

**Govt. Graduate College for Women Gujranwala**

Bs.4-year Program: 8<sup>th</sup> Semester---2025

Roll No.....

Paper: Virtual System and services

Course Code:DI-421L

Marks: 50

---

Q.1. Use a GCC compiler to compile a C program on Ubuntu.

- Create program file on ubuntu desktop.
- Write all commands that are used to create ,compile and run C program on GCC compiler.
- Write why we use GCC compiler instead of turbo.

Q.2 Implement the following.

- Install Google App Engine.
- Implement java program.
- Clone from github.
- Show result of java program on cloud shell.

Q.3. Explain all steps of Docker deployment.

## Solution

### Q.1. Use a GCC compiler to compile a C program on Ubuntu

#### Step 1: Create a C Program File on Ubuntu Desktop

1. **Open Terminal**
2. **Navigate to Desktop:**
3. `cd Desktop`
4. **Create a file named hello.c:**
5. `nano hello.c`
6. **Write the C Program:**
7. `#include <stdio.h>`
- 8.
9. `int main() {`
10. `printf("Hello, world!\n");`
11. `return 0;`
12. `}`
13. **Save the file** by pressing Ctrl + O, then press Enter, then Ctrl + X to exit.

#### Step 2: Compile the Program Using GCC

- Compile the C program using this command:
- `gcc hello.c -o hello`

#### Step 3: Run the Compiled Program

- Execute the compiled file:
- `./hello`

#### Why Use GCC Instead of Turbo C?

- **GCC (GNU Compiler Collection):**
  - Works on modern systems (Linux, macOS, Windows).
  - Open-source and regularly updated.
  - Follows modern C standards (ANSI/ISO).
  - Supports debugging, optimization, and linking tools.
- **Turbo C:**
  - Very old (from DOS era).
  - Not maintained or updated.
  - Poor compatibility with modern systems.
  - Uses outdated C standards.

### Q.2. Implement the following

#### Step 1: Install Google Cloud SDK and App Engine

- **Open Cloud Shell** from <https://console.cloud.google.com>
- Initialize SDK (if not already):
- `gcloud init`
- Install App Engine component for Java:
- `gcloud components install app-engine-java`
- Create App Engine project:
- `gcloud app create`

### Step 2: Implement a Java Program

1. Create a Java file:
2. nano HelloApp.java
3. Write the Java code:
4. public class HelloApp {
5.     public static void main(String[] args) {
6.         System.out.println("Hello from Google Cloud!");
7.     }
8. }
9. Save and compile:
10. javac HelloApp.java
11. java HelloApp

### Step 3: Clone from GitHub

- Example GitHub repository:
- git clone <https://github.com/username/repository-name.git>
- cd repository-name

### Step 4: Show Java Program Result on Cloud Shell

- Compile and run inside the cloned folder:
- javac MyProgram.java
- java MyProgram
- Output will display in the terminal window of Cloud Shell.

## Q.3. Docker Deployment Steps (Run Java App on Windows)

### Step 1: Install Docker Desktop on Windows

- Download and install Docker from: <https://www.docker.com/products/docker-desktop>
- Enable WSL2 backend during installation.

### Step 2: Create Java App

1. **Create a file named HelloDocker.java:**
2. public class HelloDocker {
3.     public static void main(String[] args) {
4.         System.out.println("Java App Running in Docker");
5.     }
6. }
7. **Compile it using terminal or command prompt:**
8. javac HelloDocker.java

### Step 3: Create Dockerfile

- Create a text file named Dockerfile (no extension):
- FROM openjdk:11
- COPY HelloDocker.class .
- CMD ["java", "HelloDocker"]

### Step 4: Build Docker Image

- Open command prompt in the folder where the Dockerfile is located:
- docker build -t javaapp .

**Step 5: Run Docker Container**

- Run the image:
- `docker run javaapp`
- You will see:
- Java App Running in Docker