.04) ...  
Setting up libasan4:amd64 (7.5.0-3ubuntu1~18.04) ...  
Setting up libcilkrts5:amd64 (7.5.0-3ubuntu1~18.04) ...  
Setting up libubsan0:amd64 (7.5.0-3ubuntu1~18.04) ...  
Setting up libtsan0:amd64 (8.4.0-1ubuntu1~18.04) ...  
Setting up linux-libc-dev:amd64 (4.15.0-193.204) ...  
Setting up liblsan0:amd64 (8.4.0-1ubuntu1~18.04) ...  
Setting up libmpx2:amd64 (8.4.0-1ubuntu1~18.04) ...  
Setting up libc-dev-bin (2.27-3ubuntu1.6) ...  
Setting up manpages-dev (4.15-1) ...  
Setting up libc6-dev:amd64 (2.27-3ubuntu1.6) ...  
Setting up libitm1:amd64 (8.4.0-1ubuntu1~18.04) ...  
Setting up libgcc-7-dev:amd64 (7.5.0-3ubuntu1~18.04) ...  
Setting up gcc-7 (7.5.0-3ubuntu1~18.04) ...  
Setting up gcc (4:7.4.0-1ubuntu2.3) ...  
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...  
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...  
bhavesh@bhavesh-VirtualBox:~$ gedit hello.c  
bhavesh@bhavesh-VirtualBox:~$ gcc hello.c -o hello
```

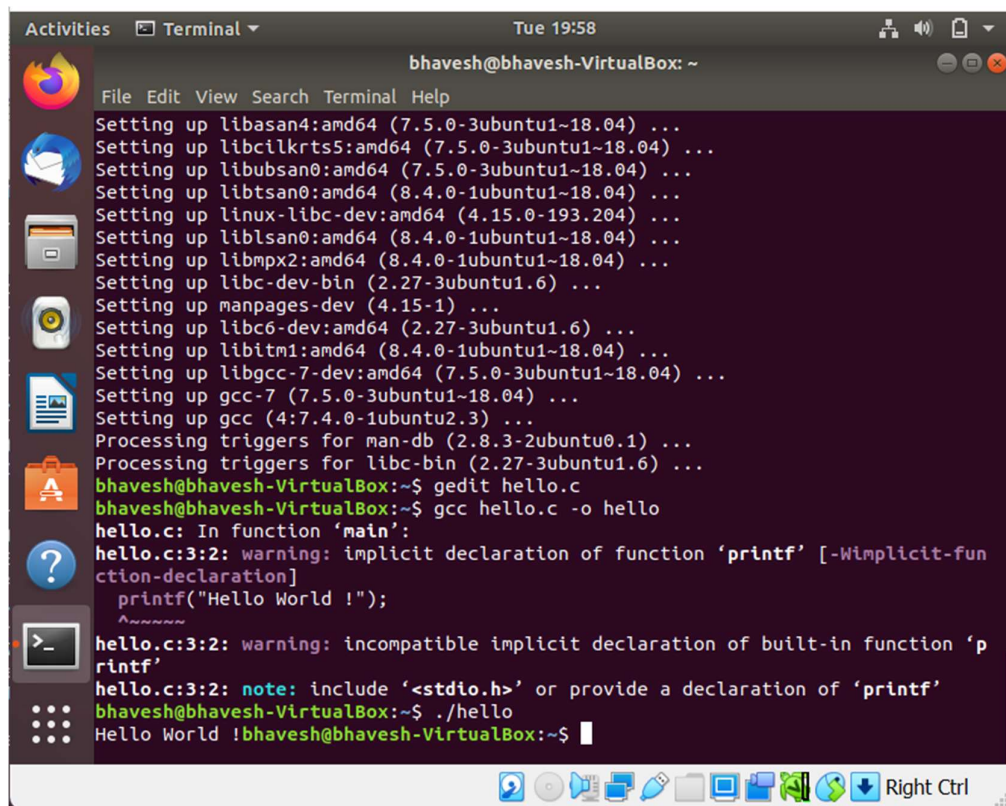
Using gcc compiler we execute the hello.c program. -o saves the output file with filename hello



The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox: ~" with a menu bar (File, Edit, View, Search, Terminal, Help) and a sidebar with application icons. The terminal output shows the installation of various packages, followed by the creation and compilation of a C program named "hello.c".

