

Blame Tracker Usage Guide

- **Python 3.11**
- **Git** (installed and accessible via the command line)

If you don't have them installed, follow the installation instructions for your operating system (Windows, macOS, or Linux).

Instructions

1. Navigate to your project repository

```
cd /path/to/your/repository
```

2. Download the Blame Tracker script

For macOS, Linux, or WSL:

Use one of the following commands depending on your version of `curl`:

- If you have `curl` version **7.74.0** or newer:

```
curl -L "https://raw.githubusercontent.com/adam-mcdaniel/blame-tracker/main/blame-tracker.py" -o blame-tracker.py
```

- If you have an older version of `curl`:

```
curl --location "https://raw.githubusercontent.com/adam-mcdaniel/blame-tracker/main/blame-tracker.py" --output blame-tracker.py
```

For Windows users:

Option 1: Download via PowerShell

Open PowerShell and run the following command to download the script:

```
Invoke-WebRequest -Uri "https://raw.githubusercontent.com/adam-mcdaniel/blame-tracker/main/blame-tracker.py" -OutFile "blame-tracker.py"
```

Option 2: Download via Browser

Alternatively, Windows users can download the script directly from the following link using a browser: <https://raw.githubusercontent.com/adam-mcdaniel/blame-tracker/main/blame-tracker.py>

Save the file as `blame-tracker.py` in your project repository.

3. Generate the commit file for each author

To create a file named `your_netid.commits.txt` containing all commits from the past two weeks, run the following command:

```
python blame-tracker.py -by "Author Name" -ex "helloplus" "sprint1" --since "9/28/2024" --until "10/12/2024" -o your_netid.commits.txt
```

Important: Check if you need to use `python` or `python3`

- On **Linux** and **macOS**, you may need to use `python3` instead of `python` if Python 2.x is set as the default. To check, run:

```
python --version
```

If it shows Python 2.x, use `python3` to run the script:

```
python3 blame-tracker.py -by "Author Name" -ex "helloplus" "sprint1" --since "9/28/2024" --until "10/13/2024" -o your_netid.commits.txt
```

- On **Windows**, the command `python` typically points to Python 3.x, so you can use `python`. If unsure, check the version with:

```
python --version
```

Blame-tracker command explanation:

- Replace `/path/to/your/repository` with the actual path to your project repository.
- Replace `"Author Name"` with your own name (or the name of any other author in the repository). You can run `git shortlog -sn` to list all the author names in your repository.
- The flags `--since "date"` and `--until "date"` ensure that only commits from the last two weeks are included. Date is in `mm/dd/yyyy` format.
- The `-ex` flag excludes files/folders that you don't want included in the `commits.txt`. Some common ones to exclude would be `package-lock.json`, `package.json`, `sprint` folders, `helloplus`, image/logo files, etc.
- Lastly, the `-o` flag just outputs the code changes to `your_netid.commits.txt`, change the `your_netid` to your net id.

Note on `--since` and `--until` flags:

- The `--since` flag is **inclusive**, meaning the script will include commits made **on the starting date**.
- The `--until` flag is **exclusive**, meaning the script will **not include commits made on the ending date**. You need to specify **the day after the demo** for `--until` if you want to capture all commits up to the demo date.

For example, if the sprint starts on **10/1/2024** and the demo is on **10/15/2024**, use:

```
--since "10/1/2024" --until "10/16/2024"
```

Notes

- Your project repository has multiple authors, so each author should generate their own `netid.commits.txt` file.
- Make sure to exclude any unnecessary files (like large binary files) by using the `ex` flag.
- Adam (the author of blame-tracker.py) has very good README on how to use this program. If you want to learn more about it, check it out here - <https://github.com/utk-cs340-fall24/blame-tracker>

For any issues or questions, feel free to reach out!
