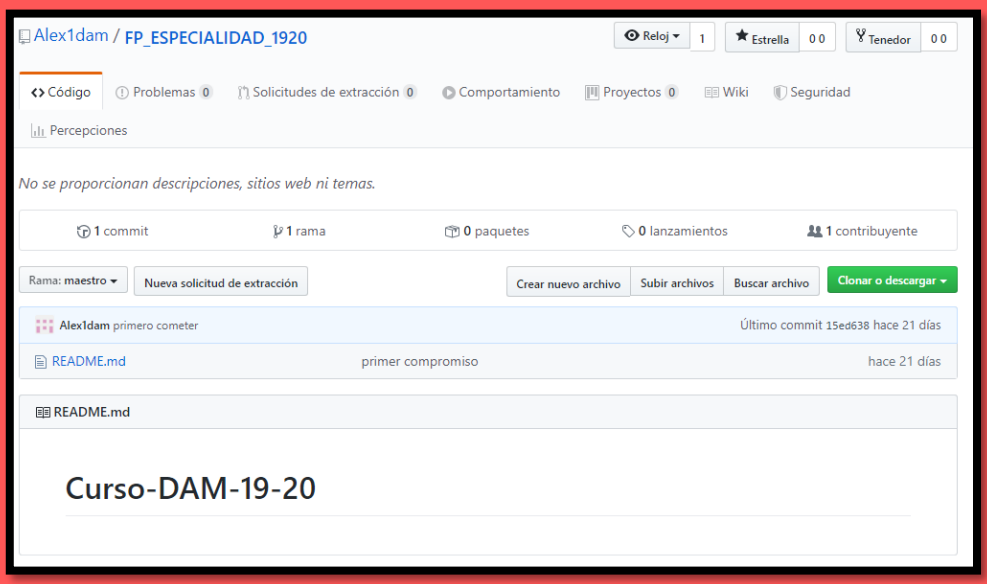
PRACTICA 1

GITHUB AVANZADO

JOSE JAIR BARRUETO VENTURA

2.1.2 – Repositorio remoto

1)



2)

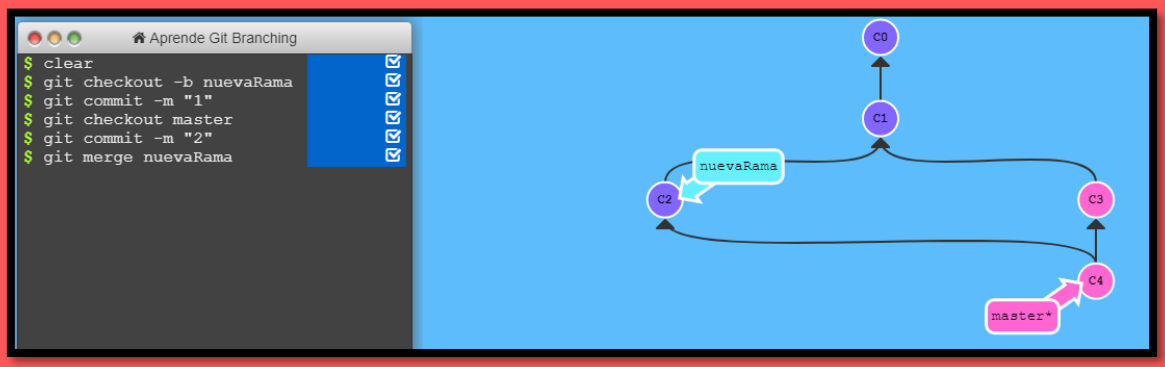
```
DELL@DESKTOP-OMJQVIP MINGW64 ~ (master)
$ git clone https://github.com/Alex1dam/FP_ESPECIALIDAD_1920.git
Cloning into 'FP_ESPECIALIDAD_1920'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 224 bytes | 13.00 KiB/s, done.

DELL@DESKTOP-OMJQVIP MINGW64 ~ (master)
$
```

3)

