

In []:

```
1  Introducción a los comandos básicos de git para trabajar de manera local.
2
3  1. Secuencia de Introducción
4      1.1 Introducción a Git
5          git commit;
6
7          git commit;
8      1.2 Branching en Git
9
10         git branch bugFix;
11         git checkout bugFix;
12
13      1.3 Mergin en Git
14         git checkout -b bugFix;
15
16         git commit;
17
18         git checkout master;
19
20         git commit;
21
22         git merge bugFix;
23
24      1.4 Introducción a Rebase
25         git checkout -b bugFix;
26
27         git commit;
28
29         git checkout master;
30
31         git commit;
32
33         git checkout bugFix;
34
35         git rebase master;
36
37  2. Acelerando
38      2.1 Detach yo' HEAD
39
40         git checkout C4;
41      2.2 Relative refs (^)
42
43         git branch -f master C6;
44
45         git branch -f bugFix C0;
46
47         git checkout C1;
48
49      2.3 Revertir cambios en Git
50         git reset local~1;
51
52         git checkout pushed;
53
54         git revert pushed;
55
56  3. Moviendonos entre trabajo
57      3.1 Introducción a Cherry-pick
58
59         git cherry-pick C3 C4 C7;
```

3.2 Introducción interactiva a rebase

```
git rebase -i master~4 --aboveAll;
```

4. A mixed bag

4.1 Grabbing just 1 commit

```
git checkout master;
```

```
git cherry-pick C4;
```

4.2 Juggling commits

```
git rebase -i caption~2 --aboveAll;
```

```
git commit --amend;
```

```
git rebase -i caption~2 --aboveAll;
```

```
git branch -f master caption;
```

4.3 Juggling commits #2

```
git checkout master;
```

```
git cherry-pick C2;
```

```
git commit --amend;
```

```
git cherry-pick C3;
```

4.5 Git tags

```
git tag v0 C1;
```

```
git tag v1 C2;
```

```
git checkout C2;
```

4.6 Git describe

```
git commit;
```

5. Topics avanzados

5.1 Rebasing over 9000 times

```
git rebase master bugFix;
```

```
git rebase bugFix side;
```

```
git rebase side another;
```

```
git rebase another master;
```

5.2 Multiple parents

```
git branch bugWork master~^2~;
```

5.6 Branch Spaghetti

```
git checkout one;
```

```
121
122     git cherry-pick C4 C3 C2;
123
124     git checkout two;
125
126     git cherry-pick C5 C4 C3 C2;
127
128     git branch -f three C2;
129
130 ## REMOTE
131 Aprendiendo comandos Git para trabajar en remoto
132
133 1. Push and Pull
134     1.1 Introducción a Clone
135
136         git clone;
137     1.2 Branchs remotas
138
139         git commit;
140         git checkout o/master;
141         git commit;
142     1.3 Git fetching
143
144         git fetch;
145     1.4 Git pulling
146
147         git pull;
148     1.5 Faking Teamwork
149
150         git clone;
151         git fakeTeamwork master 2;
152         git commit;
153         git pull;
154     1.6 Git pushing
155
156         git clone;
157         git commit;
158         git commit;
159         git push;
160     1.7 Diverged History
161
162         git clone;
163         git fakeTeamwork;
164         git commit;
165         git pull --rebase;
166         git push;
167     1.8 Locked master
168
169         git branch feature;
170         git checkout feature;
171         git push;
172         git branch -f master HEAD~1
173
174 2. Git remoto avanzado
175     2.1 Push master
176
177         git fetch;
178         git rebase o/master side1;
179         git rebase side1 side2;
180         git rebase side2 side3;
181         git rebase side3 master;
```

```
182     git push;
183 2.2 Merging with remotes
184
185     git checkout master;
186     git pull;
187     git merge side1;
188     git merge side2;
189     git merge side3;
190     git push;
191 2.3 Remote Tracking
192
193     git checkout -b side o/master;
194     git commit;
195     git pull --rebase;
196     git push;
197 2.4 Git push arguments
198
199     git push origin master;
200     git push origin foo;
201 2.5 Git push arguments – Expanded!
202
203     git push origin master~1:foo;
204     git push origin foo:master;
205 2.6 Fetch arguments
206
207     git fetch origin master~1:foo;
208     git fetch origin foo:master;
209     git checkout foo;
210     git merge master;
211 2.7 Source of nothing
212
213     git push origin :foo;
214     git fetch origin :bar;
215 2.8 Pull arguments
216
217     git pull origin bar:foo;
218     git pull origin master:side;
```