FUNDAMENTALS OF DEVEOPS

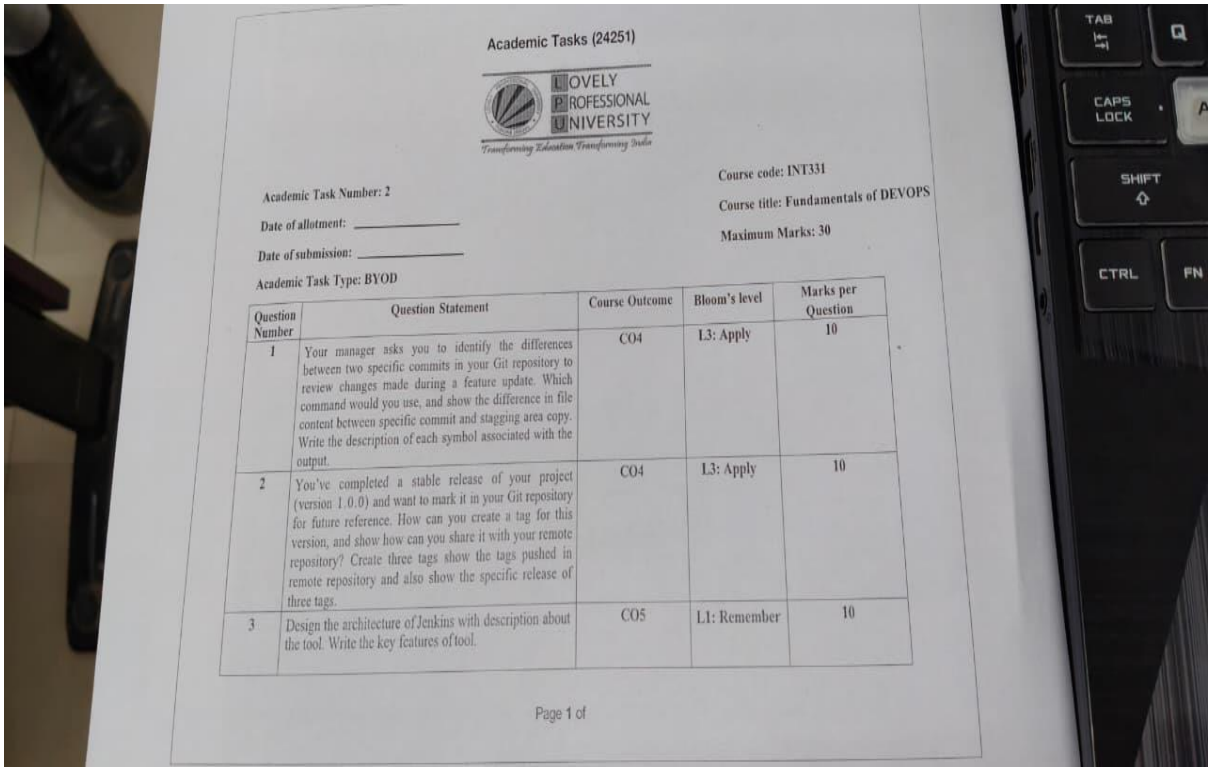
PROJECT-01

Name- Sundram Singh

Reg no. – 12312641

Section:- KM011

Roll no. 44



Academic Tasks (24251)

