On Git

A Crash Course By
Michael Nash

Who Am I?

Additional Cats

Course Notes

- This presentation and all links in it will be available online
- Course Notes/Links: https://github.com/utumno86/BlogPosts/blob/master/GitPresentation.md
- Alternately: https://github.com/utumno86/BlogPosts/blob/master/GitPresentation.key
- Alternately Alternately: https://github.com/utumno86/BlogPosts/blob/master/GitPresentation.pdf

What Is A Version Control System?

- From Wikipedia -

- "... management of changes to documents, computer programs, large web sites, and other collections of information."

-

What Is git?

Linus Torvalds invented git in 2005

Installing git

- OsX:
  - 1) Install Homebrew: (http://brew.sh)
    - 2) *brew install git*
- Windows:
  - 1) Install git for windows: (https://gitforwindows.org/)
- git Website: (https://git-scm.com/)

Configuration Overview

- Personal Info
  - *git config --global user.name "John Doe"*
  - *git config --global user.email johndoe@example.com*
- Editor
  - *git config --global core.editor emacs*
- Aliases (optional)
  - *git config --global alias.co checkout*
  - *git config --global alias.br branch*
  - *it config --global alias.ci commit*
  - *git config --global alias.st status*

gitignore global

- *touch ~/.gitignore_global*
- *git config --global core.excludesfile '~/.gitignore_global'*
- Sugested Gitignore contents: (https://github.com/github/gitignore)

Basic git commands

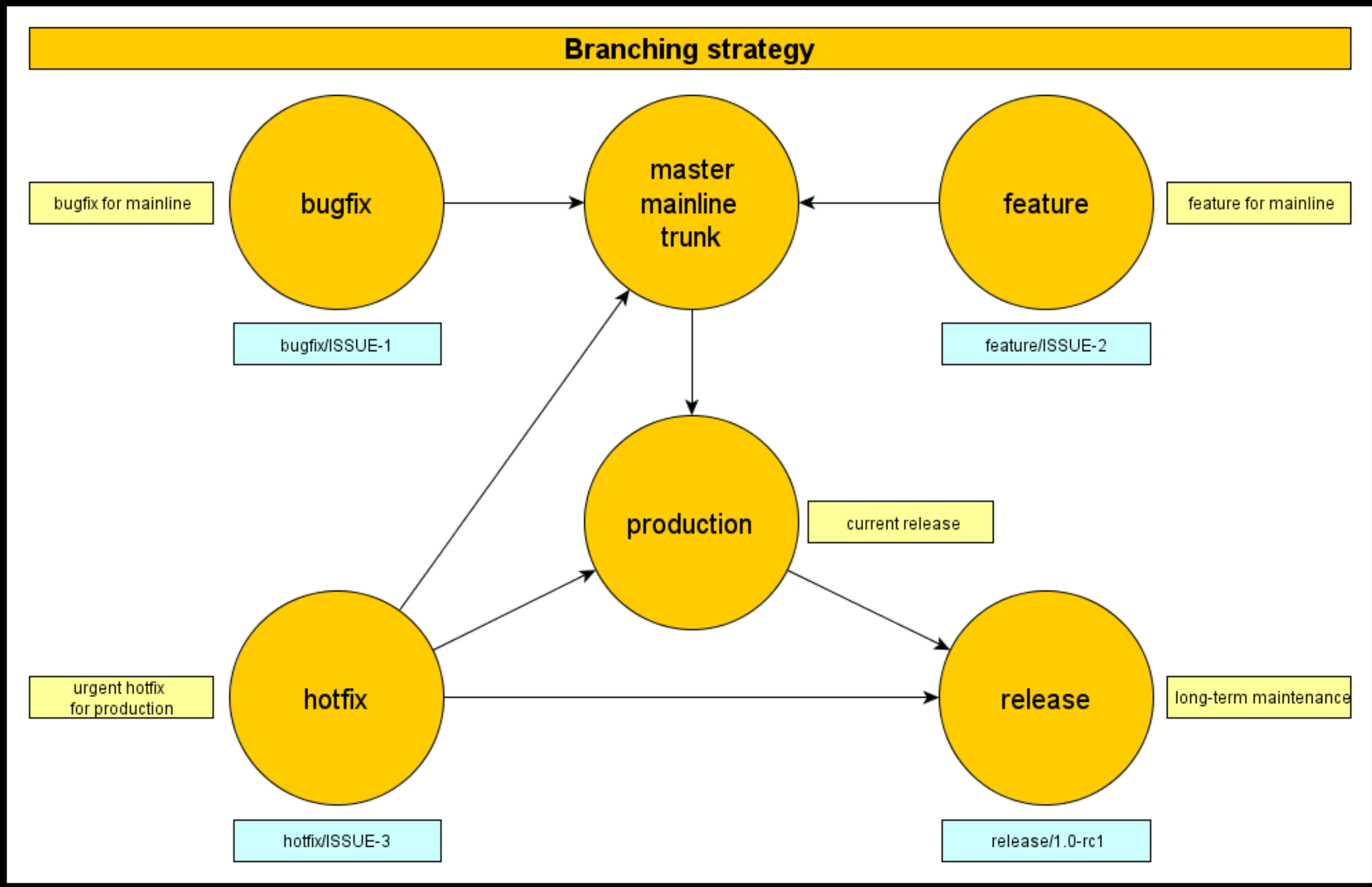
- *git init*
- *touch hello.txt*
- *git add .*
- *git commit -m “message”*

