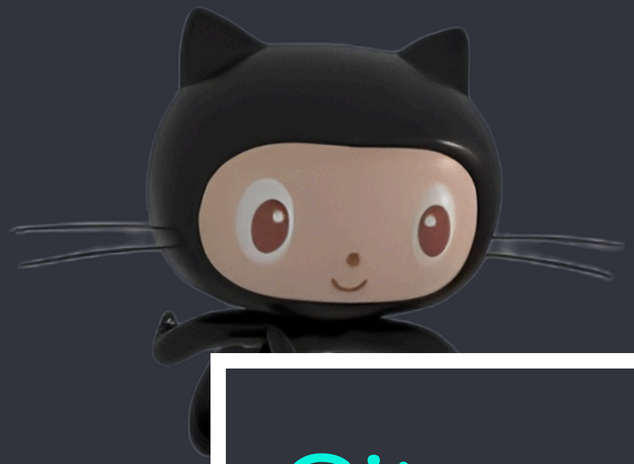
IT Club



Git and GitHub

Day-2

Remote | Branching | Merging

Nibida Ghimire | Samriddhi Karki

Day-2

IT Club

- **Basic Introduction to remote**
  - Git vs GitHub
  - Creating a GitHub account
  - Pushing our local repo to GitHub
  - Connecting a local repository to a remote (git remote add)
  - Pushing to a remote repo (git push)
  - Pulling changes from remote repo (git pull)
  - Keeping our secrets secret with .gitignore

- **Branching and Merging**
  - Understanding Branches
  - Creating/Switching Branches
    - (git branch, checkout/switch)
  - Rebase
  - Squash
  - Hands-on practice

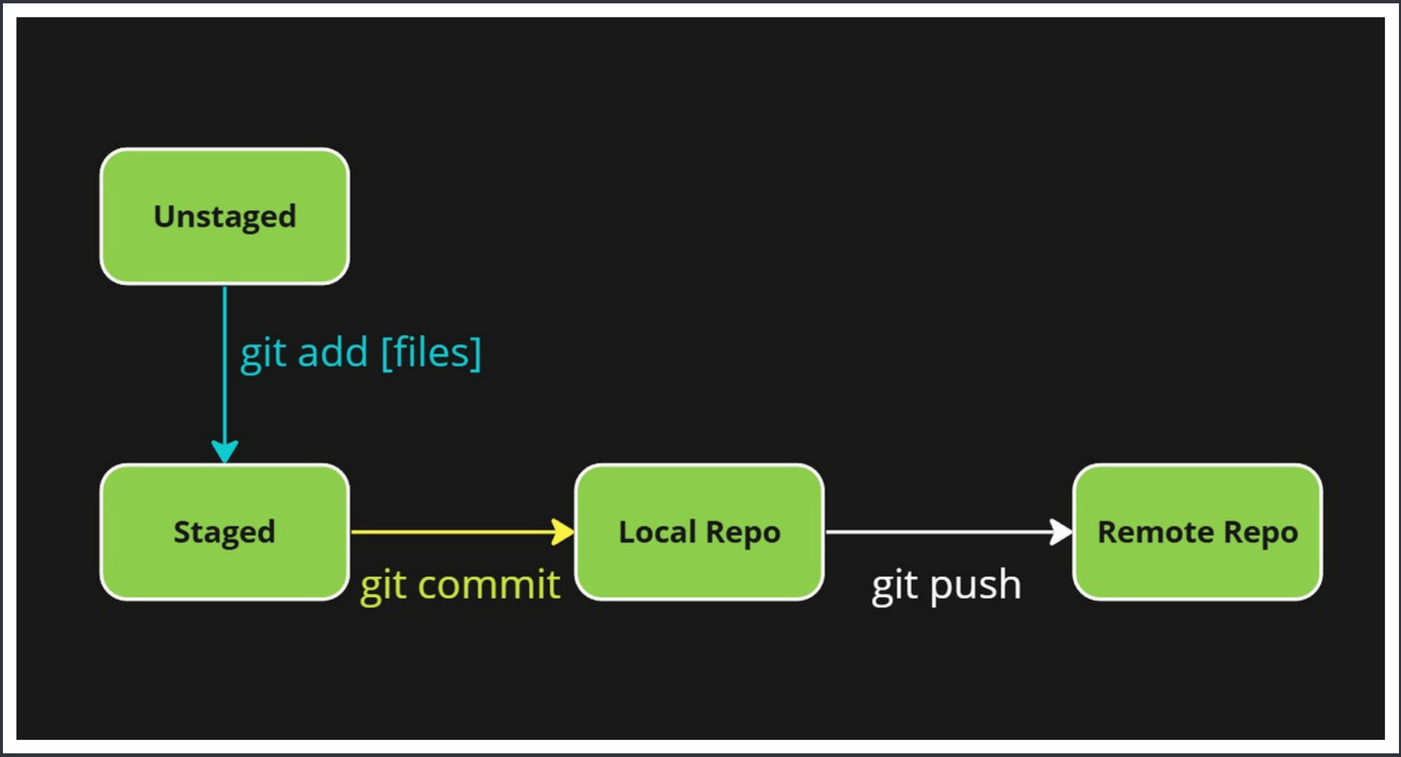
- **Merge Conflicts**
  - Understanding merge conflicts
  - Strategies to solve merge conflicts
  - Hands-on practice