LOVELY PROFESSIONAL UNIVERSITY

Transforming Education Transforming India

Academic Task Number: 2

Date of allotment: _____

Date of submission: _____

Academic Task Type: BYOD

Course code: INT331

Course title: Fundamentals of DEVOPS

Maximum Marks: 30

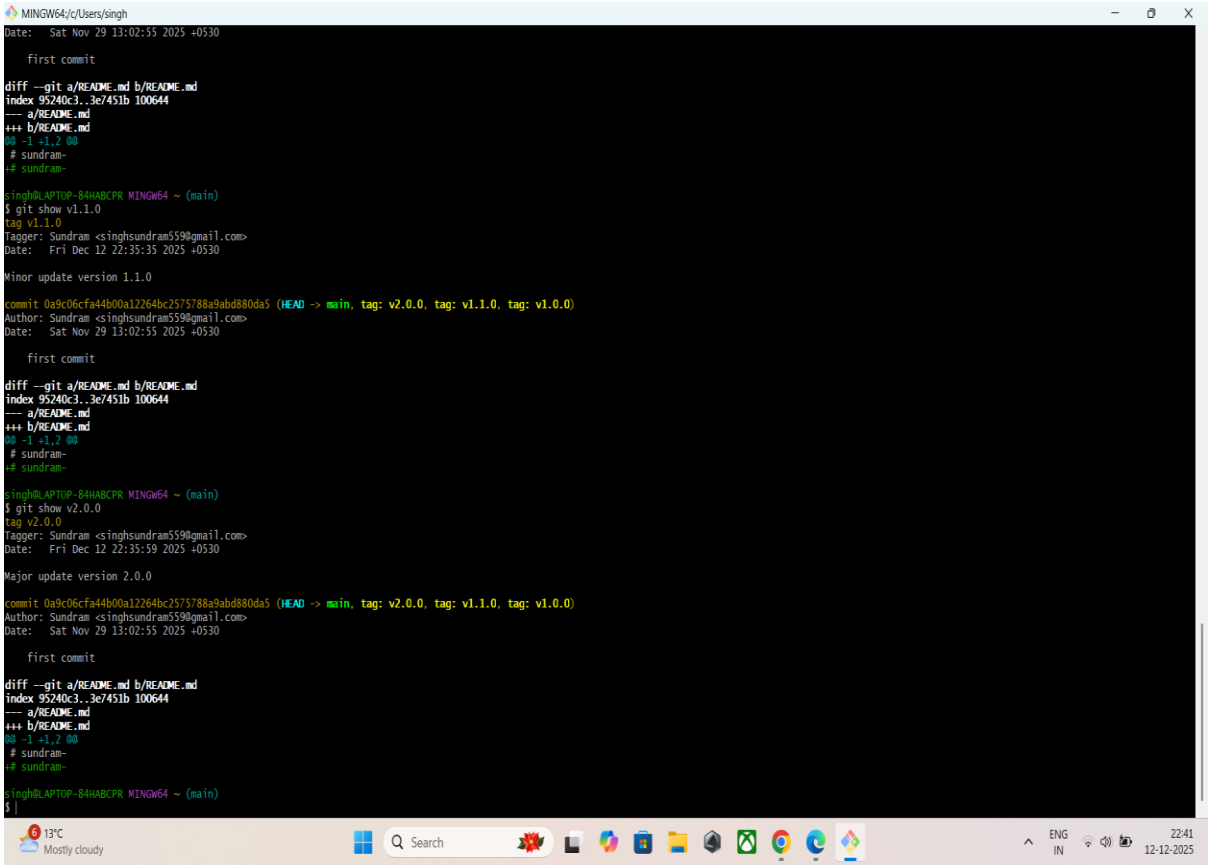
Question Number	Question Statement	Course Outcome	Bloom's level	Marks per Question
1	Your manager asks you to identify the differences between two specific commits in your Git repository to review changes made during a feature update. Which command would you use, and show the difference in file content between specific commit and stagging area copy. Write the description of each symbol associated with the output.	CO4	L3: Apply	10
2	You've completed a stable release of your project (version 1.0.0) and want to mark it in your Git repository for future reference. How can you create a tag for this version, and show how can you share it with your remote repository? Create three tags show the tags pushed in remote repository and also show the specific release of three tags.	CO4	L3: Apply	10
3	Design the architecture of Jenkins with description about the tool. Write the key features of tool.	CO5	L1: Remember	10

Page 1 of

IN QUESTION NO. 01

1. I created a new folder for the project and initialized it as a Git repository.
2. I created a new file in the project and made the first commit, which served as the base for all future tags.
3. I created three annotated tags to mark different release versions of the project:
    - Version 1.0.0 (stable release)
    - Version 1.1.0 (minor update)
    - Version 2.0.0 (major update)

4. I verified all the tags in the local repository to ensure that all three versions were successfully created.
5. I viewed the specific release information for each tag, including the tag message, author information, commit details, and file changes.
6. I connected my local repository to a remote GitHub repository by adding the remote URL.
7. I pushed the main project commit to the remote repository so that the tags could also be uploaded.
8. I pushed all three tags to the remote repository, making them visible online on GitHub.
9. I fetched and verified the tags from the remote repository to confirm that they were successfully uploaded.



