COEN-244 Tutorial #3

OUTLINE

- Git and GitHub, what are they?
- Setting up Git and GitHub
- Creating a Project: Two Methods
- Committing and pushing code
- Creating branches
- Forking a Repository
- Reviewing the previous tutorial and Completing the Code

Git and GitHub

Git: It is a version control system (a software) suitable for tracking modifications in your code. (Source Code Management (SCM))

- It is free and open-source
- It is well-distributed and high-quality

Link: https://git-scm.com/

GitHub: It is service that lets you to host your Git repository so others can see your code.

- It is a cloud-based hosting service
- It makes sharing code easier

Link: https://github.com/

Setting up Git and GitHub

Download: https://git-scm.com/downloads

- 1. Install Git once its download
- 2. Check if installed correctly: git --version
- 3. Configure Git Username and Email:
 - \$ git config --global user.name "TheBarzani"
 - \$ git config --global user.email ismaelmergasori@gmail.com
- 4. Check if configured correctlyt correctly: git config user.name
- 5. To get help: https://git-scm.com/book/en/v2/Getting-Started-Getting-Help

For GitHub: you just need to create and account or sign in.

Linking Git and GitHub

If you have a GitHub Repo:

```
$ git config credential.helper store
$ git push https://github.com/repo.git
```

Username for 'https://github.com': <USERNAME>

Password for 'https://USERNAME@github.com': <TOKEN>

ELSE TBT:

\$ git config --global credential.https://github.com.username "TheBarzani"

\$ git config --global credential.https://github.com.password [Token]

Creating a Project

Method #1 – Locally first then remote:

https://docs.github.com/en/get-started/importing-your-projects-to-github/importing-source-code-to-github/adding-locally-hosted-code-to-github

Method #2 – Cloning a GitHub repository:

- \$ git clone < https link>
- \$ cd "repo_name"
- \$ git pull origin main
- \$ git fetch --all

Making Changes, Commit, and Push

- 1. Make some changes in your repo locally.
- 2. List the changes: \$ git status
- 3. Update your repo with the remote: \$ git pull origin main
- 4. Add the new files: \$ git add . or \$ git add -A
- 5. Commit your changes: \$ git commit -m "<Message>"
- 6. Push your changes to the remote repo: \$ git push origin
- 7. Checkout the changes on GitHub.

Creating branches

If using GitHub: create the branch there first. GIT CLI:

- 1. Go to your local git repo
- 2. List branches: \$ git branch
- 3. Retrieve remote branches: \$ git fetch --all
- 4. Create a branch with the same name: \$\\$git checkout -b < name>
- 5. If branch is not on GitHub: *\$ git push -u origin <name>*
- 6. Additional linking: *\$ git branch --set-upstream-to=<name>*

Forking a Repository

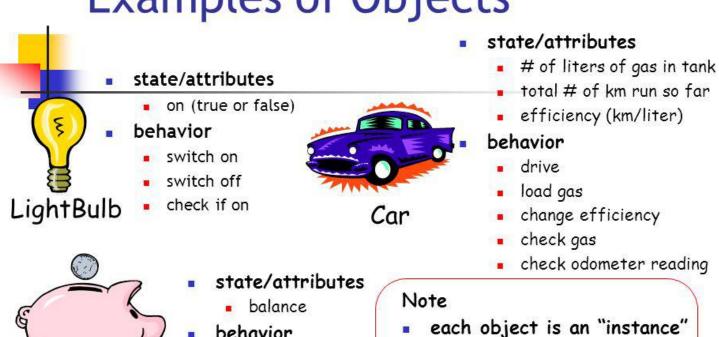
A **fork** is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. **So basically, copying code legally. | Open-Source |**

- A general method of forking is not defined properly but GitHub has a really easy and simple way to fork repositories in its server.
- Locally, you can just copy the files and make a new git environment.

LET'S TRY FORKING()

Back to

Examples of Objects



BankAccount

- behavior
 - deposit
 - withdraw
 - check balance
- each object is an "instance" of that "class" of object
- each instance has its own values for its attributes
 - e.g., different accounts can have different balances

THANK YOU