IT Club

Git vs GitHub

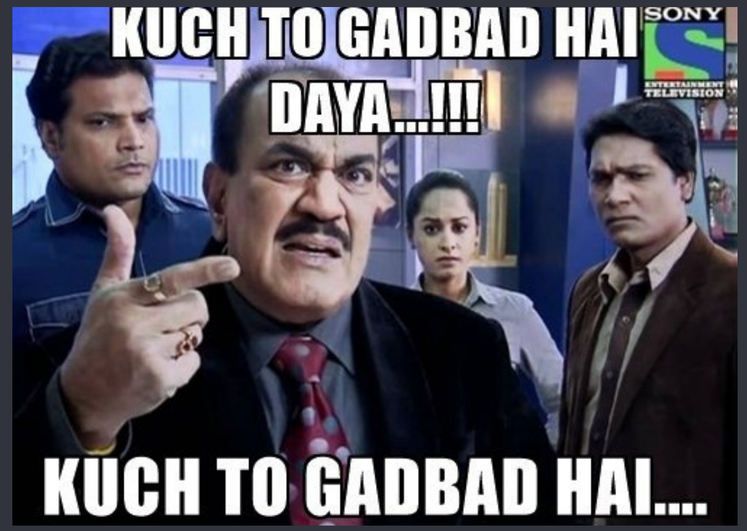
IT Club

Workflow



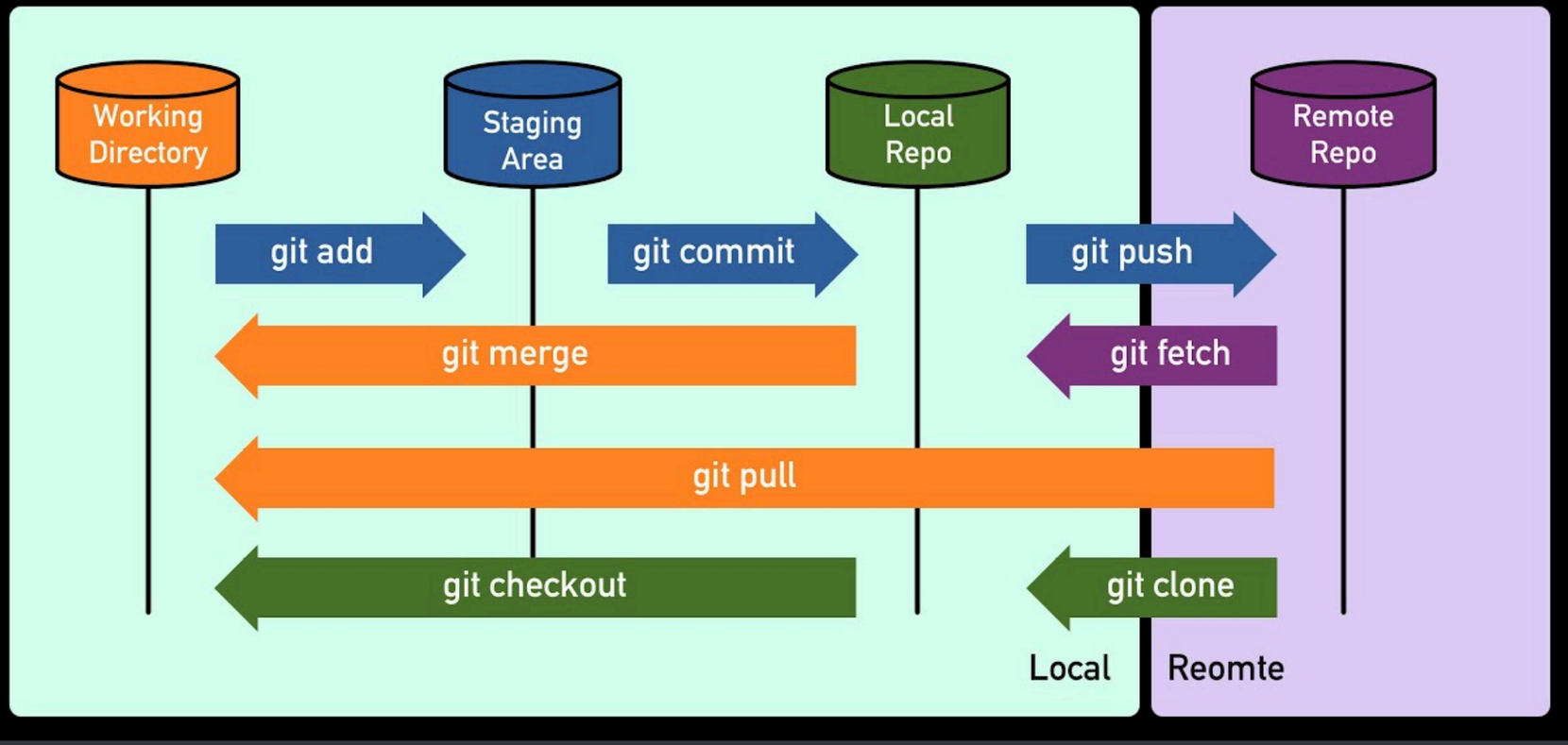
Add --> Commit --> Push

ACP



IT Club

How Git Actually Works



