**GIT Handson – 3**

* Explain branching and merging

**Solu –**

Branching in Git allows you to create an independent line of development from your main codebase. Think of it like a parallel universe for your project. When you create a new branch, you're essentially making a copy of the project's current state. This allows you to work on new features, experiment with ideas, or fix bugs without affecting the stable, main version of your code. Branching is a powerful feature that enables multiple developers to work on different tasks simultaneously without interfering with each other's work.  
  
Merging is the process of combining the changes from one branch into another. Once you've completed your work on a feature or bug fix in a separate branch, you can merge those changes back into the main branch. When you perform a merge, Git automatically attempts to integrate the changes. If the same part of a file was changed on both branches, a merge conflict will occur, and you'll have to manually resolve the differences before the merge can be completed.

* Explain about creating a branch request in GitLab

**Solu –**

**Branching and merging** are core features in Git. **Branching** is the process of creating an independent line of development from the main codebase. This allows you to work on new features, fix bugs, or experiment without affecting the stable, main version of your code. **Merging** is the process of combining the changes from a completed branch back into another branch, typically the main one.

**Creating a Branch and Merge Request in GitLab**

GitLab, being a web-based Git repository, provides a user-friendly interface to manage these tasks. A **merge request** in GitLab is a request to merge one branch into another. It's the GitLab equivalent of a pull request in GitHub and is used for proposing changes, code review, and discussion before the code is merged.

Here's how to create a branch and a merge request in GitLab:

1. **Create a New Branch**:
   * Navigate to your project in the GitLab UI.
   * On the left sidebar, go to **Code** > **Branches**.
   * In the top-right corner, click **New branch**.
   * Give your branch a descriptive name (e.g., feature/add-user-login).
   * Select the branch you want to base your new branch on (usually the main or master branch).
   * Click **Create branch**.
2. **Make and Push Your Changes**:
   * Clone the repository to your local machine or use the Web IDE.
   * Switch to your newly created branch: git checkout <branch-name>.
   * Make your changes and commit them: git commit -m "My descriptive commit message".
   * Push your branch to the GitLab repository: git push origin <branch-name>.
3. **Create a Merge Request**:
   * After you push your new branch, GitLab will often show a notification that a new branch has been pushed and will prompt you to **Create merge request**.
   * Alternatively, go to **Code** > **Merge requests** and click **New merge request**.
   * Select your new branch as the **Source branch** and the branch you want to merge into (e.g., main) as the **Target branch**.
   * Fill out the merge request details, including a title and a description of your changes.
   * You can assign a user to review your changes and add a description to provide more context.
   * Click **Create merge request**.

Explain about creating a merge request in GitLab

**Solu –**

1. Push your branch: After you've finished your work and committed your changes on your local branch, push it to the GitLab remote repository using the command: git push origin <your-branch-name>.
2. Navigate to the project: Open your project in the GitLab web interface. You'll often see a notification at the top of the project's page saying that a new branch has been pushed and prompting you to Create merge request.
3. Fill out the details:
   * Source branch: This is the branch with your changes. GitLab usually auto-selects your newly pushed branch.
   * Target branch: This is the branch you want to merge your changes into, typically main.
   * Title and Description: Provide a clear title and a detailed description of your changes. Explain what problem you solved or what feature you added.
   * Reviewers: Assign team members to review your code.
   * Labels and Milestones: Add any relevant labels or link it to a milestone for better organization.
4. Create the merge request: Click the Create merge request button. This will open the merge request page where your team can review the code, leave comments, and approve the changes.
5. Merge when ready: Once the merge request is approved and any conflicts are resolved, you can click the Merge button to integrate your changes into the target branch.