```
Setting up libatomic1:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libasan4:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up libcilkrts5:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up libubsan0:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up libtsan0:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up linux-libc-dev:amd64 (4.15.0-193.204) ...
Setting up liblsan0:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libmpx2:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libc-dev-bin (2.27-3ubuntu1.6) ...
Setting up manpages-dev (4.15-1) ...
Setting up libc6-dev:amd64 (2.27-3ubuntu1.6) ...
Setting up libitm1:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libgcc-7-dev:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up gcc-7 (7.5.0-3ubuntu1~18.04) ...
Setting up gcc (4:7.4.0-1ubuntu2.3) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...
bhavesh@bhavesh-VirtualBox:~$ gedit hello.c
bhavesh@bhavesh-VirtualBox:~$ gcc hello.c -o hello
hello.c: In function 'main':
hello.c:3:2: warning: implicit declaration of function 'printf' [-Wimplicit-function-declaration]
    printf("Hello World !");
    ^~~~~
hello.c:3:2: warning: incompatible implicit declaration of built-in function 'printf'
hello.c:3:2: note: include '<stdio.h>' or provide a declaration of 'printf'
bhavesh@bhavesh-VirtualBox:~$
```

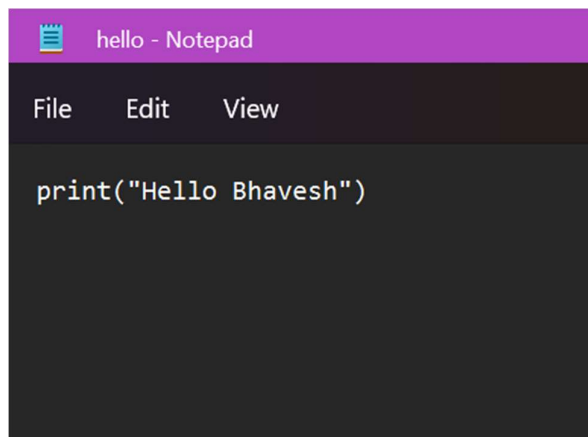
Executing the gcc command to compile the C program.



```
Activities  Terminal  Tue 19:58
bhavesh@bhavesh-VirtualBox: ~
File Edit View Search Terminal Help
Setting up libasan4:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up libcilkrts5:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up libubsan0:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up libtsan0:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up linux-libc-dev:amd64 (4.15.0-193.204) ...
Setting up liblsan0:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libmpx2:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libc-dev-bin (2.27-3ubuntu1.6) ...
Setting up manpages-dev (4.15-1) ...
Setting up libc6-dev:amd64 (2.27-3ubuntu1.6) ...
Setting up libitm1:amd64 (8.4.0-1ubuntu1~18.04) ...
Setting up libgcc-7-dev:amd64 (7.5.0-3ubuntu1~18.04) ...
Setting up gcc-7 (7.5.0-3ubuntu1~18.04) ...
Setting up gcc (4:7.4.0-1ubuntu2.3) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...
bhavesh@bhavesh-VirtualBox:~$ gedit hello.c
bhavesh@bhavesh-VirtualBox:~$ gcc hello.c -o hello
hello.c: In function 'main':
hello.c:3:2: warning: implicit declaration of function 'printf' [-Wimplicit-function-declaration]
    printf("Hello World !");
    ^~~~~
hello.c:3:2: warning: incompatible implicit declaration of built-in function 'printf'
hello.c:3:2: note: include '<stdio.h>' or provide a declaration of 'printf'
bhavesh@bhavesh-VirtualBox:~$ ./hello
Hello World !bhavesh@bhavesh-VirtualBox:~$
```

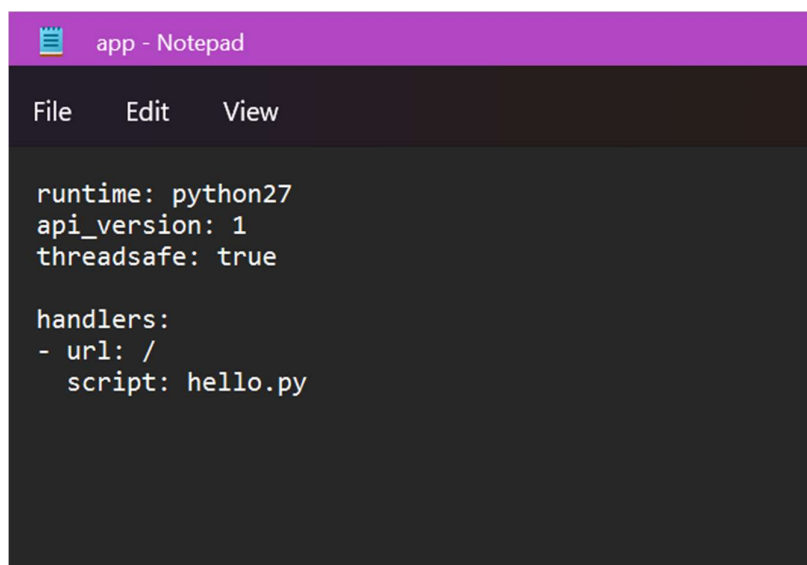
The output is saved in hello file. When we execute the `./hello` command, the program gets executed and gives the output.

3. Install Google App Engine. Create hello world app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Use GAE launcher to launch the web applications using database services.



```
print("Hello Bhavesh")
```

Sample program written in python.



```
runtime: python27
api_version: 1
threadsafe: true

handlers:
- url: /
  script: hello.py
```

The app.yaml file to be used to make web application using GAE.

Name	Status	Date modified	Type	Size
app	✖	10-10-2022 08:15	YAML File	1 KB
hello	✖	08-10-2022 19:39	PY File	1 KB
Practical-3	✖	18-09-2022 17:31	Microsoft Word Doc...	4,938 KB

Files present in the folder which will be used to deploy the application.