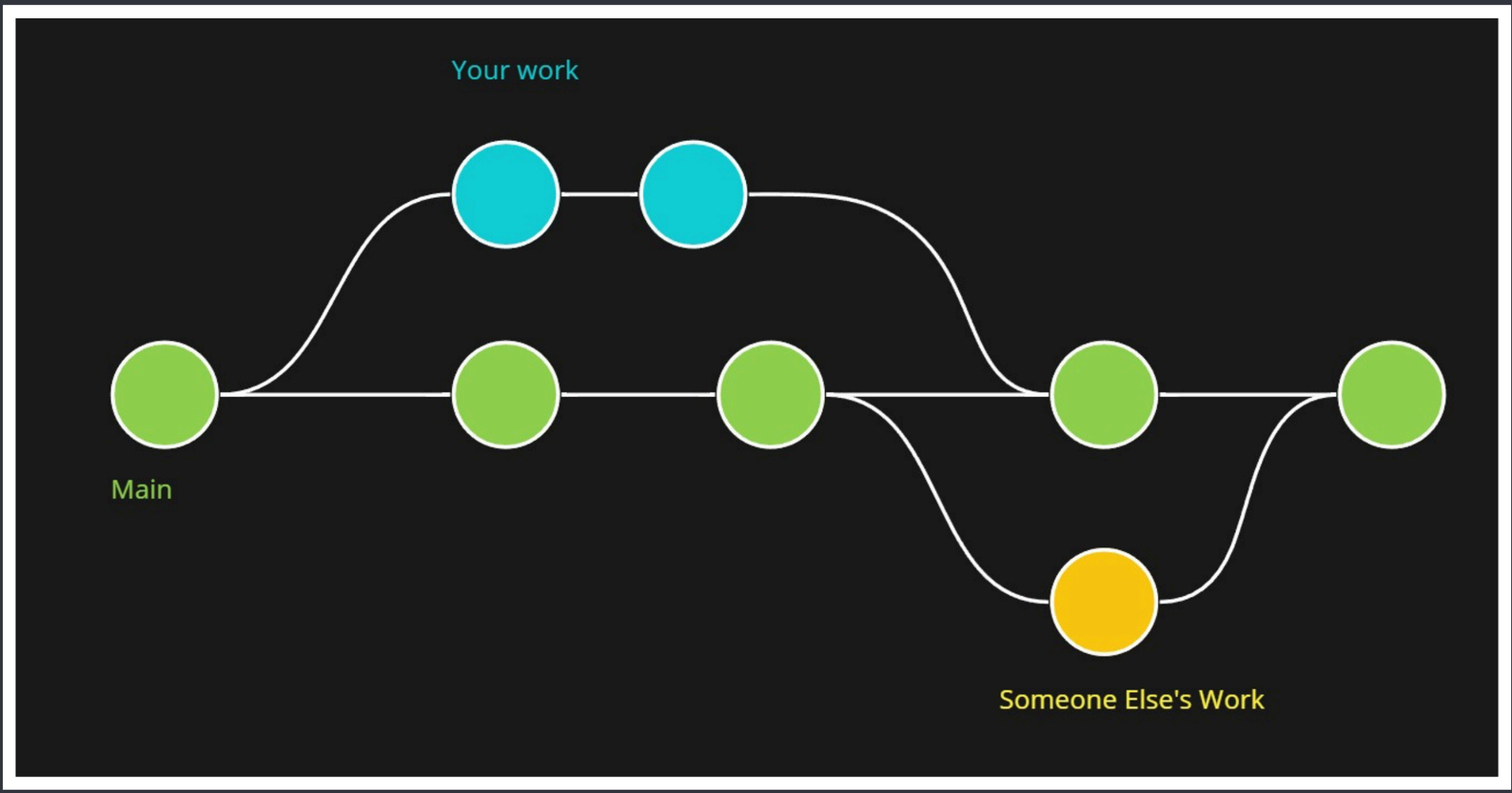
IT Club

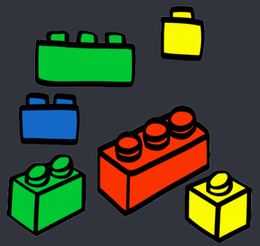
Git Ignore

Files included in .gitignore file are ignored while committing and pushing the code.