IN QUESTION NO. 02

01. **I created a new branch in my Git repository** to work on a separate feature without affecting the main project.

02. **I switched to the newly created branch** so that all new changes would be recorded in that branch only.

03. **I made changes to the project files** while working inside the new branch and saved the updates.

04. **I added the modified files to the staging area** to prepare them for committing.

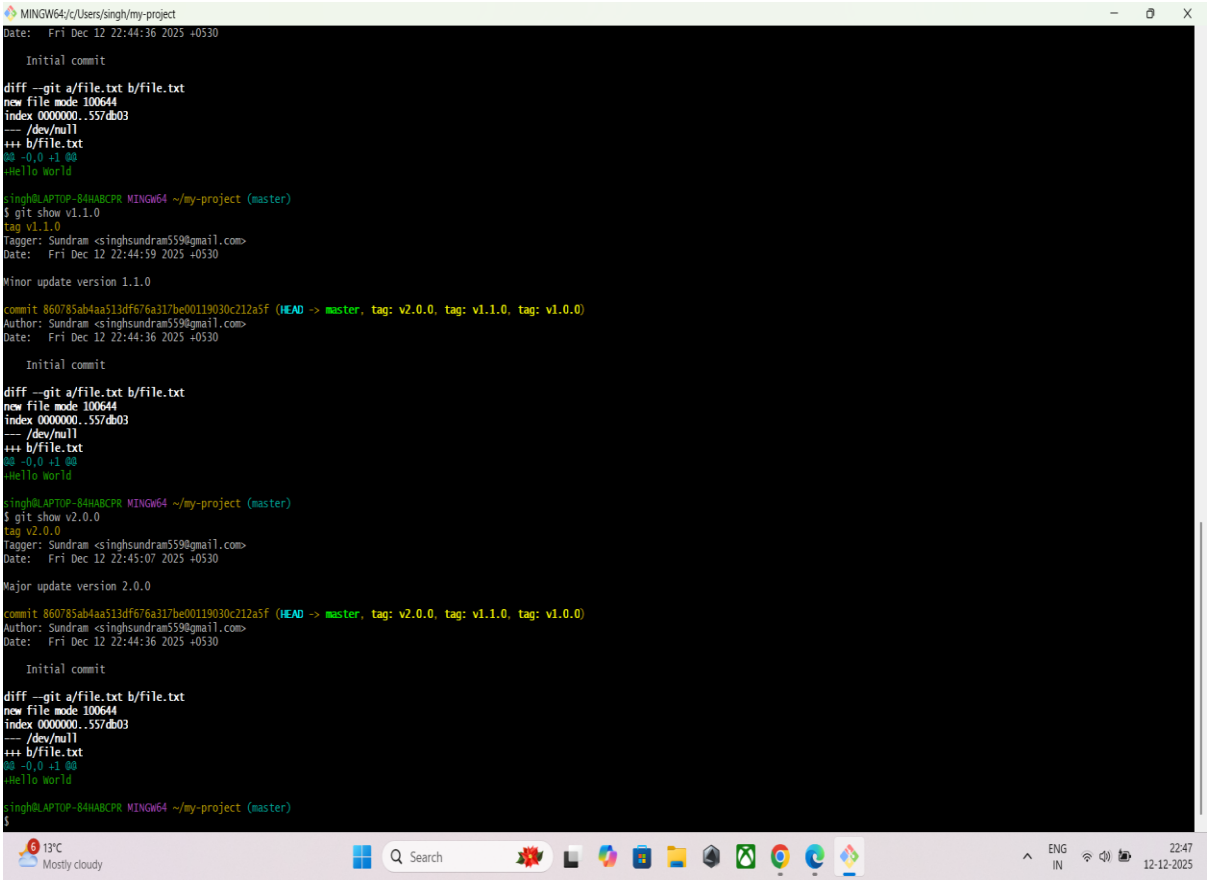
05. **I created a commit inside the new branch**, which recorded the changes made during the feature or task.

06. **I switched back to the main branch** after completing the work inside the new branch.

07. **I merged the new branch into the main branch**, combining the updated work with the stable project version.

08. **I checked the project to ensure the merge was successful** and that no conflicts occurred.

09. **Finally, I pushed both the main branch and the feature branch to the remote repository**, keeping both the updated project and branch history safely stored online.