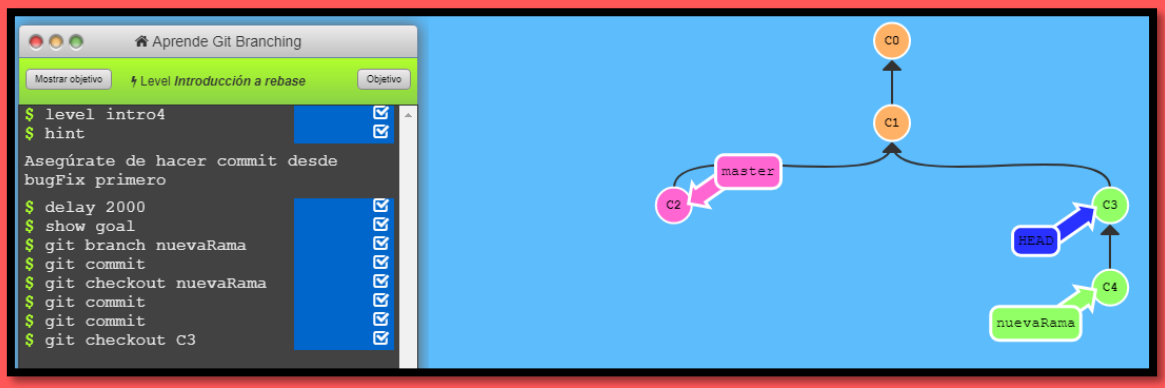
```
DELL@DESKTOP-OMJQVIP MINGW64 ~/desktop/FP_ESPECIALIDAD_1920 (master)
$ dir
L_MARCAS  README.md

DELL@DESKTOP-OMJQVIP MINGW64 ~/desktop/FP_ESPECIALIDAD_1920 (master)
$ git add L_MARCAS

DELL@DESKTOP-OMJQVIP MINGW64 ~/desktop/FP_ESPECIALIDAD_1920 (master)
$ git commit -m "subida ejercicio avanzado"
[master d23452b] subida ejercicio avanzado
 6 files changed, 53 insertions(+)
 create mode 100644 Desktop/FP_ESPECIALIDAD_1920/L_MARCAS/DTD.pdf
 create mode 100644 eclipse-workspace/2020/.classpath
 create mode 100644 eclipse-workspace/2020/.project
 create mode 100644 eclipse-workspace/2020/.settings/org.eclipse.jdt.core.prefs
 create mode 100644 eclipse-workspace/2020/bin/calculadora.class
 create mode 100644 eclipse-workspace/2020/src/calculadora.java
```

GITHUB

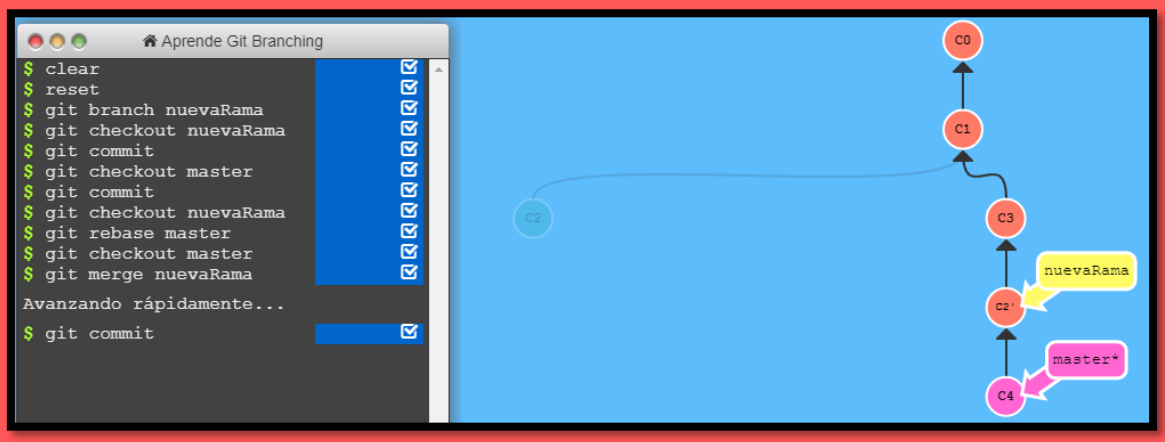
1 -


Aprende Git Branching
$ clear
$ git checkout -b nuevaRama
$ git commit -m "1"
$ git checkout master
$ git commit -m "2"
$ git merge nuevaRama
C0
C1
nuevaRama
C2
C3
C4
master*

