

Naming your files

- 150001312-H-Bee-blg101-hw1.txt
- 150001312-H-Bee-blg101-hw1-last.txt
- 150001312-H-Bee-blg101-hw1-last2.txt
- 150001312-H-Bee-blg101-hw1-last5-wednesday.txt
- 150001312-H-Bee-blg101-hw1-last11-hopefully.txt
- 150001312-H-Bee-blg101-hw1-lastlast-OMG.txt
- 150001312-H-Bee-blg101-hw1-lastlast1773-final.txt

What is a VCS?

VCS = Version Control System

- store files
- ...and their histories

useful for:

- reversing work
- sharing and incorporating work
- understanding the history of a program

Popular VCSs

- Bazaar
- Mercurial
- CVS
- Subversion
- *git*

What's git?

- A distributed version control system
 - Could be used locally
 - Could be used to share files with others
 - Could have a central storage for easier synchronization

Git Repository

- A conceptual idea in git: *repository* (aka repo)
- Any folder could be a repository
- Turn any folder into a git repository

```
cd <folder>  
git init
```

Basic git configuration

- git wants to know you

```
git config --global user.name "Honey Bee"  
git config --global user.email "bee20@itu.edu.tr"
```

Adding files

- Stage a file to track
- Getting ready to be stored

```
git add <filename>
```

- stage or index or working directory

Committing files

- Take a snapshot of all tracked files

```
git commit -m "<A descriptive message>"
```

- Sanity check: Longer messages, shorter changes

Monitoring changes

- See what happened

```
git log
```

- Check ongoing changes

```
git diff
```

Git Stash

- Stash valuable but not finished work

```
git stash
```

- List stashed work

```
git stash list
```

- Continue stashed work

```
git stash pop <order>
```

Restoring changes

- Restoring changes in the staging area

```
git restore <file>
```

- Prior to version 2.23

use “checkout” instead of “restore”

```
git checkout <file>
```

Resetting committed changes

- Scenario: missed a point in the latest commit
- Move HEAD to the previous state, fix and re-commit

```
git commit -m "<Committi messaeg>"  
git reset --soft HEAD~  
# fix your errors  
git commit -am "<Commit message>"
```

- This approach changes the commit history,
so it may broke remote repository
- Reference to the first commit is lost

Not recommended

Reverting committed changes

- Scenario: a previous commit breaks things
- Commit exactly the opposite of bad commit

```
git commit -m "<first good>"  
# break something  
git commit -m "<bad>"  
# do something  
git commit -m "<second good>"  
# do something  
git commit -m "<third good>"  
# realize "bad" commit  
git revert HEAD~2
```

- No change in the commit history, commits inverse of bad commit
- *Recommended approach*

Branches

- Generating a new branch, switching to a branch, merging branches, deleting a branch

```
git branch <new-branch>  
git switch <branch>  
git merge <branch>  
git branch -D <branch-to-delete>
```

- Prior to version 2.23

use “checkout” instead of “switch”

```
git checkout <branch>
```

Remote repositories

- Duplicate an existing repository

```
git clone <link>
```

- Or add a remote repository

```
git remote add origin <user>@<server>:<path>
```

Remote interaction

- Receive changes from a remote repository

```
git pull <remote-repo> <branch>
```

- Send contributions to a remote repository

```
git push <remote-repo> <branch>
```


Get a GitHub account

- Regular case of building a bare repository at GitHub
 - Sign up for an account on github.com
 - Choose “new repository” after clicking “+” button
 - Name your repository
 - Now, you can use this repository *publicly*

GitHub usage for the course

- Specific course organization on GitHub Classroom
- Assignments will be monitored
- Your contribution will not be *public*, but *private*
- Your contributions could not be seen by your classmates (or anybody)
- *Always be careful about what is public*

GitHub account specifications

- Generate an account specific to this course

Do not use your existing accounts or email addresses

- name the account as “ituitis20-<itu-user-name>”
- E.g. *ituitis20-surname20*

If your account name does not meet requirements, it will be removed

- Join the organization's assignment via the invitation link