### Revision Notes for Examination

#### 1. Well-Known Version Control Systems

- \*\*Git\*\*: Distributed version control, fast, efficient, supports branching and merging.

- \*\*CVS\*\*: Centralized, early version control system.

- \*\*Mercurial\*\*: Distributed, similar to Git, known for simplicity.

- \*\*SVN (Subversion)\*\*: Centralized, a successor to CVS.

- \*\*GitHub\*\*: Cloud-based platform for hosting Git repositories.

---

#### 2. Git Basic Concepts

- \*\*Repository\*\*: Collection of files tracked by Git.

- \*\*Commit\*\*: Snapshot of changes in the repository.

- \*\*Branch\*\*: Independent line of development.

- \*\*Merge\*\*: Combining changes from one branch into another.

- \*\*Remote\*\*: Copy of the repository hosted on a server.

- \*\*Pull\*\*: Fetch + Merge changes from the remote repository.

---

#### 3. Git Architecture

- \*\*Working Directory\*\*: Current version of files.

- \*\*Staging Area\*\*: Prepares changes for commit.

- \*\*Repository\*\*: Stores the full history of changes.

---

#### 4. Git Workflow

1. \*\*Create a New Branch\*\*: Work independently without affecting the main branch.

2. \*\*Make Changes\*\*: Modify code.

3. \*\*Commit Changes\*\*: Save changes with a descriptive message.

4. \*\*Push Changes\*\*: Upload commits to a remote repository.

5. \*\*Create Pull Request\*\*: Request to merge changes into the main branch.

6. \*\*Review Changes\*\*: Team members review and comment.

7. \*\*Merge Changes\*\*: Approved changes are merged.

8. \*\*Update Local Repository\*\*: Pull the latest main branch changes.

---

#### 5. Basic Git Commands

- `git init`: Initialize a new repository.

- `git clone <URL>`: Copy a remote repository.

- `git add <file>`: Stage changes.

- `git commit -m "message"`: Commit changes.

- `git push`: Upload commits to remote.

- `git pull`: Update local repo with remote changes.

- `git status`: Check the status of changes.

- `git branch <name>`: Create a branch.

- `git merge <branch>`: Merge changes.

---

#### 6. Git Branching

- \*\*Create\*\*: `git branch <name>`

- \*\*Switch\*\*: `git checkout <name>`

- \*\*Delete\*\*: `git branch -d <name>`

- \*\*Merge\*\*: `git merge <branch>`

- \*\*List\*\*: `git branch`

---

#### 7. GitHub Key Features

- Version control, hosting, collaboration tools (issues, pull requests).

- Continuous integration and deployment tools.

- Social coding with stars and forks.

---

#### 8. Benefits of Version Control

- \*\*Collaboration\*\*: Multiple developers can work simultaneously.

- \*\*History Tracking\*\*: Detailed log of changes.

- \*\*Branching & Merging\*\*: Develop features independently.

- \*\*Backup\*\*: Recover from errors easily.