IT Club

Branching And Merging

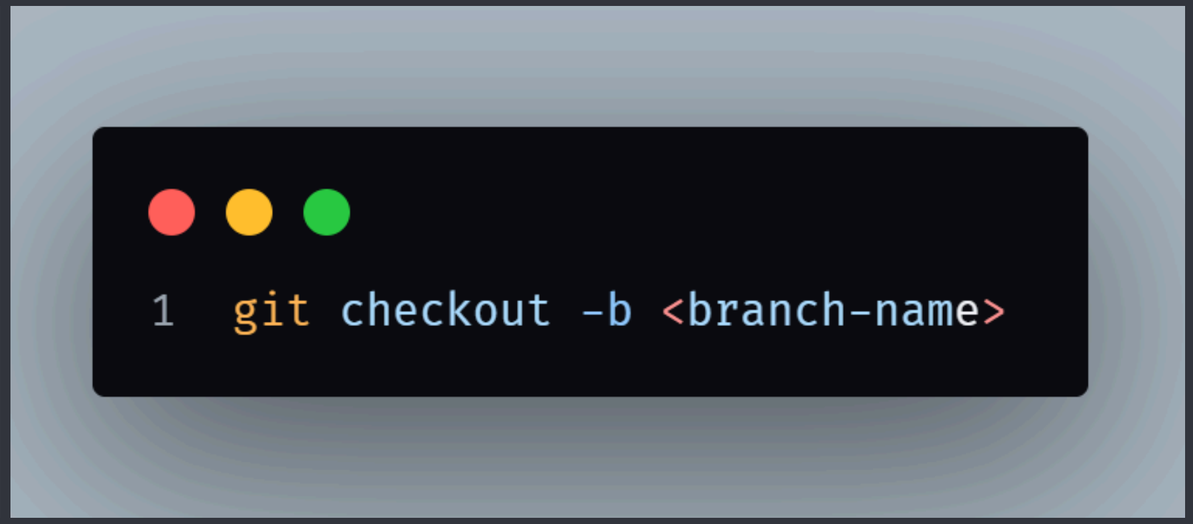




IT Club

Branching

Branching & Accessing the Branch



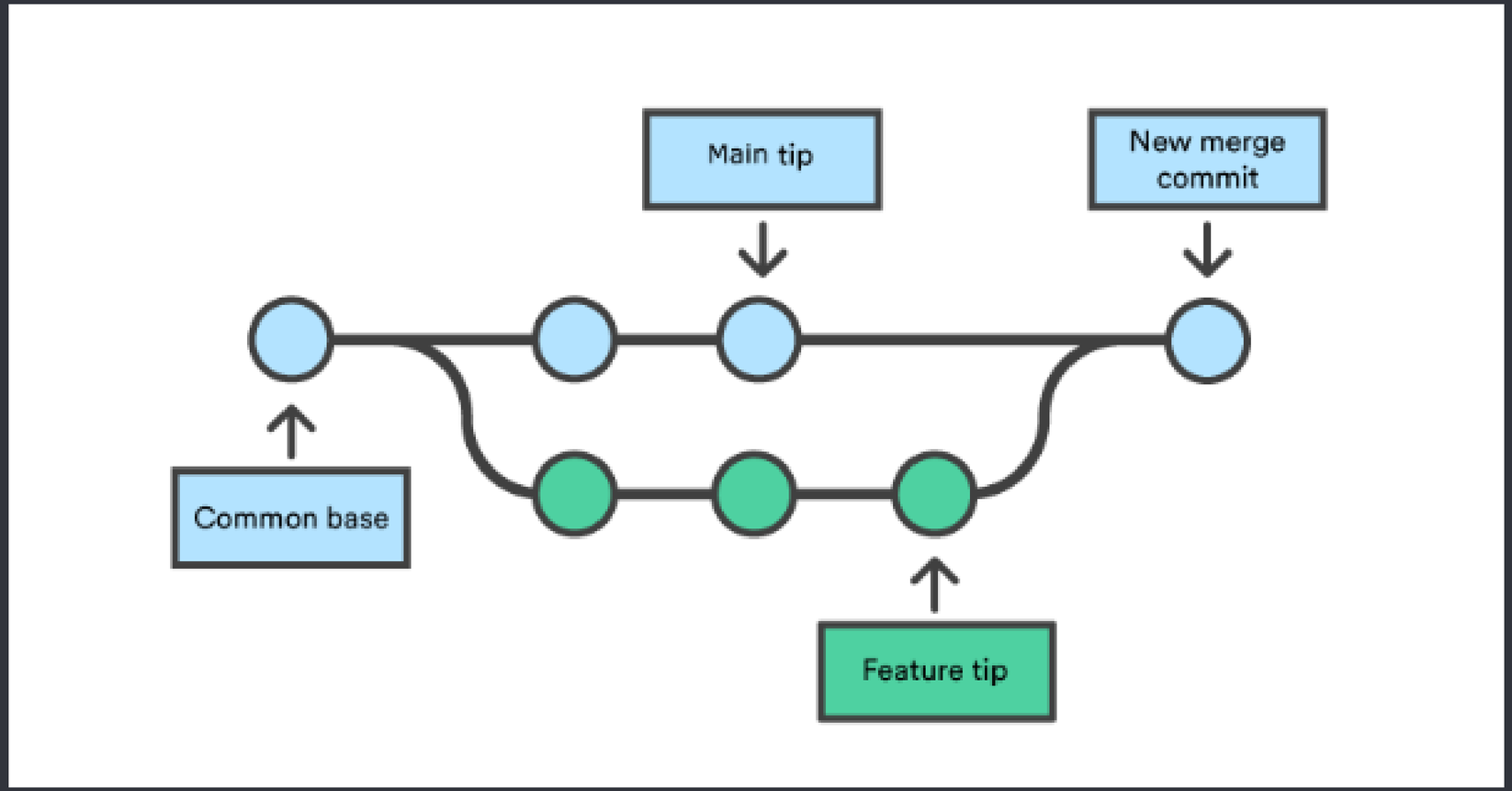
Single Line Branching & Accessing

IT Club

Naming Conventions

1. Use descriptive and meaningful names for branches, such as: "feature/login-page", "fix/bug-123"
2. Use hyphens or slashes to separate words in branch names
3. Use prefixes like "feature/", "fix/", "hotfix/" to indicate the purpose of the branch
4. Use issue or ticket numbers in branch name for better tracking and traceability.

IT Club



IT Club

Merge

- Integrates changes from one branch.
- Combines changes from a feature or development branch into the main or release branch.
- Preserves the history of changes and ensures collaboration among team members.
- Allows developers to incorporate new features, bug fixes, or updates into the target branch

