

# TECHNICAL DOCUMENTATION GE01

## PYTHON AND GIT/GITHUB

### Table of Contents

Resources .....	1
How to .....	2
Issue Resolution .....	3

# RESOURCES

## MERGING

<https://git-scm.com/docs/git-merge>

- Git Bash website details the git “merge” command with brief overview of the syntax, capabilities, and all the available options.
- Details types of merging, possible conflicts that could arise from non-updated tree branches and solutions, and available configurations.
- Includes examples and specific commands.

Team used this resource to understand the merge command and assess what the assignment was asking by understanding the capabilities and conflicts arising from separate commits.

---

## MERGING CONFLICTS

<https://www.atlassian.com/git/tutorials/using-branches/merge-conflicts>

- Bit Bucketlist website about resolving merging conflicts.
- Overviews causes of merge conflicts, how to identify them, and commands for fixing them.

We used this to help resolve our merge conflict from an not updated file versions.

---

## GIT COMMANDS

<https://education.github.com/git-cheat-sheet-education.pdf>

- Git hub produced git command cheat sheet.
- Includes git command syntax and one sentence overview for accomplishing basic tasks including set up, configurations, pushing, merging, and analyzing.
- Does not include in depth analysis or options.

Team used this resource for command and syntax help.

## PYTHON VIRTUAL ENVIRONMENT

<https://packaging.python.org/en/latest/guides/installing-using-pip-and-virtual-environments/#creating-a-virtual-environment>

- Python website about using a virtual environment.
- Its uses of virtual environment and brief instructions of setting up
- Details package installation and uses from multiple sources

Team used this for help creating/understanding python virtual environments.

---

## HOW TO

### MERGE

`git checkout <branch merging to>; git merge <branch merging from>; git push`

*Note: Merging is for updating versions of the same file located in separate branches, not separate files. Make sure versions have the same file name in each branch.*

---

### TAG

`git tag -a <tag name> -m '<tag description>'; git tag; git push origin <tag name>`

---

### PUSH A DELETED FILE

`git rm <file name>; git commit -m "delete"; git push`

## ISSUE RESOLUTION

### MERGING CONFLICT

Issue: Cannot push updates to github repository.

Cause: Branch version being edited is behind the branch version currently uploaded to github repository.

Solution: git pull; edit the file (vi <filename>) and remove indications of file edit placements such as “<<<<<< HEAD”; git add <file name>; git commit -m <file name>; git push

Error Message:

```
! [rejected] Iteration01 -> Iteration01 (non-fast-forward)
error: failed to push some refs to 'https://github.com/abiConway/cs3300kaja.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. If you want to integrate the remote changes,
hint: use 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

File after being pulled:

```
print("KAJA")
print("Joshua Fuentes")
print("Andrew Douangprachanh")
print("Kim Call")
print("Abigail Conway")
<<<<<< HEAD
print("This is merge attempt 1")
=====

print("The mission to MERGE!")
>>>>>> 09a1919470dba17fdef43ac3386f0ce69bc403c3
~
~
~
```