IN QUESTION NO. 03

Step 1: Open Jenkins dashboard and log in.

Step 2: Click **New Item**, create a Pipeline project named **hello-jenkins-1**, and save.

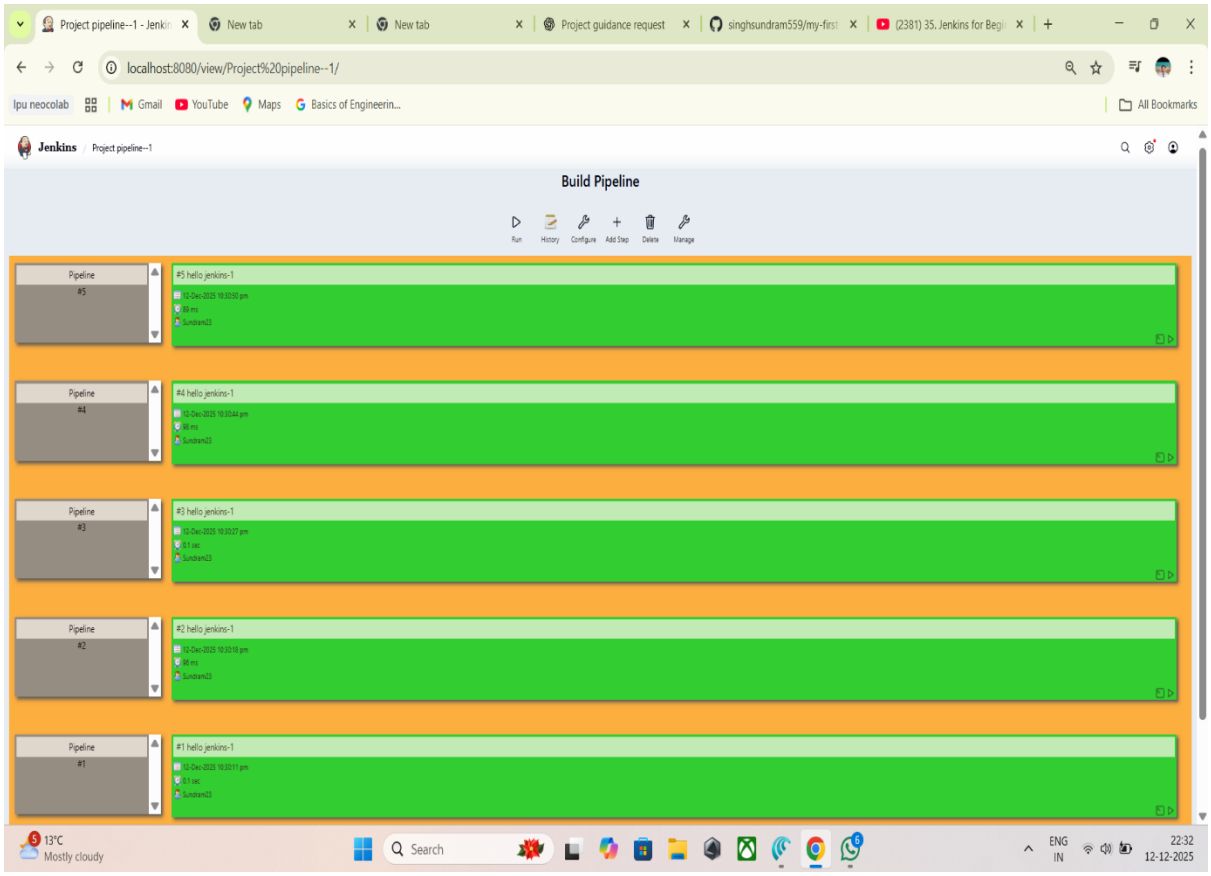
Step 3: Go to **Configure → Pipeline** and add a simple pipeline script with stages (Build, Test, Deploy).

Step 4: Click **Build Now** to run the pipeline. Jenkins creates builds (#1, #2, #3, #4, #5).

Step 5: Click any build number to open **Console Output** and view the logs.
The page also shows options like *console re-run*.

Step 6: Open **Pipeline History** to see all past pipeline runs with time and status.

Step 7: Jenkins version is shown at the bottom (Jenkins 2.528.2).



THANK YOU