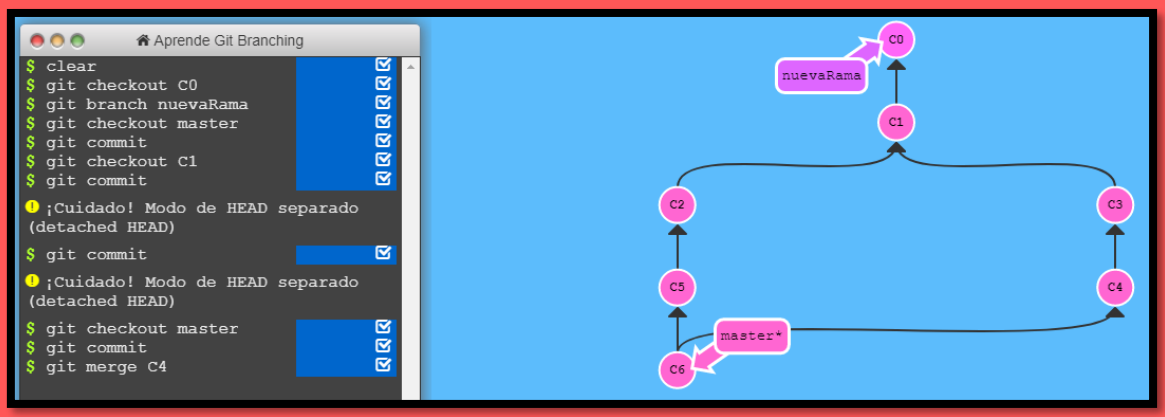
2-


Aprende Git Branching
Mostrar objetivo
Level Introducción a rebase
Objetivo
$ level intro4
$ hint
Asegúrate de hacer commit desde bugFix primero
$ delay 2000
$ show goal
$ git branch nuevaRama
$ git commit
$ git checkout nuevaRama
$ git commit
$ git commit
$ git checkout C3
C0
C1
master
C2
C3
HEAD
C4
nuevaRama

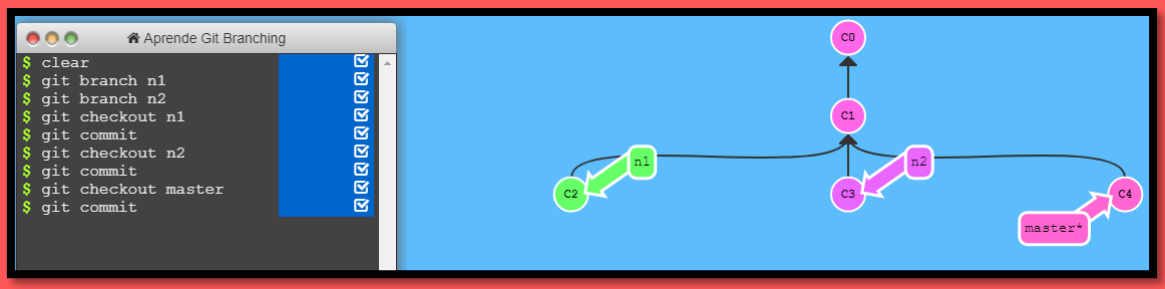
3-


Aprende Git Branching
$ clear
$ reset
$ git branch nuevaRama
$ git checkout nuevaRama
$ git commit
$ git checkout master
$ git commit
$ git checkout nuevaRama
$ git rebase master
$ git checkout master
$ git merge nuevaRama
Avanzando rápidamente...
$ git commit
C0
C1
C2
C3
nuevaRama
C2'
master*
C4

