26.04.2018

CS Essentials

GIT Exercise

26.04.2018

STEP 1

1) Create a GitHub account. (If you already have, you don't have to create.)

2) Start a project and create a public empty repository. Repository name could be "git-exercise".

3) You are done with GitHub for now. You can close the website.

STEP 2

1) Create a folder called "git-exercise" in your computer.

2) Make this folder git repository and set the configuration of git repository. Username should be your real name and email should be your email address.

3) Create two files, called "computerscience.txt" and "engineering.txt". Fill these files with appropriate sentences. (You can write whatever you want, but you should write something on these files.)

4) Add file1.txt to repository and commit the state with appropriate commit message.

5) Add file2.txt to repository and commit the state with appropriate commit message.

6) Create another file called "computerscience_v2.txt" and fill this file with appropriate sentences. Delete the computerscience.txt file. Add these changes to repository and commit the state with appropriate commit message.

7) At this state, you should have two files (computerscience_v2.txt and engineering.txt), three commits. Check whether you have three commits.

26.04.2018

STEP 3

1) Create a folder called “Sabanci University” in your “git-exercise” repository.

2) Create a new file called “2018.sabanci” in Sabanci University folder. In this file, you will write three courses you are taking right now. Add these changes to repository and commit the state with appropriate message.

3) Same as “Sabanci University” folder, create “Koc University” folder in your repository under the “git-exercise” folder. Create “2018.koc” file in it. Write “NOT ATTENDED” in this file. Also add these changes and commit the state with appropriate messages.

4) Open GitHub account again and your “git-exercise” repository that is previously created by you at STEP 1. Follow the instruction “…or push an existing repository from the command line” in that website.

5) After you finished the instructions, refresh your repository website and you should see 5 commits and 1 branch in your repository.

6) Add a README file in your repository through website. Commit the README file. You can examine how you can create README with good user interface on this link after the workshop.

7) In your computer, pull the changes from the remote. (You should get README.md file with this pulling operation.)

26.04.2018

STEP 4

1) Create two branches called “exams” and “projects”.

2) In branch “exams”, you should create “exams” folder under Sabanci University folder and create a file called “examdates.sabanci”. In that file, write one of your course and its exam date. Add, commit and push this branch to remote repository.
3) In branch “projects”, you should create “projects” folder under Sabanci University folder and create a file called “projects.sabanci”. In that file, write a course name you are doing a project. Add, commit and push this branch to remote repository.

4) Check GitHub and see your branches. Master branch shouldn’t contain what you made in other branches. Also, “exams” branch shouldn’t contain the folder called “projects”.

5) Checkout your master branch and merge “exams” branch to master branch and push the changes to remote. Delete “exams” branches from local repository and remote.

6) In the GitHub website, open a pull request to merge project branch into master branch. Accept the pull requests and delete the project branches from GitHub.

7) Pull the changes from the local repository to sync your repo to remote one. Delete “projects” branch from local.

8) In the end, you should have 9 commits in your master branch and projects structure should be as follow:

26.04.2018

-- git-exercise

    -- Koc University

        - 2018.koc

    -- Sabanci Univesity

        -- projects

            - projects.sabanci

        -- exams

            - examdates.sabanci

        - 2018.sabanci

    - README.md

    - computerscience_v2.txt

    - engineering.txt

THANKS