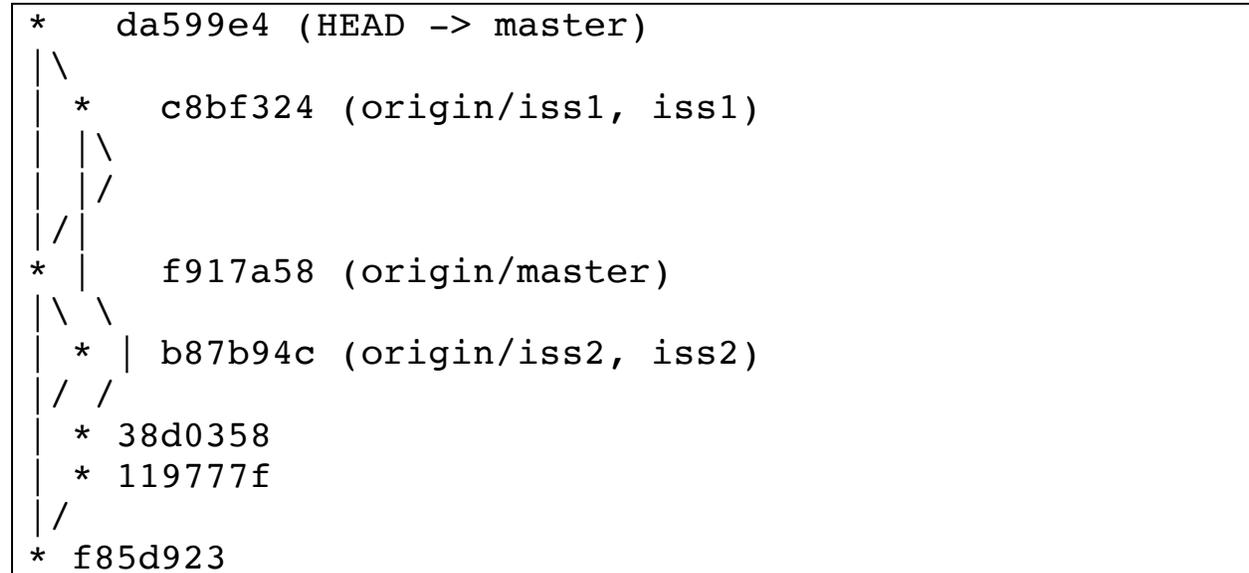


# Git Commands for Remote Repos

Consider this Git log graph (log messages omitted). Assume that `origin` refers to a remote GitHub repo, and assume that each local branch is tracking with respect to its corresponding remote branch.



What does `(origin/master)` denote?

**The latest commit on the remote GitHub master branch.**

What is the difference between `iss2` and `origin/iss2`?

**`iss2` is a branch in the local repo, whereas `origin/iss2` is a branch in the remote GitHub repo.**

Consider this Git log graph (log messages omitted). Assume that `origin` refers to a remote GitHub repo, and assume that each local branch is tracking with respect to its corresponding remote branch.

```
*    da599e4 (HEAD -> master)
| \
|  *    c8bf324 (origin/iss1, iss1)
|  | \
|  | /
|  / |
| *    f917a58 (origin/master)
| \ \
|  * | b87b94c (origin/iss2, iss2)
|  / /
|   * 38d0358
|   * 119777f
|   /
| * f85d923
```

List each branch for which the version history is exactly the same in the local repository as it is in the remote repository.

**iss1, iss2 are exactly the same in the local repo as they are in the remote repo (origin)**

---

---

List each branch for which the version history is different in the local repository than it is in the remote repository.

**master has four more commits (da599e4, c8bf324, 38d0358, and 119777f) in the local repo than it does in the remote repo (origin).**

---

Consider this Git log graph (log messages omitted). Assume that `origin` refers to a remote GitHub repo, and assume that each local branch is tracking with respect to its corresponding remote branch.

```
*    da599e4 (HEAD -> master)
| \
|  *    c8bf324 (origin/iss1, iss1)
|  | \
|  | /
|  / |
| *    f917a58 (origin/master)
| \ \
|  * | b87b94c (origin/iss2, iss2)
|  / /
|   * 38d0358
|   * 119777f
|   /
| * f85d923
```