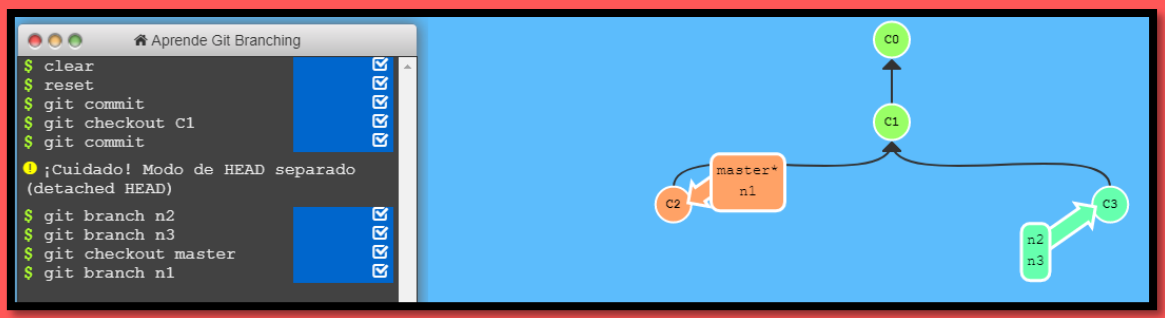
4-


Aprende Git Branching
$ clear
$ git checkout C0
$ git branch nuevaRama
$ git checkout master
$ git commit
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout master
$ git commit
$ git merge C4
C0
nuevaRama
C1
C2
C3
C5
C4
master*
C6

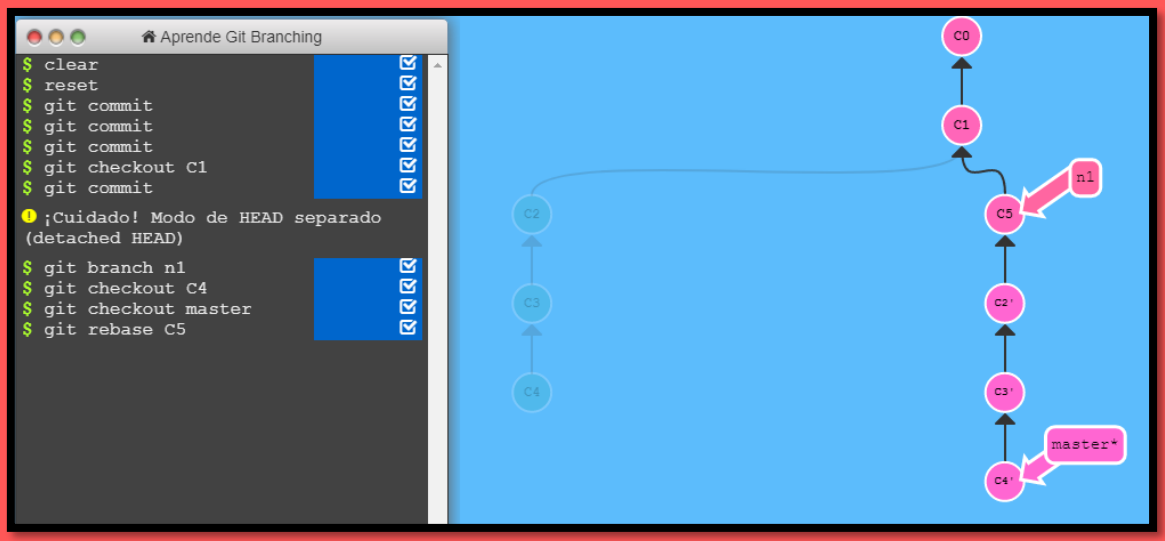
5-


Aprende Git Branching
$ clear
$ git branch n1
$ git branch n2
$ git checkout n1
$ git commit
$ git checkout n2
$ git commit
$ git checkout master
$ git commit
C0
C1
n1
n2
C2
C3
C4
master*

