#### Practice Quiz: Introduction to GitHub

**TOTAL POINTS 4**

1.

Question 1

When we want to update our local repository to reflect changes made in the remote repository, which command would we use?

**1 / 1 point**



git clone <URL>



git push



git pull



git commit -a -m

**Correct**

Right on! git pull updates the local repository by applying changes made in the remote repository.

2.

Question 2

git config --global credential.helper cache allows us to configure the credential helper, which is used for ...what?

**1 / 1 point**



Troubleshooting the login process



Dynamically suggesting commit messages



Allowing configuration of automatic repository pulling



Allowing automated login to GitHub

**Correct**

Nice work! By configuring the credential helper, we can avoid having to type in our username and password repeatedly.

3.

Question 3

Name two ways to avoid having to enter our password when retrieving and when pushing changes to the repo. (Check all that apply)

**1 / 1 point**



Implement a post-receive hook



Use a credential helper

**Correct**

Awesome! The credential helper caches our credentials for a time window, so that we don't need to enter our password with every interaction.



Create an SSH key-pair

**Correct**

Great job! We can create an SSH key-pair and store the public key in our profile, so that GitHub recognizes our computer.



Use the git commit -a -m command.

4.

Question 4

Before we have a local copy of a commit, we should download one using which command?

**1 / 1 point**



git commit -a -m



git push



git pull



git clone <URL>

**Correct**

Woohoo!. If the repository already exists locally, this may raise an error.

## Git Remotes Cheat-Sheet

| **Command** | **Explanation & Links** |
| --- | --- |
| git remote | [Lists remote repos](https://git-scm.com/docs/git-remote) |
| git remote -v | [List remote repos verbosely](https://git-scm.com/docs/git-remote#Documentation/git-remote.txt--v) |
| git remote show <name> | [Describes a single remote repo](https://git-scm.com/docs/git-remote#Documentation/git-remote.txt-emshowem) |
| git remote update | [Fetches the most up-to-date objects](https://git-scm.com/docs/git-remote#Documentation/git-remote.txt-emupdateem) |
| git fetch | [Downloads specific objects](https://git-scm.com/docs/git-fetch) |
| git branch -r | [Lists remote branches](https://git-scm.com/docs/git-branch#Documentation/git-branch.txt--r); can be combined with other branch arguments to manage remote branches |

You can also see more in the video [Cryptography in Action](https://www.coursera.org/learn/it-security/item/P1I8z) from the course [IT Security: Defense against the digital dark arts](https://www.coursera.org/learn/it-security/home/welcome).

#### Practice Quiz: Using a Remote Repository

**TOTAL POINTS 5**

1.

Question 1

In order to get the contents of a remote branch without automatically merging, which of these commands should we use?

**1 / 1 point**



git pull



git remote update



git checkout



git log -p -1

**Correct**

You got it! git remote update will fetch the contents of all remote branches and allow us to merge the contents ourselves.

2.

Question 2

If we need to find more information about a remote branch, which command will help us?

**1 / 1 point**



git fetch



git checkout



git remote update



git remote show origin

**Correct**

Right on! If you want to see more information about a particular remote branch, you can use the git remote show command. Don't forget the commit ID!

3.

Question 3

What command will download remote branches from remote repositories without merging the content with your current workspace automatically?

**1 / 1 point**



git checkout



git pull



git fetch



git remote update

**Correct**

Nice work! git fetch will download remote updates, such as objects and refs, from the remote branch.

4.

Question 4

What type of merge creates a new merge commit?

**1 / 1 point**



Fast-forward merge



Implicit merge



Explicit merge



Squash on merge

**Correct**

Woohoo! An explicit merge creates a new merge commit. This alters the commit history and explicitly shows where a merge was executed.

5.

Question 5

What method of getting remote contents will automatically merge the remote branch with the current local branch?

**1 / 1 point**



git fetch



git checkout



git remote update



git pull

**Correct**

Great job! git pull automatically merges the remote branch with the current branc

Basic Interaction with GitHub Cheat-Sheet

There are various remote repository hosting sites:

* [GitHub](http://github.com/)
* [BitBucket](https://bitbucket.org/product)
* [Gitlab](https://gitlab.com/).

Follow the workflow at <https://github.com/join> to set up a free account, username, and password. After that, [these steps](https://help.github.com/articles/create-a-repo/) will help you create a brand new repository on GitHub.

Some useful commands for getting started:

| **Command** | **Explanation & Link** |
| --- | --- |
| git clone URL | [Git clone is used to clone a remote repository into a local workspace](https://git-scm.com/docs/git-clone) |
| git push | [Git push is used to push commits from your local repo to a remote repo](https://git-scm.com/docs/git-push) |
| git pull | [Git pull is used to fetch the newest updates from a remote repository](https://git-scm.com/docs/git-pull) |

This can be useful for keeping your local workspace up to date.

* <https://help.github.com/en/articles/caching-your-github-password-in-git>
* <https://help.github.com/en/articles/generating-an-ssh-key>

Conflict Resolution Cheat Sheet

Merge conflicts are not uncommon when working in a team of developers, or on Open Source Software. Fortunately, GitHub has some good documentation on how to handle them when they happen:

* <https://help.github.com/en/github/collaborating-with-issues-and-pull-requests/about-merge-conflicts>
* <https://help.github.com/en/github/collaborating-with-issues-and-pull-requests/resolving-a-merge-conflict-using-the-command-line>

You can also use [git rebase branchname](https://git-scm.com/book/en/v2/Git-Branching-Rebasing) to change the base of the current branch to be branchname

The git rebase command is a lot more powerful.  Check out [this link](https://git-scm.com/book/en/v2/Git-Tools-Rewriting-History) for more information.

#### Practice Quiz: Solving Conflicts

**TOTAL POINTS 5**

1.

Question 1

If you’re making changes to a local branch while another user has also made changes to the remote branch, which command will trigger a merge?

**1 / 1 point**



git push



git pull



git rebase



git fetch

**Correct**

Nice job! The git pull command runs git fetch with the given parameters, then calls git merge to merge the retrieved branch heads into the current branch.

2.

Question 2

Which of the following is a reason to use rebase instead of merging?

**1 / 1 point**



When you want to keep a linear commit history



When you want a set of commits to be clearly grouped together in history



When you are on a public branch



When pushing commits to a remote branch

**Correct**

Way to go! git rebase is useful for maintaining a clean, linear commit history.

3.

Question 3

Where should we keep the latest stable version of the project?

**1 / 1 point**



The master branch



A separate branch from the master branch



The debug branch



A remote branch

**Correct**

Great work! It's common practice to keep the latest version in the master branch and the latest stable version in a separate branch.

4.

Question 4

Which of the following statements represent best practices for collaboration? (check all that apply)

**1 / 1 point**



When working on a big change, it makes sense to have a separate feature branch.

**Correct**

Right on! This lets you work on new changes, while still enabling you to fix bugs in the other branch.



You should always rebase changes that have been pushed to remote repos.



Always synchronize your branches before starting any work on your own.

**Correct**

Awesome! That way, whenever you start changing code, you know that you're starting from the most recent version, and you minimize the chances of conflicts or the need for rebasing.



Avoid having very large changes that modify a lot of different things.

**Correct**

Woohoo! Instead, try to make changes as small as possible, as long as they’re self-contained.

5.

Question 5

What command would we use to change the base of the current branch?

**1 / 1 point**



git checkout <branchname>



git pull



git rebase <branchname>



git fetch

**Correct**

Right on! You can also use git rebase <branchname> to change the base of the current branch to be <branchname>.