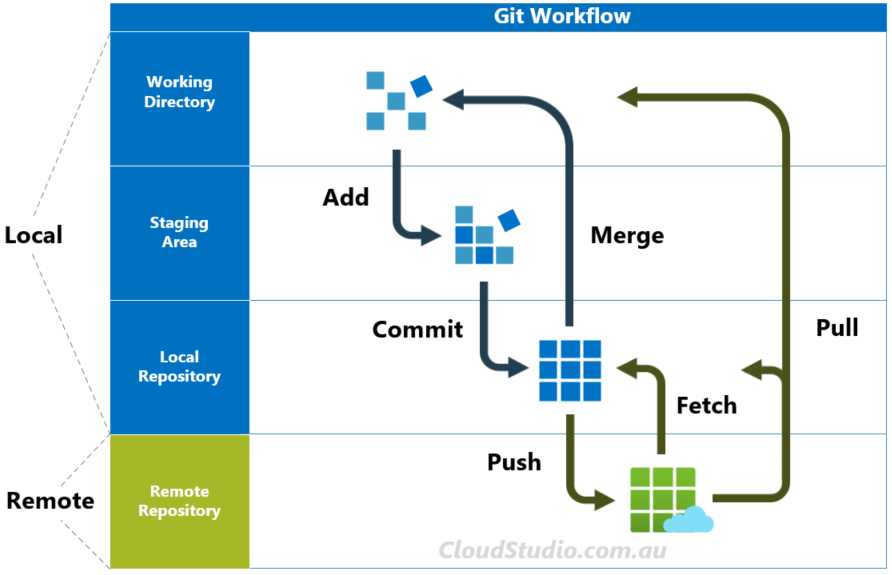
**05/03/2025**

**Git Important topics:**

1. **Explain the workflow of git with a neat diagram**

Git is version control system. It is used to track of the changes and history of projects. Developers work on their local repositories and synchronize changes with a remote repository

**Work flow:**



**Step1:** Cloning

git clone :- clone the repository from remote repository to local machine

**Step2:** Create a new branch

git checkout -b branch\_name: create a new branch and switches to it

**Step3:** stage files

git add file\_name : make changes in file and add the file to repository

**step4**: commit changes

git commit -m “message”: once staged, commit the changes with a message

**step5:** push the changes to remote server

git push origin branch\_name: push the changes to the remote repository

**step6:** create a pull request or merge

git checkout main

git merge branch\_name

a pull request is created for code review

once approved, the branch is merged into the main branch

**step7: pull changes**

**git pull origin main:** to stay updated with remote changes, fetch and merge updates

1. **What is git and how to stage a file with git add**

Git is a distributed version control system that helps developers track changes in their code, collaborate efficiently, and manage different versions of a project. It allows multiple people to work on the same project without overwriting each other’s changes.

**Staging:**

**git add file\_name:** this command is used to move a file from local repository to the staging area

**git add file\_name1 file\_name2…file\_namen:** this command is used to move a files from local repository to the staging area

**git add –all:** this command is used to move all files from local repository to the staging area

**git status:** shows changes in working directory

git commit -m “message”: this command records the changes permanently in the version history

git push origin master: upload the changes to github

1. **Difference between git, GitHub, Gitlab**

**Git:**

Git stands for global information tracker

Git is a open source distributed version control system

Git is used track changes in source code

Git allows developers to work independently with a full copy of the codebase on their local machines

Git is light weight and fast

Git's powerful branching model enables developers to create separate branches for features, bug fixes, or experiments, seamlessly merging them back into the main codebase after review.

**GitHub:**

GitHub is a cloud-based hosting service that provides a user friendly interface for managing git repositories

It is used to store, share and collaborate on their codebase with teams or the open-source community

GitHub functionality is hosting git repositories remotely, providing a centralized location for developers to push and pull code changes

GitHub collaboration by allowing developers to follow project contribute code and interact through discussions, issues and pull requests

GitHub offers integrated project management tools, such as boards, wikis, and project tracking, to help teams organize and manage their development workflows.

**GitLab:**

Gitlab is a web-based platform that streamlines development workflows. It does this by merging git repository management with continuous integration, deployment, and collaboration tools.

Gitlab is powerful DevOps platform

Gitlab provides a central location for hosting git repositories, enabling teams to collaborate on code and manage version control

Gitlab has built-in CI/CD pipelines, allowing teams to automate the entire software delivery process, from code commit to production deployment.

GitLab provides a comprehensive DevOps toolchain, including features for container management, monitoring, and security scanning. This streamlines the entire development lifecycle within a single platform

1. How to create a tag how can we push it?

Git tag is used to mark specific commits, usually for releases

**Create a tag**

git tag name: eg git tag v1.0.0: lightweight tag

git tag -a v1.0.0 -m “Release version” : includes metadata like author, date and message

**to see the all tags use the below command**

**git tag**

**push tag to remote repository**

git push origin v1.0.0

**checkout a specific tag**

git checkout v1.0.0

git checkout -b new-branch v1.0.0

**delete a tag**

git tag -d v1.0.0

**delete from remote repository**

git push origin --delete v1.0.0

1. **explain the steps which are necessary to send a project from local repository to remote repository**

**step1:** git init

Initialize empty repository. This creates a .git directory in your project folder.

**Step2:** git add .

Add all files to the staging area

**Step3:** git commit -m “message”

Records the changes in version history permanently

**Step4:** Create remote repository

Create a new repository in GitHub

Copy the link

**Step5:** git remote add origin url:

Used to connect the local repository to remote repository

**Step6:** push the local repository to remote repository

git push **-**u origin master: uploads the changes in GitHub

1. **What is a pull request explain its procedure**

Pull request is used to allows developers to propose changes to a repository. It enables code review, discussion and merging before integrating the changes into the main branch

Step1: clone the repository

Step2: create a new branch

Step3: make changes and commit

Step4: push changes to remote from local

Step5: create a pull request

* Go to the repository on GitHub
* Click on new pill request
* Select your source branch and target branch
* Add a title and description explaining the changes
* Submit the pull request

1. What is fork why is it necessary?

Fork is a copy of a repository that is created under a different users account allowing independent notifications without affecting the original project. It is commonly used in open source contributions

Why forking:

* We can freely modify a project without needing direct permission from the original repository
* We can test new features without affecting the main project
* Maintain our own versions of a project with custom changes
* Keep a copy of the repository in case the original is deleted

**Differences between Linux and windows operating system**

**Linux:**

* Linux is a open source operating system
* Linux is free of cost
* Linux uses a monolithic kernel
* In Linux, file names are case sensitive
* In Linux we have same file names with different cases
* More efficient and fast
* Directories are separated by forward slashes
* More secure
* Root user has all administrative privileges
* Linux has 3 types of user accounts

1. Regular
2. Root
3. Service account

**Advantages:**

1. Open source
2. Reliability
3. Licensing cost
4. Backward compatability

**Disadvantages:**

1. Lack of standards
2. Support costs
3. Proprietary software
4. Steep learning curve
5. Unsupported hardware

**Windows:**

* Window is not a open source
* Windows is a paid operating system
* Windows uses a hybrid kernel
* In windows, files are not case sensitive
* In windows we can not have the same file names
* Less efficient
* Directories are separated by backward slash
* Less secure
* Administrative user has all administrative privileges
* Windows has 4 types of user accounts
* Administrator
* Standard
* Child
* Guest