



GTR

REAL

HISTORY & CONFIGURATION

LEVEL 7

VIEWING THE LOG

```
$ git log
```

```
commit 915f242e262052b11c511dc07bef237fabb7ed85 ← SHA hash
```

```
Author: Gregg <gregg@codeschool.com>
```

```
Date: Thu Jun 28 02:10:57 2012 -0700
```

```
Update index. ← commit message
```


COLORIZING THE LOG

```
$ git config --global color.ui true
```

```
$ git log
```

```
commit 915f242e262052b11c511dc07bef237fabb7ed85
```

```
Author: Gregg <gregg@codeschool.com>
```

```
Date: Thu Jun 28 02:10:57 2012 -0700
```

```
Update index
```

```
$ git log --pretty=oneline
```

```
08f202691c67abd12eb886b587ac7b26d51005c7 Update index
```


LOG FORMAT

```
$ git log --pretty=format:"%h %ad- %s [%an]"
```

placeholder	replaced with
%ad	author date
%an	author name
%h	SHA hash
%s	subject
%d	ref names

any string you want
& placeholder data

run "git help log" for more options!

PATCH

```
$ git log --oneline -p
3ea7f70 I'm telling you, it's 'Octopi'.
diff --git a/index.html b/index.html
index 021a54e..640d66d 100644
--- a/index.html
+++ b/index.html
@@ -8,7 +8,7 @@
<nav>
  <ul>
    <li><a href="cat.html">Cats</a></li>
-   <li><a href="octopus.html">Octopuses</a></li>
+   <li><a href="octopi.html">Octopi</a></li>
```


STATS

```
$ git log --oneline --stat
```

```
3ea7f70 I'm telling you, it's 'Octopi'.
```

```
index.html | 2 +-  
1 file changed, 1 insertion(+), 1 deletion(-)
```

```
96776a4 Add index.
```

```
index.html | 30 ++++++-----  
1 file changed, 15 insertions(+), 15 deletions(-)
```


GRAPH

```
$ git log --oneline --graph
```

```
* 30b1f8f Merge branch 'bird' into master
```

```
| * 8b8f950 Revise silly hipster name for bird aisle.
```

```
* | 915f242 Add emphasis.
```

```
| /  
* 69728cd Update index descriptions.
```

visual representation of the branch merging into master

DATE RANGES

```
$ git log --until=1.minute.ago
```

until

```
$ git log --since=1.day.ago
```

since (days)

```
$ git log --since=1.hour.ago
```

since (hours)

```
$ git log --since=1.month.ago --until=2.weeks.ago
```

since & until (relative)

```
$ git log --since=2000-01-01 --until=2012-12-21
```

since & until (absolute)

DIFFS

```
$ git diff
```

```
diff --git a/index.html b/index.html
```

```
@@ -8,7 +8,10 @@
```

```
<nav>
```

```
<ul>
```

```
<li><a href="cat.html">Cats</a></li>
```

```
- <li><a href="octopus.html">Octopuses</a></li>
```

```
+ <li><a href="birds.html">Birds</a></li>
```

```
+ <li><a href="hamsters.html">Hamsters</a></li>
```

```
</ul>
```

```
</nav>
```

```
</body>
```

removed line

added lines

UNCOMMITTED CHANGES

```
$ git diff HEAD
```

```
diff --git a/index.html b/index.html
```

```
index 021a54e..1ceb9d6 100644
```

```
@@ -8,7 +8,10 @@
```

```
<ul>
```

```
  <li><a href="cat.html">Cats</a></li>
```

```
...
```

```
diff --git a/octopus.html b/octopus.html
```

```
index 55806be..ce8a2c7 100644
```

```
@@ -2,6 +2,6 @@
```

```
<html lang="en">
```

```
...
```