IT Club

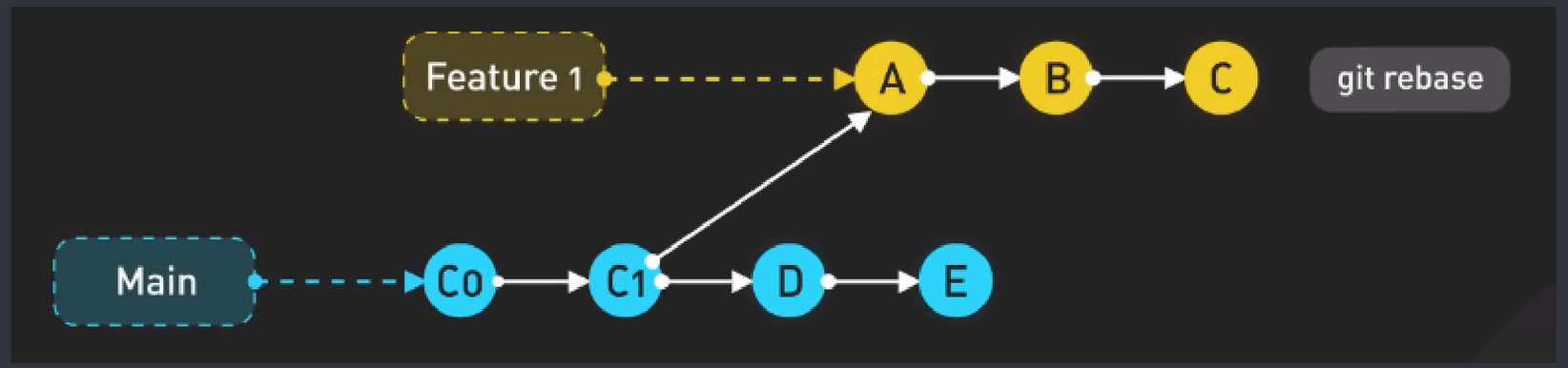

git merge <branch-name-to-merge>
#example:
git checkout main
git merge <branch-name>

