Git Tagging Explained

In Git, tagging is used to mark specific points in the commit history, typically for version releases like v1.0, v2.1, etc. Tags help identify important commits and are commonly used in software versioning.

Types of Git Tags

- Lightweight Tag

```
git tag v1.0
```

Simple pointer to a commit without extra metadata.

- Annotated Tag

```
git tag -a v1.0 -m "Version 1.0 release"
```

Stores author, date, and a message with the tag.

- Signed Tag

```
git tag -s v1.0 -m "Signed version 1.0 release"
```

Cryptographically signed using a GPG key.

Working with Tags

- List all tags

```
git tag
```

- View details of a tag

```
git show v1.0
```

- Push a specific tag to GitHub

```
git push origin v1.0
```

- Push all tags to GitHub

```
git push origin --tags
```

- Checkout a specific tag

```
git checkout v1.0
```

- Delete a local tag

```
git tag -d v1.0
```

- Delete a remote tag

```
git push origin --delete v1.0
```

Example Use Case

- You have completed a new feature for a project.
- You create a tag like v1.0 to mark the release version.
- Later, users can easily access version v1.0 via:

  git checkout v1.0

- If a bug is found, you create a new tag (v1.1) after fixing it.