Intro to Git and Github

Hyper Island - FED27

Per Enström – 2025-09-05

Git
Merge conflicts

- When the same line has changed in two different branches and we try to merge them together, we will encounter a *merge conflict*

- This means that Git can't automatically figure out what version to keep, and we have to help it out

- Merge conflicts pauses a merge in a conflict stage, and waits for you to resolve the files and commit the correct version

- In the conflicting file it will look something like:

```
1   << HEAD
2   This is how the line looks on our current branch
3   =======
4   This is how the line looks on the branch we want to merge into our
5   >> main
```

- We select our change by replacing this whole block of five lines with the single line we want to keep

Git

Merge conflicts

- Merge conflicts often occur when we're done with a feature and want to merge it into `main`, when something has happened on `main`

- Best practice is to:

  - merge the other way around first, merge `main` into our feature branch

  - resolve the merge conflict on our feature branch and commit that

  - merge the feature branch into `main`, which should now be conflict free