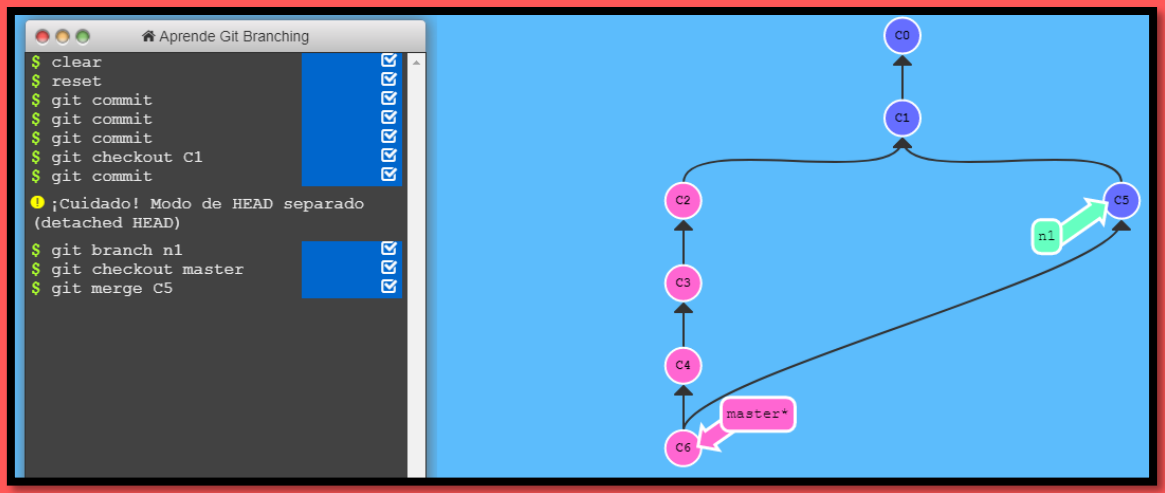
6-


Aprende Git Branching
$ clear
$ reset
$ git commit
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git branch n2
$ git branch n3
$ git checkout master
$ git branch n1
C0
C1
master*
n1
C2
C3
n2
n3

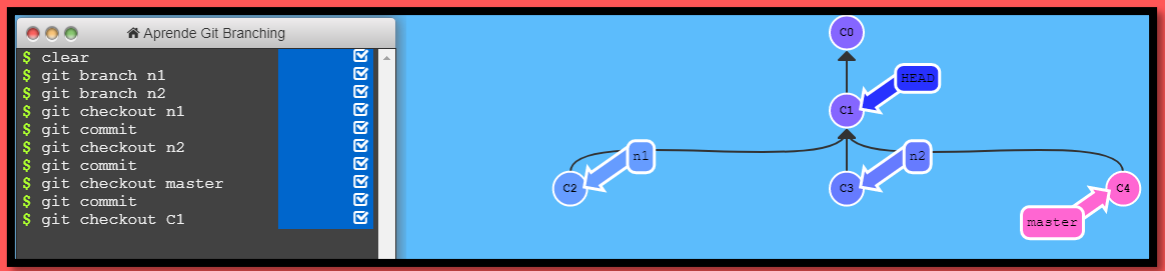
7-


Aprende Git Branching
$ clear
$ reset
$ git commit
$ git commit
$ git commit
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git branch n1
$ git checkout C4
$ git checkout master
$ git rebase C5
C0
C1
C2
C3
C4
n1
C5
C2'
C3'
master*
C4'

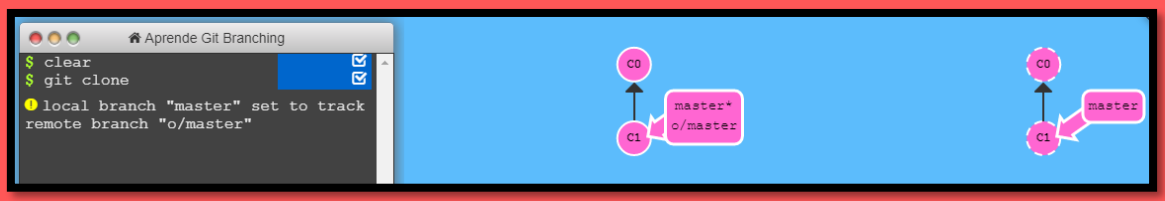
8-


Aprende Git Branching
$ clear
$ reset
$ git commit
$ git commit
$ git commit
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git branch n1
$ git checkout master
$ git merge C5
C0
C1
C2
C3
C4
C6
master*
n1
C5

