CptS 223 Micro Assignment #1 - Hello World

For this micro assignment you will be required to ensure your whole Git and Java development environments are setup by building hello world and using Git.

From a coding side, there's one thing to do for this assignment. You need to finish out my "Hello World" program in the MA1-HelloWorld directory. It should print out "Hello World".

My instructions here work in UNIX style command line environments. Once you have git or git-bash (git-scm) installed on Windows, Linux, or Mac they should work just fine. Of course, if you use an IDE with Git support, you could use that as well, but all of my instructions will target the command line since it's the most universal.

The process of doing this assignment is:

1) Ensure you've completed the first assignment on blackboard telling the instructor what your EECS username is and that you've logged into the EECS GitLab server
    a) I would have created your Git repository for the class in this format:
        cs233-[EECS Username]
2) Accept the MA1-HelloWorld merge request
    a) This will make the MA1-HelloWorld branch merge into the master branch
    b) The easy way to do this is to go to the http://gitlab.eecs.wsu.edu web interface
    c) Find your CptS 233 class repository and open it up in the web browser
    d) Click on the "Merge Requests" menu item on the left side
    e) Click on the "MA1 merge to master" merge request
    f) Click on the green "Merge" button
3) Clone your git repository:
    a) Get the URL for your EECS class GitLab repository (created by the instructor) and clone the repository onto the SSH server by running this command:
    b) git clone [The URL from the server]
4) [ALTERNATIVE to #3] - If you already have your Git repo cloned, then run a "git pull" to pick up the new changes to the master branch
    a) cd into your repo on your computer
    b) run: "git pull"
5) cd into the directory created by 'git clone' and then cd into the MA1-HelloWorld directory
    a) cd cs233-[EECS Username]
    b) cd MA1-HelloWorld
6) Edit and/or build the HelloWorld.java starting file
    a) Add in the line to print out "Hello World" to STDOUT (the terminal)

7) Once that works, save your completed HelloWorld.java file
8) Running ‘git status’ should show that the HelloWorld.java file is ready to be staged for commit
9) Stage the HelloWorld.java file for committing with:
    a) git add HelloWorld.java
10) Running ‘git status’ should show that you have files staged for commit.
11) Create a commit with this single file (since that’s all you’ve changed)
    a) git commit -m “Added printing out ‘Hello World’ to MA1”
12) NOTE: #11 might give an error if you haven’t set your name and email address for your git client. It shows you what to do, but I’ll repeat it here. Run these two commands with your name and email address to set things up:
    a) git config --global user.name "FIRST_NAME LAST_NAME"
    b) git config --global user.email "MY_NAME@example.com"
    c) You can use any name and email address you like - they are not tied to any computer login or account information. They’re merely strings attached to the commits you make which allows future devs to contact people who worked on the code for a project. If you have a nickname and/or a non-WSU email address you want to be known by as a developer, this is where you’d put them in.
    d) On any given computer you’ll only have to do this once to set things up.
13) Push the newly updated repo to the server:
    a) git push
14) Running ‘git status’ should show that your repo is clean and up to date with origin
15) Check to make sure things are right by looking at the repo’s CI/CD tests
    a) It should be a green light for the MA1-HelloWorld tests in about a minute
16) Make Blackboard happy by uploading a small file to the assignment.
    a) A small file (Less than 100kb to save space) could be anything, but I know that graders like to find cute animal photos or web comics with nerdy things when they’re grading.

Grading

Your submission will be graded based on the following:

1. [10] Your working Hello World program is in your Git repo