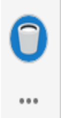
How to: Bitbucket Commands

Making a New Repository:

1. Log into your account on bitbucket.org
2. In the menu bar at the top of the page there is a drop-down button called ‘Create’ click it.
3. Then from the list of options click ‘Create Repository’ (the first option).
4. The page that appears will ask you to give the repository a name; name the repository according to the naming convention noted in the lab assignment.
5. (OPTIONAL) in the language section select Java from the dropdown box if this repository will mainly contain Java programs
6. You will be taken to the “Repository Setup” page.
7. Look to the left side of your web browser, you will see a menu bar with a blue icon with a bucket in it (which looks exactly like the the image directly left of this point). Directly below this icon you will see three grey dots. Click the three grey dots and a menu of actions will appear. Select the “Clone” option.



8. A new box will appear! Select HTTPS from the drop-down menu and then copy the text that is located in the text box (ex: git clone https://….)
9. Now in your terminal navigate to the folder in which you want your new repository to exist. Keeping in mind that the clone will create the folder for you!

   <u>IMPORTANT: DO NOT PUT A REPOSITORY INSIDE A REPOSITORY</u>

   In the example below, if cs111 is a repo, do not put (repo_name) inside of it

(Ex: >cd home/(your username)/cs111/labs/

```
├── home
│   ├── (your username)
│   │   ├── cs111
│   │   │   ├── labs //run git clone at this level
│   │   │   │   ├── (repo_name) //if you want the folder here
│   │   │   │   ├── Lab[lab_number] //run git clone at this level
│   │   │   │   |   ├── (repo_name) //if you want the folder here
```

10. Paste the copied text into the terminal window.
11. Your new repository will clone into your file system. This is a very exciting moment, please take a minute to celebrate and then begin your work.

Essential Commands:

To commit your recent changes it is necessary to complete the following three steps:

1. *git add (file1).java (file2).java (file3).txt*

You have to navigate into the directory containing the files you would like to add. You can also use *git add* * command to add all of the files inside the directory you are currently in.

2. *git status* (make sure the files you want are added in correctly)
3. *git commit -m "(your commit message)"*
4. *git push -u origin master* (after the first push, you can just type ***git push***)

If you are adding multiple new files from different subdirectories you may use:

1. git add -A (or *git add --all*) (from the top repository directory)
2. *git commit -m "describe committed materials"*
3. *git push origin master* (after the first push, you can just type ***git push***)

To get recent changes:

1. *git pull* (or *git pull origin master*)
2. If you have a merge conflict type:
    a. *git stash*
    b. *git pull origin master*
    c. IMPORTANT: check the code that was pulled (it will be in the git pull log) in a text editor and make sure that it is not a better (more recent version) than your own code
    d. If your code is more recent run: git stash apply
        i. If merge conflicts still appear, they need to be fixed manually by modifying the file that is causing the conflict
    e. If it is not, you can run git stash drop

When you don't know what you're doing or something goes wrong or just for fun do this:

git status

Non-Essential Commands:

git rm --cached fileName.java	this will remove a file from your git repository
git stash	saves changes in the local
git merge	merges branches of a directory (use this if conflicts exist)
git status	shows current changes that you have made using git ex: any files that have been added
git reset	undo any git actions that have been made including commits or adds but not pushes