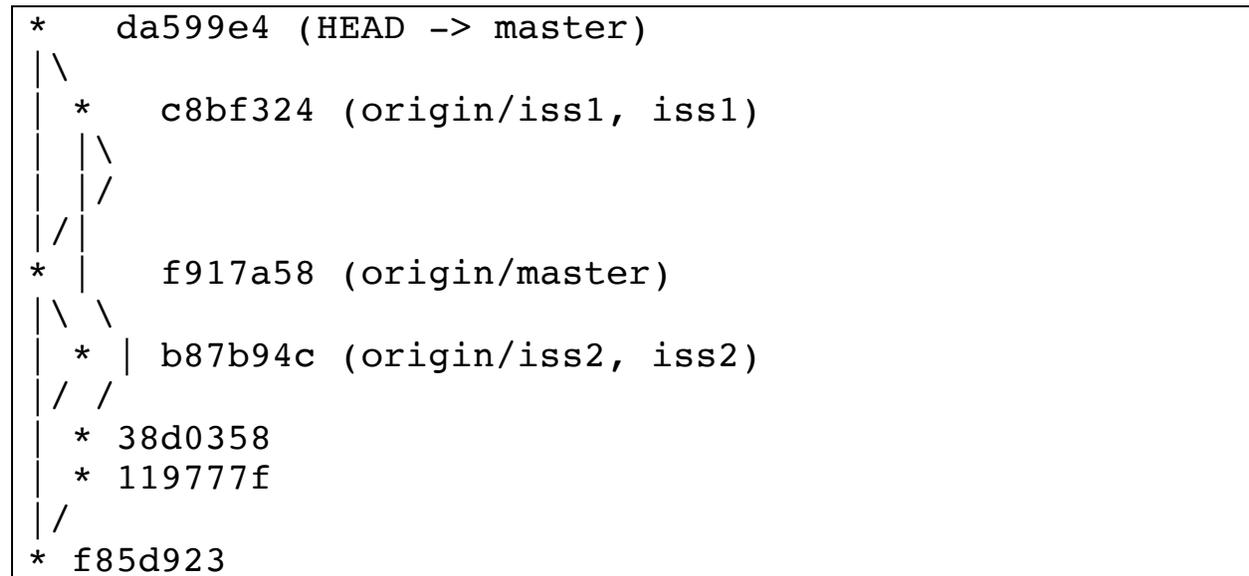
Which commits are only in the local repo and not in the remote repo?

**da599e4 is only in the local repo. You can tell because it isn't contained in the version history of any remote branch.**

Which commits are only in the remote repo and not in the local repo?

**There are no commits that are in only the remote repo and not in the local repo.**

Consider this Git log graph (log messages omitted). Assume that `origin` refers to a remote GitHub repo, and assume that each local branch is tracking with respect to its corresponding remote branch.



Which of the following would `git status` report?

- a) Your branch is up-to-date with 'origin/master'
- b) Your branch is ahead of 'origin/master' ✓
- c) Your branch is behind 'origin/master'

What would happen if the developer ran `git checkout origin/iss1`?

**Checking out a remote branch is different than checking out a local branch. In this case, it would be equivalent to `git checkout c8bf324` and would result in a detached HEAD.**

Which Git command will download all the latest remote branch log data?

**git fetch**

Each of the following problems presents a Git log graph (log messages omitted) of a local repo and a scenario. Give the updated graph that would result from each scenario.

- If you need to add a commit, use the hash `c1c1c`.
- If a command would be rejected by GitHub (e.g., because the remote contains work that you do not have locally), write “REJECTED” on the top line.
- Assume that all remote bookmarks depicted are up to date.
- Assume that all issue branches are tracking with their corresponding branches on the remote.

Scenario: Developer makes changes to the code, stages the changes, and commits.

```
* d3994b3 (HEAD -> iss13, origin/iss13)
* | 0152ac4 (origin/master, master)
|/
* 77a9025
```

---

---

```
* c1c1c1c (HEAD -> iss13)
* d3994b3 (origin/iss13)
* | 0152ac4 (origin/master, master)
|/
* 77a9025
```

---

Scenario: Developer runs `git push`.

```
* d3994b3 (HEAD -> iss9)
* | 0152ac4 (origin/master, master)
|/
* 77a9025 (origin/iss9)
```

---

```
* d3994b3 (HEAD -> iss9, origin/iss9)
* | 0152ac4 (origin/master, master)
|/
* 77a9025
```

Scenario: Developer runs `git push`.

```
* 79bb885 (HEAD -> iss20)
* | 33a99ac (origin/iss20, master)
|/
* d23531d (origin/master)
```

---

**REJECT.** This push would be the equivalent of saying "merge branch `iss20` into `origin/iss20`"; however, pushes are allowed only when such a merge would be a fast-forward merge.

---

---

Scenario: Developer runs `git pull`. Assume that auto-merge, if used, would complete successfully with no merge conflicts.

```
* d3994b3 (origin/iss9)
* | 0152ac4 (origin/master, master)
|/
* 77a9025 (HEAD -> origin/iss9)
```

---

```
* d3994b3 (origin/iss9, HEAD -> origin/iss9)
* | 0152ac4 (origin/master, master)
|/
* 77a9025
```

Scenario: Developer runs `git pull`. Assume that auto-merge, if used, would complete successfully with no merge conflicts.

```
* 79bb885 (origin/iss20)
* | 33a99ac (HEAD -> iss20, master)
|/
* d23531d (origin/master)
```

---

```
* c1c1c1c (HEAD -> iss20)
|\
| * 79bb885 (origin/iss20)
* | 33a99ac (master)
|/
* d23531d (origin/master)
```

Scenario: Developer runs `git pull origin master`. Assume that auto-merge, if used, would complete successfully with no merge conflicts.

```
* a3b25e7 (origin/master)
* e41c5b6 (master)
* 9d63832 (HEAD -> iss8)
* 40f26d8 (origin/iss8)
```

---

```
* a3b25e7 (origin/master, HEAD -> iss8)
* e41c5b6 (master)
* 9d63832
* 40f26d8 (origin/iss8)
```

Scenario: Developer runs `git pull origin master`. Assume that auto-merge, if used, would complete successfully with no merge conflicts.

```
* d3994b3 (HEAD -> iss19, origin/iss19)
* | 0152ac4 (origin/master, master)
|/
* 77a9025
```

---

```
* c1c1c1c (HEAD -> iss19)
|\
| * d3994b3 (origin/iss19)
* | 0152ac4 (origin/master, master)
|/
* 77a9025
```