Lab 1.2: Git Setup
IN705 Databases Three

Introduction

We will use Git and GitHub for version control in this paper. In this lab we will walk through the setup of our Git repositories. The book *Pro Git* (listed in the resources section of the course directive) is a useful reference.

Git is already installed on your EC2 server. You will need to log into that server to complete the tasks below.

1 Setting Git preferences

On your EC2 server, execute the following:

```
$ git config --global user.name "Your Name"
$ git config --global user.email "your email address"
```

The email address you use should be the one associated with your GitHub account.

2 Setting up ssh keys

You will use ssh to transfer your code to and from GitHub. To make this easier, you need to set up ssh keys. This process is documented well at `https://help.github.com/articles/generating-ssh-keys`. Follow the instructions there.

Note that the section mentioning xclip is specific to GNU/Linux desktops (and probably Macs). Copy and paste the key text in the manner appropriate to your desktop system.

3 Create a repository on GitHub

You will keep all of your code for tis paper in one Git repository that we wil l name "db3". To create this repository, log onto GitHub and go to `https://github.com/new`. Fill out the form there to create a public repository. Don't tick the box to add a README, License, or .gitignore file. We will add those files later.

4 Initialise your local repository and push it to GitHub

Log onto your EC2 server and execute the following:

```
$ cd
$ mkdir db3
$ mv work/demo db3
$ cd db3
$ touch README.md
$ git init
$ git add .
$ git commit -m "initial commit"
$ git remote add origin git@github.com:your-git-username/db3.git
$ git push -u origin master
```

This assumes that you placed your hello world application in a directory named work. Briefly here's what you did: First, you organised your code into a directory named "db3". Inside that directory you created an empty README file. Then you initialised your Git repository, committed your code to it, and then added a reference from your local repository to you GitHub repository. Finally, you uploaded your code to GitHub.

Now you have a repository to store and track your work. You should make frequent commits to this repository - one or more for each working day. All of your code for this paper should be tracked in your GitHub repository.

5 Wrapping up

Email the lecturer with the URL for your GitHub repository.