How do I get out of vim?

- *:q*

- How to exit the vim editor: (https://stackoverflow.com/questions/11828270/how-to-exit-the-vim-editor)

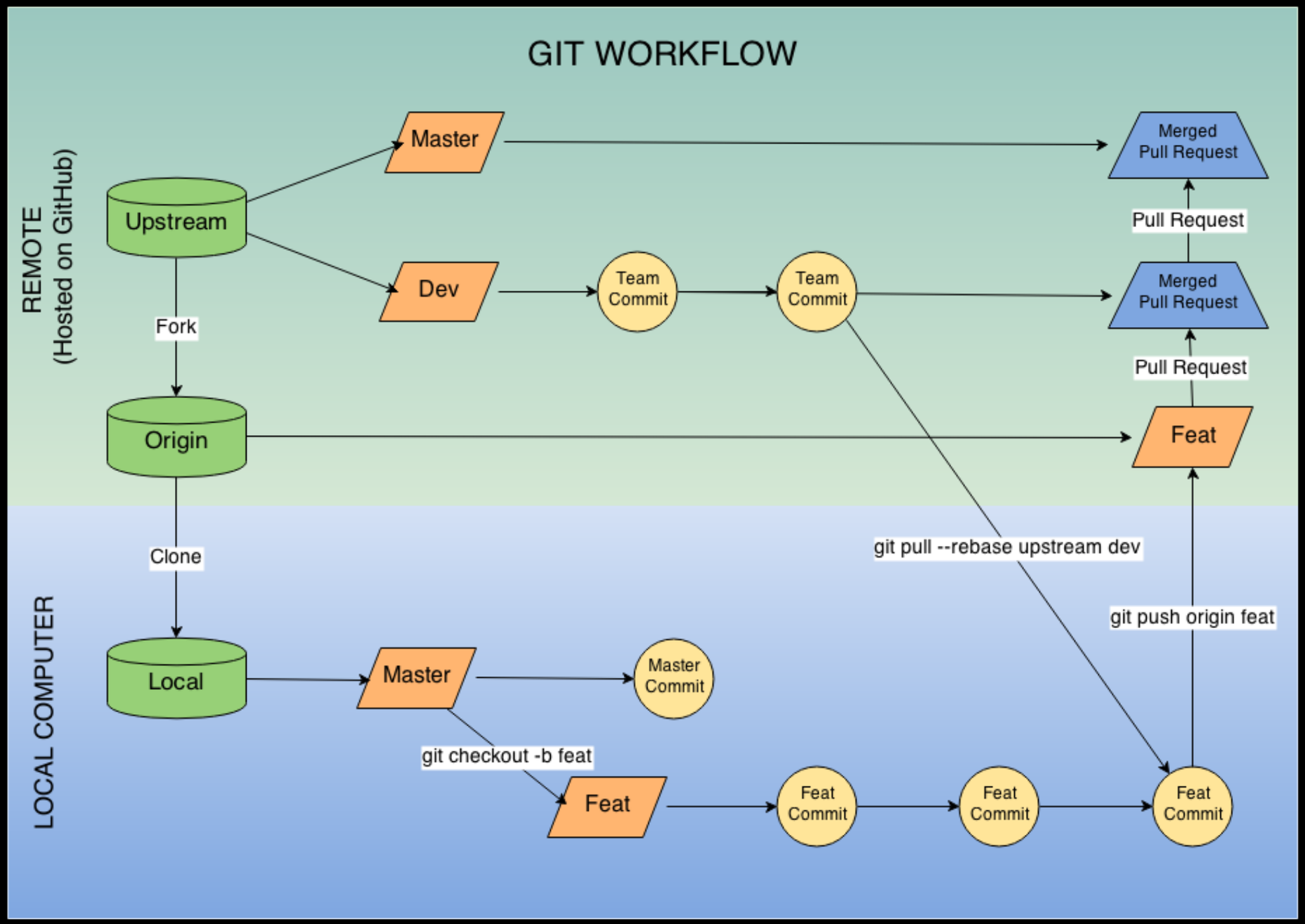
git branching



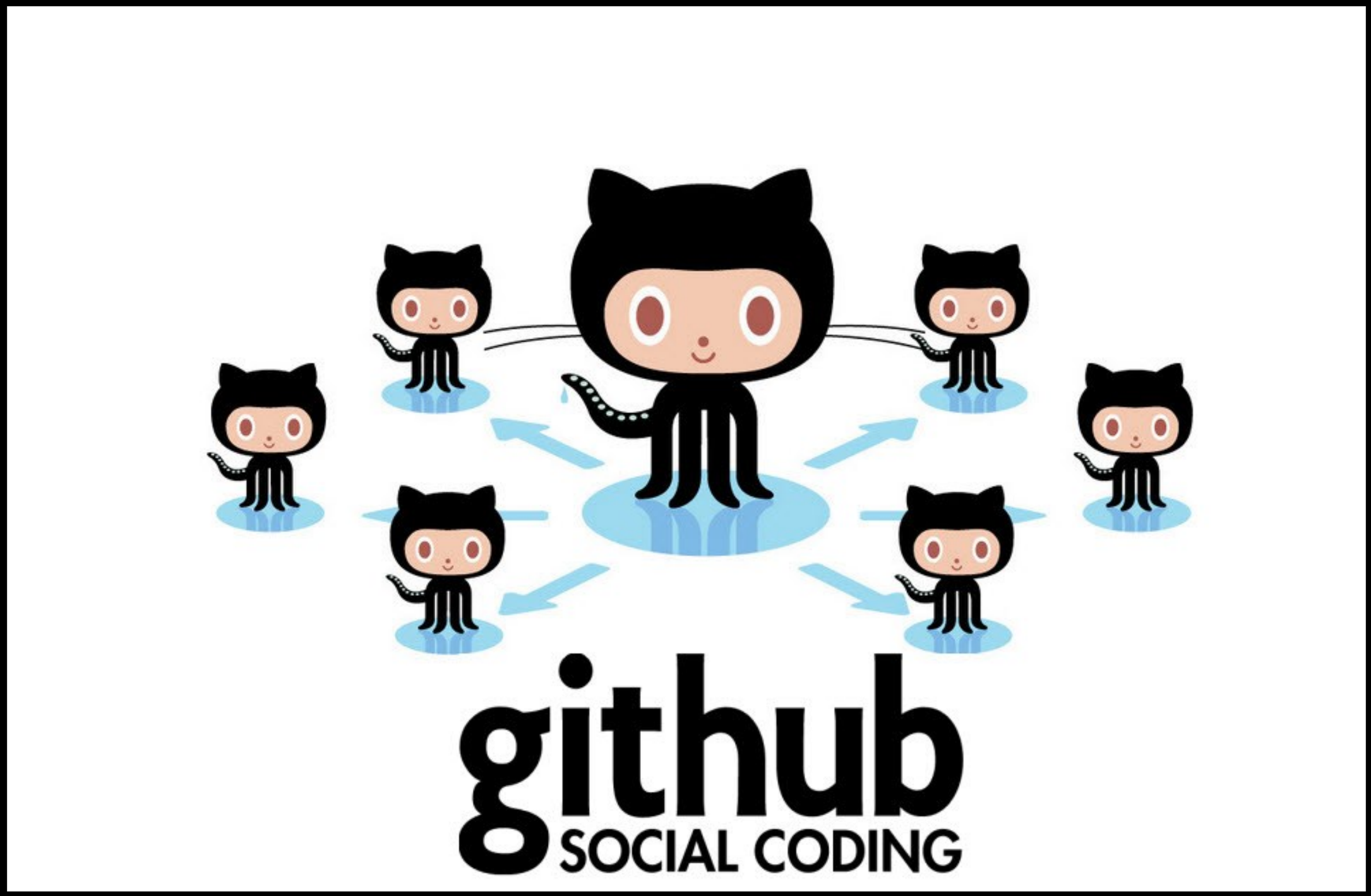
git branching commands

- *git checkout -b testing*
- *git branch*
- *git checkout master*
- *git merge testing*

Remote Repositories Workflow



Github



SSH

- from Wikipedia: "Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network. ... The best known example application is for remote login to computer systems by users."