```
C:\ProgramData\MicrosoftW x + -
Welcome to the Google Cloud CLI! Run "gcloud -h" to get the list of available commands.

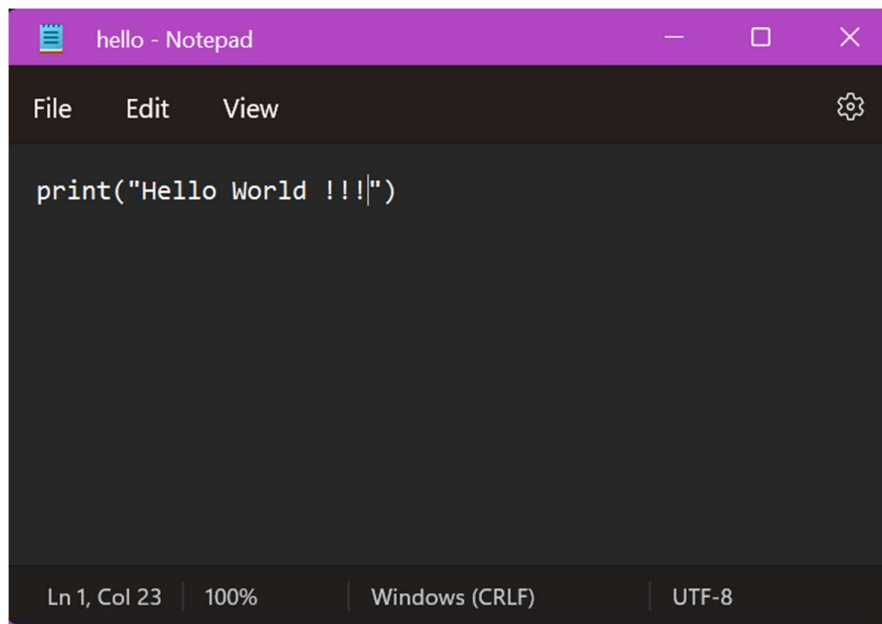
C:\Program Files (x86)\Google\Cloud SDK>python google-cloud-sdk/bin/dev_appserver.py "C:\Users\bhave\OneDrive\Desktop\CC Lab\Practica
l - 2/app.yaml"
INFO 2022-10-10 18:41:04,463 devappserver2.py:240] Using Cloud Datastore Emulator.
We are gradually rolling out the emulator as the default datastore implementation of dev_appserver.
If broken, you can temporarily disable it by --support_datastore_emulator=False
Read the documentation: https://cloud.google.com/appengine/docs/standard/python/tools/migrate-cloud-datastore-emulator
Help us validate that the feature is ready by taking this survey: https://goo.gl/forms/UArIcs8K9CUCm733
Report issues at: https://issuetracker.google.com/issues/new?component=187272

INFO 2022-10-10 18:41:04,473 devappserver2.py:317] Skipping SDK update check.
INFO 2022-10-10 18:41:05,232 datastore_emulator.py:156] Starting Cloud Datastore emulator at: http://localhost:54858
INFO 2022-10-10 18:41:07,832 datastore_emulator.py:162] Cloud Datastore emulator responded after 2.600000 seconds
INFO 2022-10-10 18:41:07,834 <string>:384] Starting API server at: http://localhost:54914
INFO 2022-10-10 18:41:07,900 <string>:374] Starting gRPC API server at: http://localhost:54915
INFO 2022-10-10 18:41:08,168 dispatcher.py:280] Starting module "default" running at: http://localhost:8080
INFO 2022-10-10 18:41:08,171 admin_server.py:70] Starting admin server at: http://localhost:8080
```

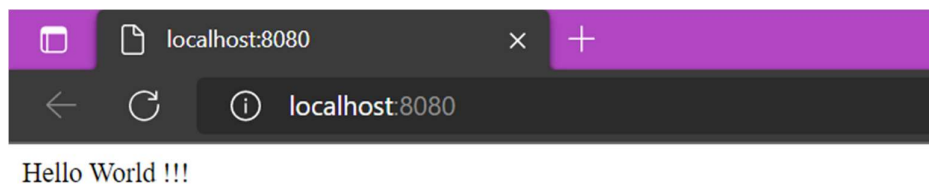
Launching the Google Cloud SDK shell. The application is deployed using the command “python google-cloud-sdk/bin/dev_appserver.py “[Path to the Directory]” “



The application is launched after the command has been executed



Now editing the file to make changing to Hello World



The application has been launched again using the above command

The same steps are used to launch other applications using above steps.

Conclusion:

In this practical, we install different types of OS on Oracle VirtualBox. We executed a simple C Program in Linux. We also saw how PaaS is useful and how Google App Engine is used to deploy applications. Google App Engine is one of PaaS service used today by many people.