*diff between last commit
& current state*

includes both staged and unstaged files

EARLIER COMMITS

```
$ git diff HEAD^
```

parent of latest commit

```
$ git diff HEAD^^
```

grandparent of latest commit

```
$ git diff HEAD~5
```

five commits ago

```
$ git diff HEAD^..HEAD
```

second most recent commit vs. most recent

EARLIER COMMITS

```
$ git diff f5a6sdfsdfsdf9..4sdfsdfsdfsdfb063f
```

range of SHAs

```
$ git log --oneline
257256c cat
4fb063f Add index
f5a6ff9 Add catalog pages
```

```
$ git diff 4fb063f..f5a6ff9
```

range of abbreviated SHAs

```
$ git diff master bird
```

diff between two branches

```
$ git diff --since=1.week.ago --until=1.minute.ago
```

time-based diff

BLAME

```
$ git blame index.html --date short
```

```
...
```

```
96776a42 (Gregg 2012-06-29 9) <ul>
```

```
96776a42 (Gregg 2012-06-29 10) <li>Cats</li>
```

```
3ea7f709 (Jane 2012-06-30 11) <li>Octopi</li>
```

```
96776a42 (Gregg 2012-06-29 12) </ul>
```

```
...
```

commit hash

author

date

line #

content

EXCLUDING FILES

```
$ git status
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
# experiments/ ← we don't want to commit this...
```

.git/info/exclude

```
experiments/      will exclude this folder from git
```

```
$ git status
# On branch master
nothing to commit (working directory clean)
```

the experiment directory is now invisible to git

EXCLUDE PATTERNS

tutorial.mp4

exclude this file

*.mp4

exclude all .mp4 files

experiments/

exclude directory

logs/*.log

exclude .log files in logs directory

EXCLUDING FROM ALL COPIES

```
$ git status
```

```
# On branch master
```

```
# Untracked files:
```

```
#   (use "git add <file>..." to include in what will be committed)
```

```
#
```

```
# logs/server.log ← don't commit log files, they create conflicts
```

```
.gitignore
```

```
logs/*.log
```

add this pattern

```
$ git status
```

```
# On branch master
```

```
nothing to commit (working directory clean)
```

no more logs

REMOVING FILES

```
$ git rm README.txt
```

```
$ git status  
# Changes to be committed:  
#  
#   deleted:    README.txt
```

DELETED from the local filesystem & untracked

```
$ git commit -m "Remove readme"
```


UNTRACKING FILES

```
$ git rm --cached development.log
```

what if you're already tracking log files?

```
$ git status
```

```
# Changes to be committed:
```

```
#
```

```
# deleted:    development.log
```

not deleted from the local file system, only from Git

UNTRACKING FILES

.gitignore

logs/*.log *will ignore all .log files inside the logs directory*

\$ git add .gitignore

\$ git commit -m "Ignore all log files."

[master bccdc8c] Ignore all log files.

2 files changed, 1 insertion(+), 1 deletion(-)

create mode 100644 .gitignore

delete mode 100644 development.log

CONFIG

```
$ git config --global user.name "Gregg Pollack"  
$ git config --global user.email "gregg@codeschool.com"
```

remember these? there's more 

```
$ git config --global core.editor emacs
```

use emacs for interactive commands

```
$ git config --global merge.tool opendiff
```

use opendiff for merging conflicts

OS X only

LOCAL CONFIG

```
$ git config user.email "spamme@example.com" sets email for current repo
```

```
$ git config --list
```

```
user.name=Gregg
```

```
user.email=gregg@codeschool.com
```

```
color.ui=true
```

```
core.editor=mate -w
```

```
user.email=spamme@example.com
```

← same key can
be set twice

```
$ git config user.email
```

```
spamme@example.com
```

the global config loaded first, then repo config

ALIASES

aliases for log formats

```
$ git config --global alias.mylog \  
"log --pretty=format:'%h %s [%an]'" --graph"
```

```
$ git config --global alias.lol \  
"log --graph --decorate --pretty=oneline --abbrev-commit --all"
```

```
$ git mylog  
* 19f735c Merge branch 'admin' [Jane]  
| \  
| * 7980856 Add user admin [Jane]  
* | 5c9ed90 Add dashboard. [Jane]  
| /  
* ab48a3f Create quantum cat. [Jane]
```


ALIASES

git config alias.<name> <command>

```
$ git config --global alias.st status
```

git st ↔ git status

```
$ git config --global alias.co checkout
```

git co ↔ git checkout

```
$ git config --global alias.br branch
```

git br ↔ git branch

```
$ git config --global alias.ci commit
```

git ci ↔ git commit

```
$ git st
```

```
# On branch master
```

```
nothing to commit (working directory clean)
```




GTR

REAL