git tips and tricks

10 tips you may not know

about me

Sebastian Feldmann

Teamlead CHECK24.de

PHP since <2000

@movetodevnull

Autocompletion

Download the completion script

```
curl -O https://raw.github.com/git/git/master/contrib/completion/git-completion.bash
mv git-completion.bash .git-completion.bash
```

Then in your `.bash_profile` add the following

```
if [ -f ~/.git-completion.bash ]; then
    . ~/.git-completion.bash
fi
```

git log

`--autor="Sebastian"`	Only show commits made by a certain author
`--name-only`	Only show names of files that changed
`--oneline`	Show commit data compressed to one line
`--graph`	Show dependency tree for all commits
`--reverse`	Show commits in reverse order (Oldest commit first)
`--after`	Show all commits that happened after certain date
`--before`	Show all commits that happened before certain data

Example

```
$ git log --author="Sebastian" --after="1 week ago" --oneline
```

git log

You can use the regular less command to search

```
/{{your-search-here}}
```

Use lower case **`n`** to navigate to the next occurrence and upper case **`N`** to the previous one.

reset files

Return to a previous version and discard all changed

```
$ git reset --hard {{some-commit}}
```

I want to forget all the changes I've made, clean start `--hard HEAD`

Return to a previous version, changes are unstaged

```
$ git reset {{some-commit}}
```

I want to edit, re-stage and re-commit files in some different order

Return to a previous version, changes are staged

```
$ git reset --soft {{some-commit}}
```

I just want to re commit past 3 commits, as one big commit

ignore whitespace

You can easily ignore whitespace for diff and blame

```
$ git diff -w
$ git blame -w
```

partial add

add only some changes in a file

```
$ git add -p
```

os aliases

Use system aliases

```
# I’m lazy as hell
alias g='git'

# git status in a flash
alias gs='git status’
alias gss='git status -s’

# go to repository root directory
alias gr='[! -z `git rev-parse --show-cdup`] && cd `git rev-parse --show-cdup || pwd`'
```

git aliases

Add via terminal or edit ~/.gitconfig

```
$ git config --global alias.unstage "reset HEAD"
$ git config --global alias.wtf "log -p"
$ git config --global alias.l "log --oneline --graph"
$ git config --global alias.b "branch"
$ git config --global alias.c "commit"
$ git config --global alias.p "pull –rebase"
```

git rebase

> *Ahh, but the bliss of rebasing isn't without its drawbacks, which can be summed up in a single line:*
>
> *Do not rebase commits that exist outside your repository*
>
> *If you follow that guideline, you'll be fine. If you don't, people will hate you, and you'll be scorned by friends and family.*

–git book

git rebase

Committing on shared branches

> `Merge remote-tracking branch ‘origin/master’`

No big deal and completely safe, but still messes up the log history a bit.

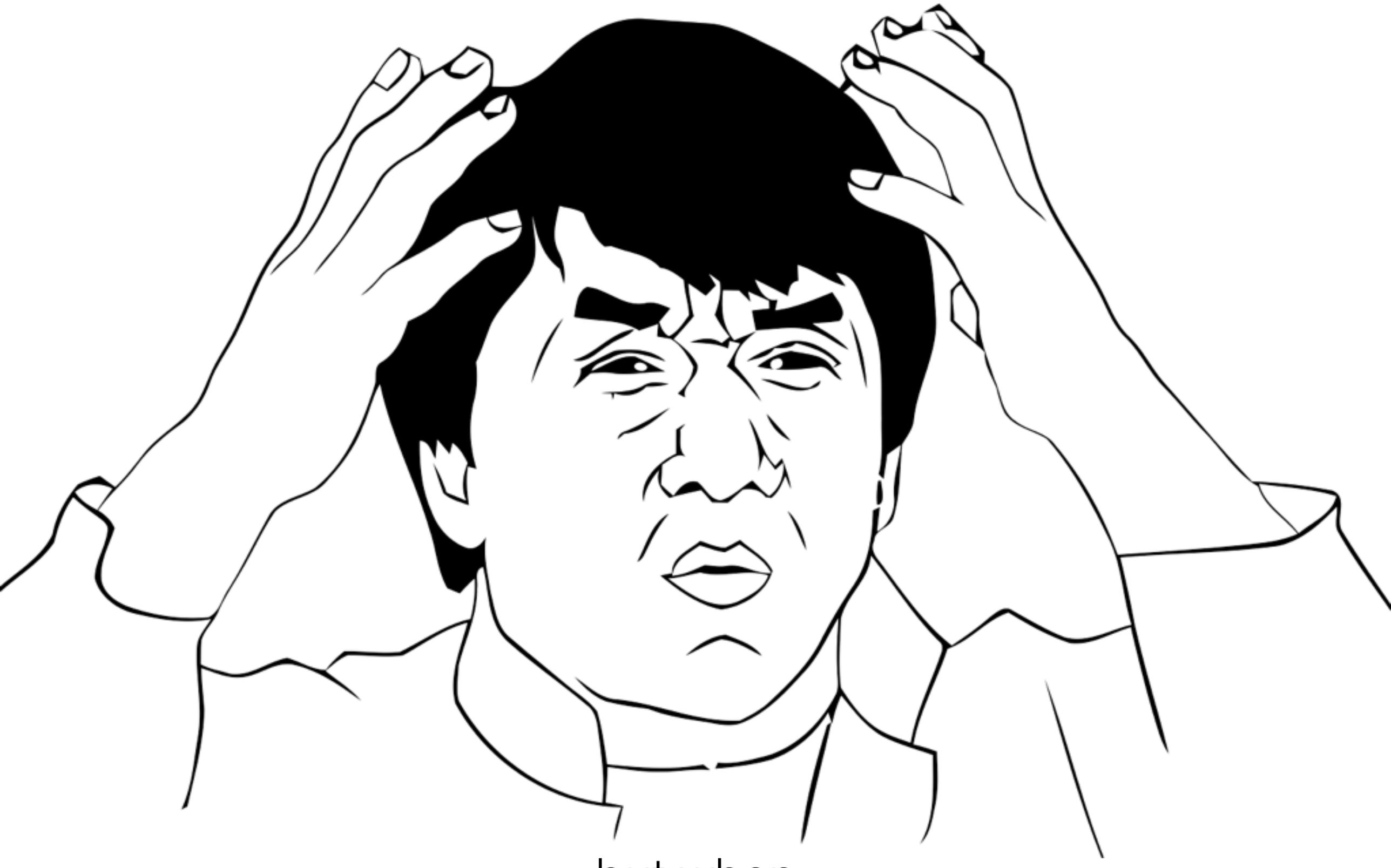
```
$ git pull --rebase
```

git amend

Let’s say you made a commit and realized you made a Typo.

```
$ git add path/toFile/withFixedTypo.php
$ git commit --amend
```

someone broke my stuff



but when

git bisect

Find commits that break your code fast

```
$ git bisect start
$ git bisect good {{some-commit}}
$ git bisect bad HEAD
```

Test and flag the commit

```
$ git bisect good|bad
```

git bisect

Go back to the starting point

```
$ git bisect reset
```

Get a summary report of last bisect run

```
$ git bisect log
```

Thanks

@movetodevnull