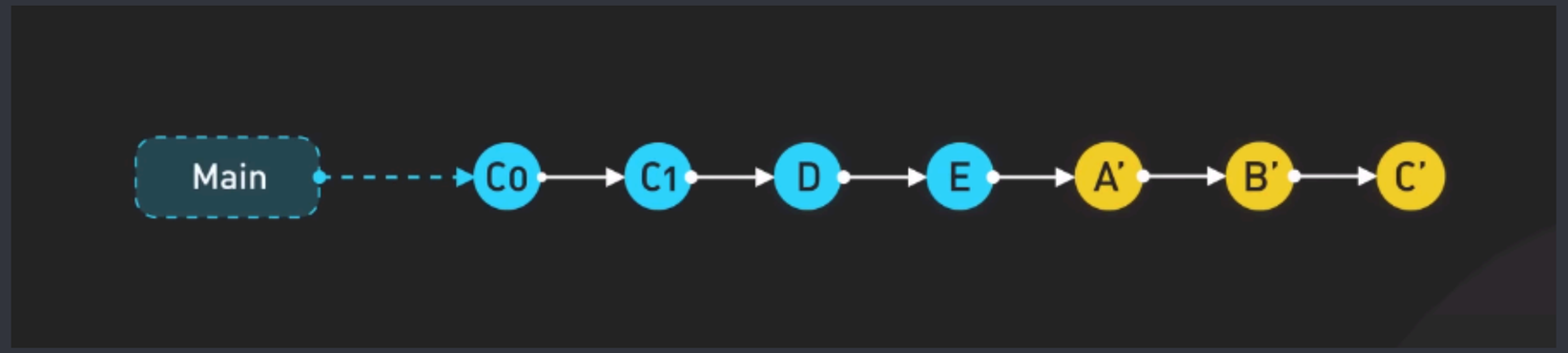
IT Club

Rebase

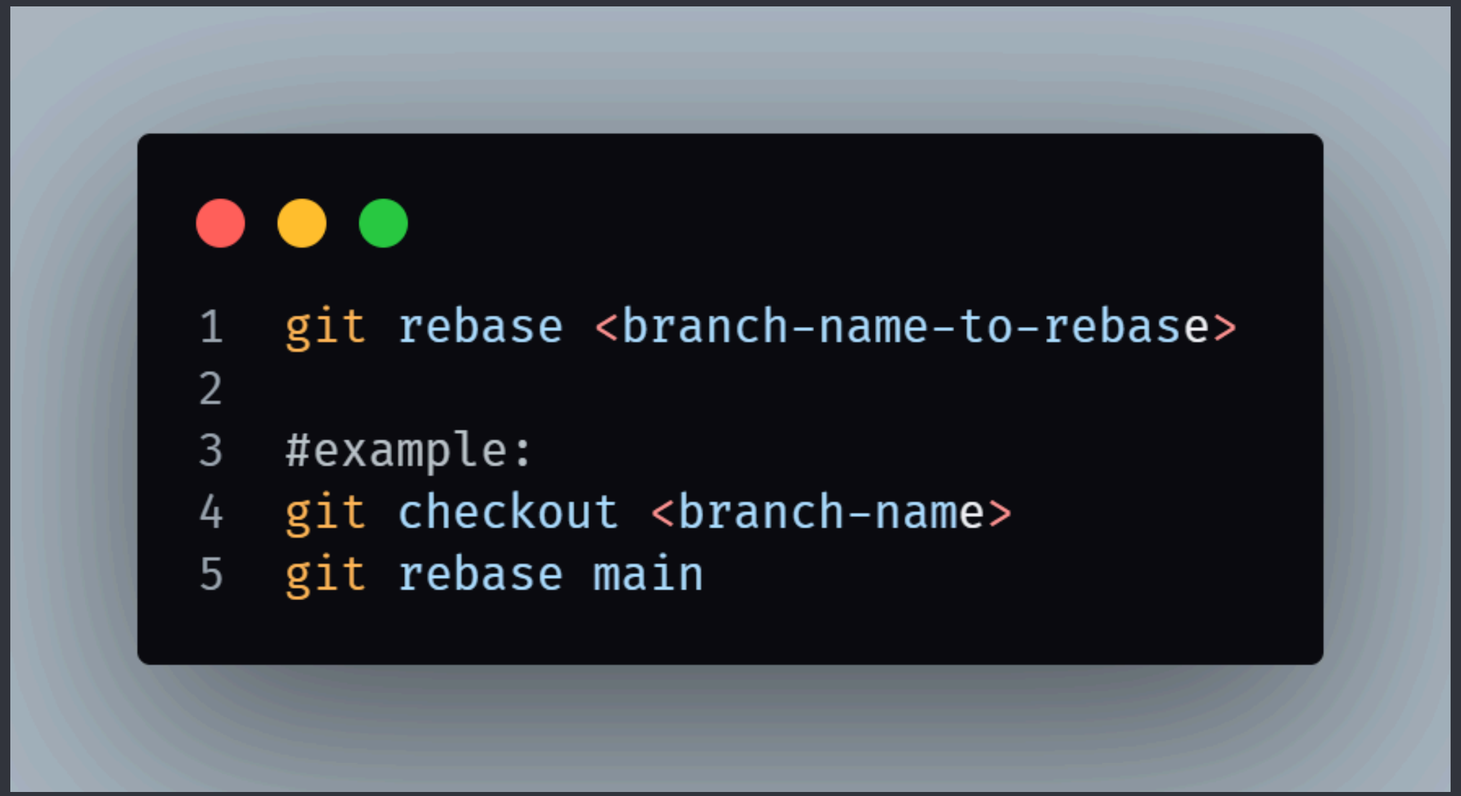
- Process of moving or combining a sequence of commits to a new base commit.
- Rewrites the commit history of a branch by placing it on top of the target branch.
- Useful for maintaining a clean and linear commit history, especially in collaborative environments

IT Club





IT Club



```
git rebase <branch-name-to-rebase>

#example:
git checkout <branch-name>
git rebase main
```

IT Club

Squash

- Combines multiple commits into a single commit.
- Helps to condense the commit history, making it more concise and easier to follow.
- Useful for tidying up a series of minor or related commits before merging into the main branch.

