USING TORTOISEGIT - MERGE CONFLICTS, REVERTS
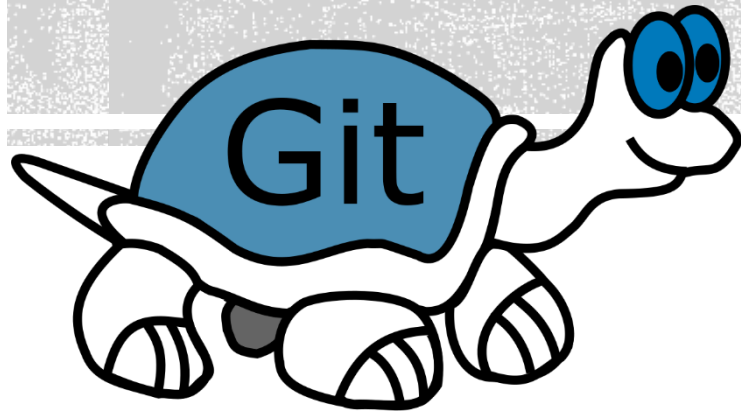
TEAM :

CHINMAYI AMBATI

MAHA LAKSHMI
KONGARI

DEEPTHI TEJASWANI
CHOKKA

WHAT IS A TORTOISE GIT?

➢TortoiseGit is a free open-source client for the *Git* version control system.

➢It help in managing files over time.

➢Files are stored in a repository.

➢The repository is much like an ordinary file server, except that it remembers every change ever made to your files and directories.

➢Every Git clone is a full-fledged repository with complete history and full revision tracking capabilities, not dependent on network access or a central server.

ADVANTAGES OF TORTOISE GIT

➢It's free! You don't need to pay to use it.

➢Allows you to recover older versions of files and examine the history of how and when your data changed, and who changed it.

➢Tortoise git can be used in which ever development tool you prefer as its not in integration with a specific IDE.

➢Branching and merging are fast and easy to do.

FEATURES OF TORTOISEGIT

➢**Shell integration :**

TortoiseGit integrates seamlessly into the Windows shell (i.e. the explorer). This means you can keep working with the tools you're already familiar with.

➢**Easy Access to Git Commands :**

All Git commands are available from the explorer context menu. TortoiseGit adds its own submenu there.

FEATURES OF TORTOISEGIT (contd.,)

➢ **Powerful commit dialog :**

Auto completion of paths and keywords of the modified files and integrated spell checker for log messages is done in an efficient manner.

➢ **Helpful Tool :**

Helps in resolving conflicts and shows the changes that you made in your file.

WHAT IS A MERGE CONFLICT

➤ **Merge Conflicts** happen when you merge branches that have competing commits, and Git needs out help to decide which changes to incorporate in the final merge.

➤ **Git can**'t know which of the changes to keep, and thus needs human intervention to **resolve** the **conflict**.

RESOLVING A MERGE CONFLICT

To resolve a merge conflict, you must manually edit the conflicted file to select the changes that you want to keep in the final merge.

There are **TWO** ways for this.

1. If your merge conflict is caused by competing line changes, such as when people make different changes to the same line of the same file on different branches in your Git repository, you can resolve it on GitHub using the conflict editor.

2. For all other types of merge conflicts, you must resolve the merge conflict in a local clone of the repository and push the change to your branch on GitHub.

REVERTING A MERGE

➢ Sometimes, we get into a situation where we merge branches but need to undo the merge. In such cases we use revert of a merge concept.

➢ To revert a merge, we need to first select the required file and select revert and choose the text which we need to keep.

References

- https://stackoverflow.com
- https://tortoisegit.org