CSE HACKS SOLUTION

GIT / GITHUB

MASTERING GIT: A VERSION CONTROL POWER TOOL

Created by MAHADI HASAN MISHUK

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

INTRODUCTION TO VERSION CONTROL

Definition of version control

Version control is a system that tracks changes to files and directories over time, enabling multiple contributors to collaborate on a project while maintaining a historical record of revisions

Importance in software development

Version control is crucial in software development as it facilitates collaborative work, ensures code integrity, and enables easy tracking of changes, making it a fundamental tool for efficient and organized development

Problems version control solves

Version control solves issues like data loss, code conflicts, and the lack of a documented history by providing a structured way to manage and coordinate changes in a project, improving collaboration and accountability.

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

WHAT IS GIT?

What is Git?

Git is a distributed version control system (DVCS) that enables users to track changes, collaborate on projects, and manage source code efficiently

Git as a distributed version control system

Git is a DVCS that allows multiple contributors to work on a project independently, with their own local repositories, while still being able to synchronize and share changes with others, enhancing flexibility and redundancy.

Git's popularity

Git has gained immense popularity and is widely used in software development due to its speed, flexibility, and the availability of online platforms like GitHub and GitLab, making it a standard tool for version control in the industry.

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

WHY GIT?

Using Git offers several advantages over other version control systems

Scalability

Git is adaptable to projects of all sizes, from small personal endeavors to large-scale software development efforts, making it versatile and efficient.

Speed

Git's local operations ensure rapid and responsive version control actions, enhancing developer productivity and reducing waiting times.

Branching Capabilities

Git's branching and merging features empower developers to work on separate tasks in isolation, facilitating collaboration and experimentation while maintaining code stability.

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

BASIC GIT COMMANDS

1. git init

Initializes a new Git repository in the current directory.

bash

```
git init
```

2. git clone

Creates a copy of a remote repository on your local machine.

bach

```
git clone https://github.com/username/repository.git
```

3. git add

Stages changes for committing to the repository.

bash

```
git add filename.txt
```

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

BASIC GIT COMMANDS

4. git commit

Records the changes staged using git add to the repository.

bash

```
git commit -m "Added a new feature"
```

5. git push

Uploads your local changes to a remote repository.

bach

```
git push origin main
```

6. git pull

Fetches and merges changes from a remote repository to your local repository.

bash

```
git pull origin main
```

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

BASIC GIT COMMANDS

7. git branch

Lists, creates, or deletes branches in the repository.

bash

```
List branches: git branch
Create a new branch: git branch new-feature
Delete a branch: git branch -d old-feature
```

8. git checkout

Switches to a different branch or commit.

bach

```
Switch to a branch: git checkout feature-branch
Switch to a specific commit: git checkout commit-hash
```

9. git merge

Combines changes from one branch into another.

bash

```
git checkout main
git merge feature-branch
```

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

BASIC GIT COMMANDS

10. git status

Shows the status of files in your working directory, including changes to be

bash

```
bash git status
```

11. git log

Displays a log of all commits in the repository.

bach

```
bash git log
```

12. git remote

Lists and manages remote repositories.

bash

```
List remotes: git remote -v
Add a new remote: git remote add origin
https://github.com/username/repository.git
```

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

BASIC GIT COMMANDS

13. git fetch

Downloads objects and refs from another repository, but does not merge changes.

bash

```
bash git fetch origin
```

14. git push

Pushes your changes to a remote repository.

bash

```
bash git push origin main
```

15. git pull

Fetches and merges changes from a remote repository.

bash

```
bash git pull origin main
```

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09

BASIC GIT COMMANDS

16. git remote show

Displays detailed information about a remote.

bash

```
bash git remote show origin
```

17. git log

Shows a commit log with options for formatting and filtering.

bach

```
bash git log --oneline --graph
```

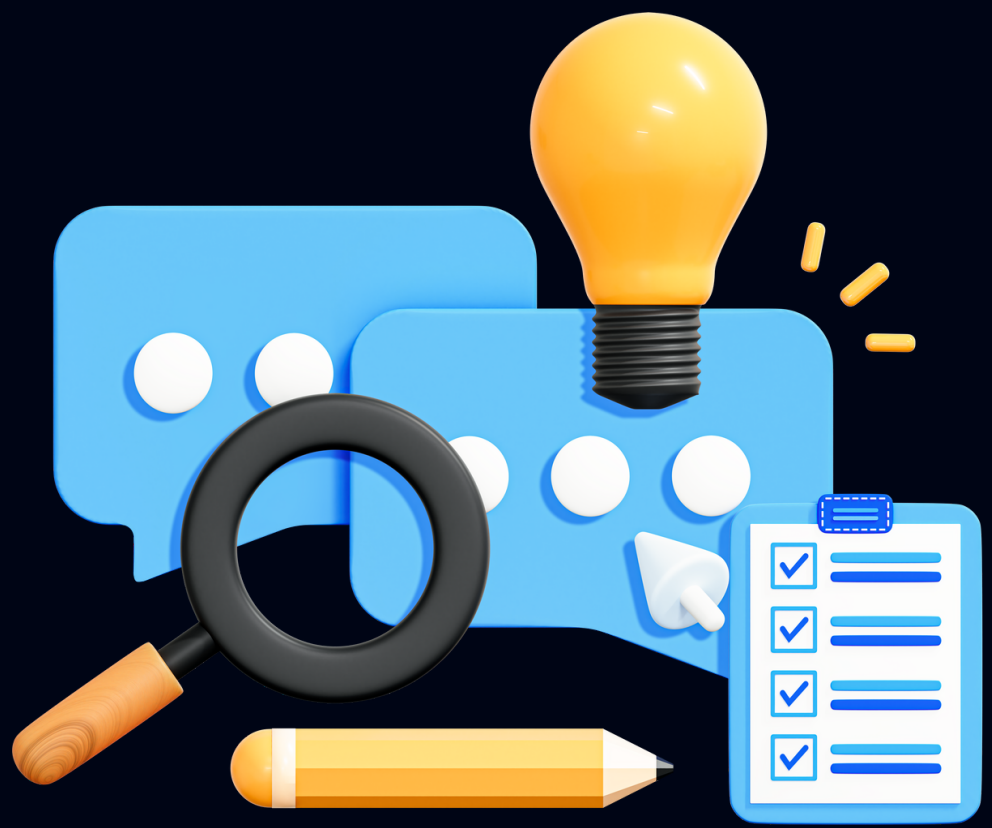
18. git stash

Temporarily saves changes that are not ready for commit.

bash

```
bash git stash
```

mishukinfo09@gmail.com
www.mahadiinfo.me/Portfolio
www.github.com/mishuk09



I hope this information is helpful for your learning