Steps to Set Up an SSH Connection With Github

- 1) Check for existing SSH Keys: (https://help.github.com/articles/checking-for-existing-ssh-keys/)

- 2) Generating an New SSH Key and/or adding it to the SSH agent: (https://help.github.com/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent/)

- 3) Add the SSH Key To Your Github: (https://help.github.com/articles/adding-a-new-ssh-key-to-your-github-account/)

git clone

- *git clone git@github.com:utumno86/git_tutorial.git GitTutorial*

git remote

- *git remote*
- *git remote add origin git@github.com:utumno86/git_tutorial.git*

Interacting With Remote Repositories

- *git push*
- *git pull*
- Additional Command: *git fetch*

Resolving Merge Conflicts

Merge Revisions for C:\t\MergeDemoDevelopment\app\src\main\java\ca\cmpt276\mergedemo\NumberFun.java

Highlight words

6 changes. 2 conflicts

Local Changes (Read-only) LF | Result CRLF | Changes from Server (revision 50314cc8638a07fa4a88653d0752ec656f9... LF

```
Local Changes (Read-only)                    Result                                   Changes from Server
public int getMin() {                        * NumberFun class manages some data an generates   * NumberFun class manages some data an genera
    int min = data[0];                       * My Changes and teammate's changes applied!       * Teammate's changes applied!
    for (int value : data) {                 */                                                */
        if (value < min) {                                                                     
            min = value;                     public class NumberFun {                          public class NumberFun {
        }                                        private int[] data;                               private int[] data;
    }                                            public NumberFun(int[] data) {                    public NumberFun(int[] data) {
    return min;                                      this.data = data;                                 this.data = data;
}                                                }                                             }
                                                                                               
public int getAverage() {                        public int getAverage() {                         public int getAverage() {
    if (data.length == 0) {                          int sum = 0;                                  int sum = 0;
        return 0;                                    int i = 0;                                    for (int i = 0; i < data.length; i++)
    }                                                while (i < data.length) {                         sum += data[i];
                                                         sum += data[i];                           }
    int sum = 0;                                         i++;                                      return sum / data.length;
    int i = 0;                                       }                                         }
    while (i < data.length) {                        return sum / data.length;                 
        sum += data[i];                          }                                             public int getMax() {
        i++;                                                                                       int max = data[0];
    }                                            public void printData() {                         for (int value : data) {
    return sum / data.length;                        // Print out the data                            if (value > max) {
}                                                    int i = 0;                                        max = value;
                                                     while (i < data.length) {                     }
public void printData() {                                int value = data[i];                      }
    // Print out the data                                System.out.print(value + ", ");           return max;
    for (int value : data) {                             i++;                                  }
        System.out.print(value + ", ");              }                                         
    }                                                System.out.println();                     public void printData() {
    System.out.println();                                                                          // Print out the data
                                                     // Print out the stats:                       for (int i = 0; i < data.length; i++)
    // Print out the stats:                          int avg = getAverage();                           int value = data[i];
    int avg = getAverage();                          System.out.println("Stats:   avg = " + avg         System.out.print(value + ", ");
    int min = getMin();                          }                                                 }
```

Accept Left | Accept Right | Apply | Abort

VS Code Git Packages

- GitLens: https://gitlens.amod.io/

Additional Resources

- Git Cheatsheet: (https://github.com/k88hudson/git-flight-rules)
- Git GUI Options
    - SourceTree: (https://www.sourcetreeapp.com/) -- An open source Atlassian product
    - Github Desktop: (https://desktop.github.com/) -- Open source election-based crossed platform sponsored by Github
    - GitKraken (https://www.gitkraken.com/) -- Beautiful, but paid!

Contact Info

- Github: https://github.com/utumno86
- Twitter: https://twitter.com/utumno86
- LinkedIn: https://www.linkedin.com/in/michaelearlnash