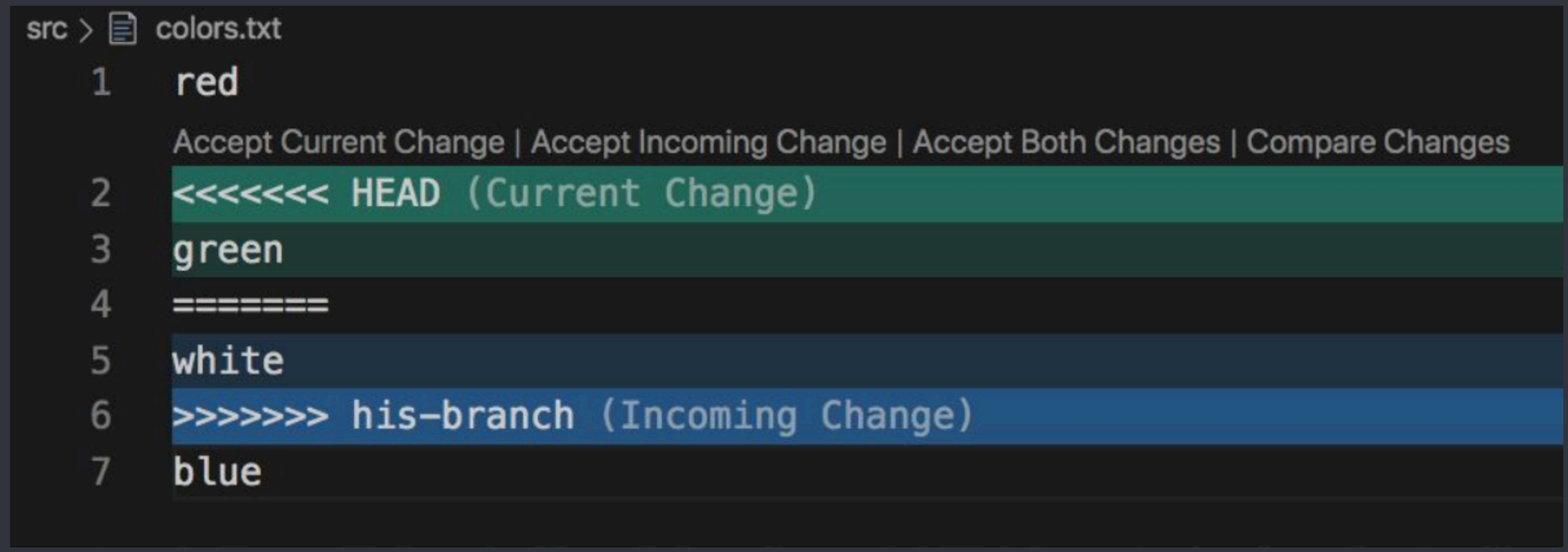
IT Club

Conflicts

- Generally arise when two people have changed the same lines in a file, or if one developer deleted a file while another developer was modifying it.
- Git cannot automatically determine what is correct.
- Git will mark the file as being conflicted and halt the merging process.

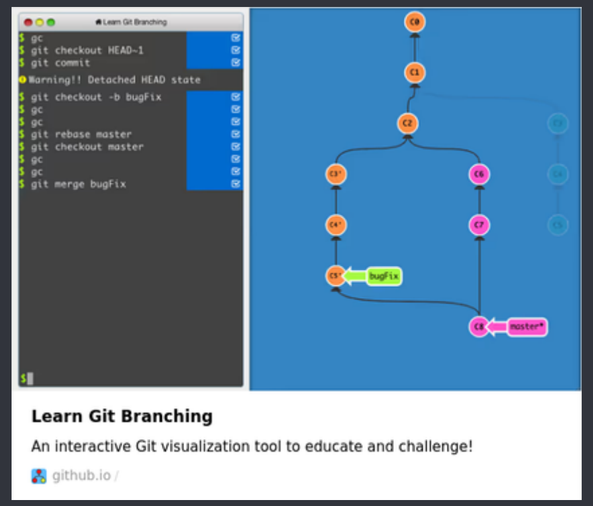
IT Club

Resolving Conflicts



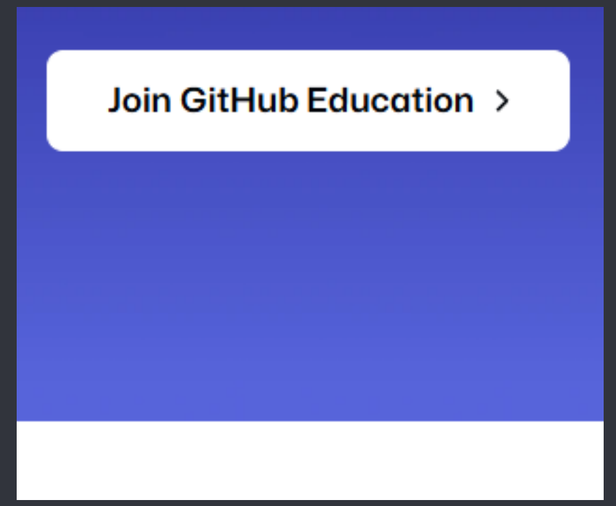
IT Club

Resources



https://learngitbranching.js.org/

https://education.github.com/git-cheat-sheet-education.pdf



https://github.com/edu/students

IT Club

Thank

You!