Repository

* The container for the code project

Git

* It is a tool to manage your repository

Github

* The actual concrete location to store your project, specifically for git project

How does it all fit together?

* When you start the new java project, the local folder is your repository
* Problem is we can’t track your changes, and share your code, allowing others to edit etc
* So we turn this project into a git project (by calling: git init)
* Now we need a place to store it, therefore we push the code to Github
* In result, we can manage our project

1. First thing you should every day when you start working

* Get latest changes from remote origin
  + Git fetch origin
  + Git rebase origin/master (you may get a message regarding conflict, ask James how to resolve it)

1. How to upload files after doing work on my local machine

* First track the files
  + cd project-directory
  + git add --a
  + git commit -m “Message related to the files added”
* Get latest changes from remote origin
  + git fetch origin
  + git rebase origin/master (you may get a message regarding conflict, ask James how to resolve it)
* Upload changes to remote origin
  + git push

1. How to download project when you don’t have it yet on your local machine

* Copy github URL from the project page
* Cd to working directory (example: cd /c/home/eclipse-workspace)
* Git clone pastURL

1. How to start the project when it’s not in your remote yet

* Go to github and start a new project
* Copy the link
* Open gitbash, go to the directory of your project
* Call git init (this initialize your project as git project)
* Call git remote add origin theCopiedLink (this tells your local git folder, where to push your code to on github)
* Call git add --all (this stages all your project code)
* Call git commit –m “insert msg here” (this commits your staged code and adds a message to it)
* Call git push origin master (it pushes to a remote location called origin, in a branch called master)