

Cheatsheet for the fugitive.vim VIM-Madrid meetup

 **cheatsheet**

```

1  # Basic commands
2
3      :Git [args]          # does what you'd expect
4
5  all of your `~/.gitconfig` aliases are available.
6
7      :Git! [args]         # same as before, dumping output to a tmp file
8
9  Moving inside a repo.
10
11      :Gcd [dir]           # :cd relative to the repository
12
13      :Glcd [dir]          # same as above, local to the current window
14
15  # Blame
16
17      :Gblame
18
19  - `A` resizes to end of author column
20  - `C` resizes to end of commit column
21  - `D` resizes to end of date/time column
22  - `` quits blame buffer, then opens commit
23
24  # Working with the index
25
26      :Gwrite              # add
27
28      :Gread               # checkout
29
30      :Gremove             # rm
31
32      :Gmove               # mv
33
34  in the `Gmove` context, `/` represents the root of the git repository.
35
36      :Gstatus             # brace yourselves
37
38      <CR> |:Gedit|
39      <C-n> and <C-p> for moving around files
40
41      - |:Git| add          # both work in Visual mode too
42      - |:Git| reset (staged files)
43
44      cc |:Gcommit|
45
46      D |:Gdiff|
47      ds |:Gsdiff|
48      dp |:Git!| diff (p for patch; use :Gw to apply)
49      dp |:Git| add --intent-to-add (untracked files)
50      dv |:Gvdiff|
51
52      o |:Gsplit|
53
54      p |:Git| add --patch
55
56      q close status
57      R reload status
58
59  # Diffs
60

```

```

61     diffget [bufspec]          # `do` can be used instead (*d*iff *o*btain)
62     diffput [bufspec]
63     diffupdate
64
65     `[c` and `]c` for navigating changes.
66
67     diffget //1 | diffupdate # `|` is used to chain operations
68
69 # Solving merge conflicts with 2-way diffs
70
71     :Gdiff                    # on the file with the conflict
72
73 # Browsing the git object database
74
75     `:Gedit`
76
77 can be used to open a _read-only_ buffer with the contents of any git object
78
79 ## Blobs
80
81     :Gedit branchname:path/to/file
82     :Gedit feature:%
83
84 ## Trees, commits and tags can be browsed too
85
86     :Gedit SHA                # will open a buffer with the git object referenced by `SHA`
87
88 # History
89
90 ## Revisions of a file
91
92     :Glog [args]              # opens a buffer with the revisions of the current file
93
94     `:cnext`, `:cprevious`, `:cfirst` and `:clast` can be used to move between revisions
95
96     :Glog -10                 # opens only the last 10 revisions
97
98 ## Commits
99
100     :Glog --                  # load all commits into the quickfix list
101
102     :Glog -- %                # load all commits that affected the current file into the quickfix list
103
104 # Searching
105
106 ## Working copy files
107
108     :Ggrep 'whatever'
109
110 ## Index
111
112     :Ggrep --cached 'whatever'
113
114 ## Branches
115
116     :Ggrep whatever branch   # search for `whatever` in `branch`
117
118 ## Tags
119
120     :Ggrep whatever tagname   # search for `whatever` in tag `tagname`
121
122 ## Commit messages
123
124     :Glog --grep=whatever --  # search for `whatever` in all commit messages
125
126     :Glog --grep=w00t -- %    # search for `w00t` in all commit messages that touched the current file
127
128 ## Added or removed words
129
130     :Glog -Swhatever --       # search for `whatever` in the diff for each commit
131     :Glog -Swhatever -- %     # search for `whatever` in the diff for each commit that touched the current file
132
133 # Status line

```

```
134
135     set statusline=%<f\ %h%m%r%{fugitive#statusline()}%=-14.(%l,%c%V%)\ %P
136
137 # Further information
138
139     :help fugitive
140
141 Vimcasts on fugitive.vim
142
143 - http://vimcasts.org/episodes/fugitive-vim---a-complement-to-command-line-git/
144 - http://vimcasts.org/episodes/fugitive-vim-working-with-the-git-index/
145 - http://vimcasts.org/episodes/fugitive-vim-resolving-merge-conflicts-with-vimdiff/
146 - http://vimcasts.org/episodes/fugitive-vim-browsing-the-git-object-database/
147 - http://vimcasts.org/episodes/fugitive-vim-exploring-the-history-of-a-git-repository/
148
149 # Related plugins
150
151 - vim-gitgutter
152 - vim-mercenary (`fugitive` port for mercurial)
```