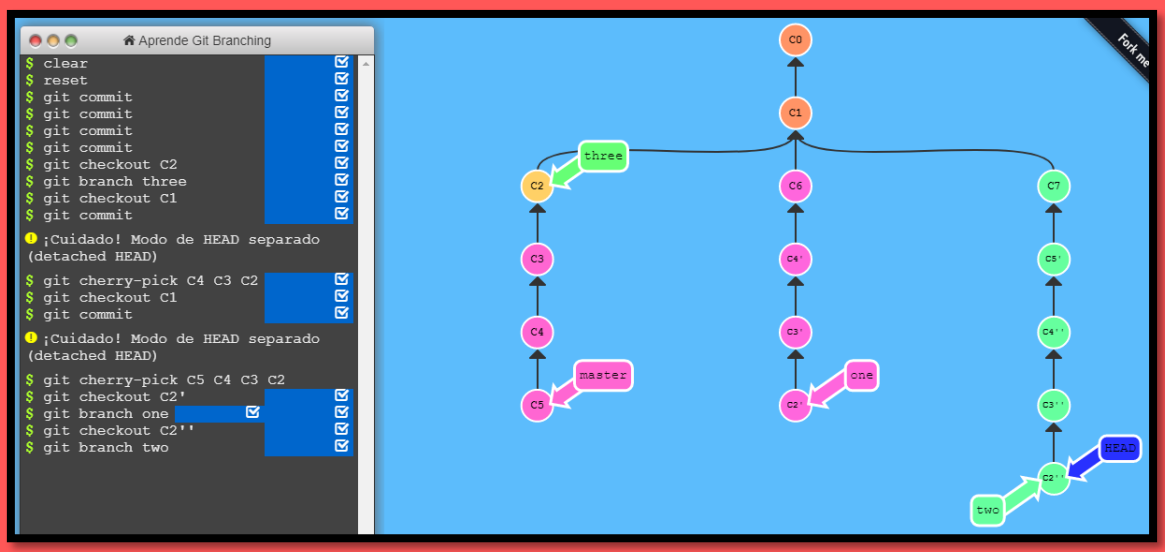
9-


Aprende Git Branching
$ clear
$ git branch n1
$ git branch n2
$ git checkout n1
$ git commit
$ git checkout n2
$ git commit
$ git checkout master
$ git commit
$ git checkout C1
C0
HEAD
C1
n1
C2
n2
C3
C4
master

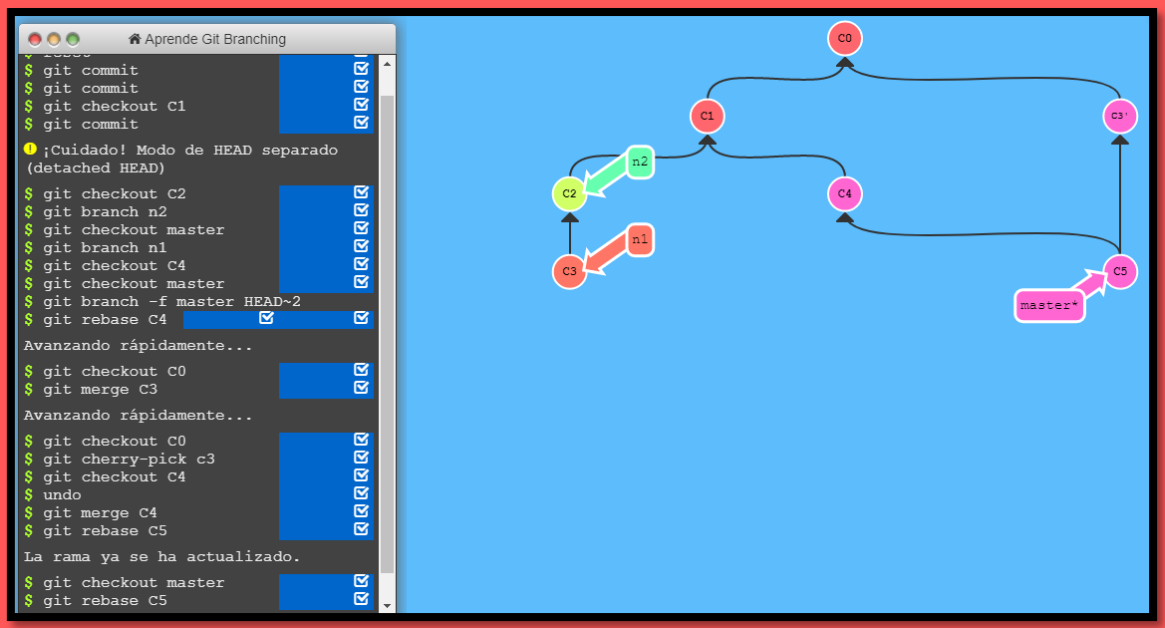
10-


Aprende Git Branching
$ clear
$ git clone
local branch "master" set to track remote branch "o/master"
C0
C1
master*
o/master
C0
C1
master

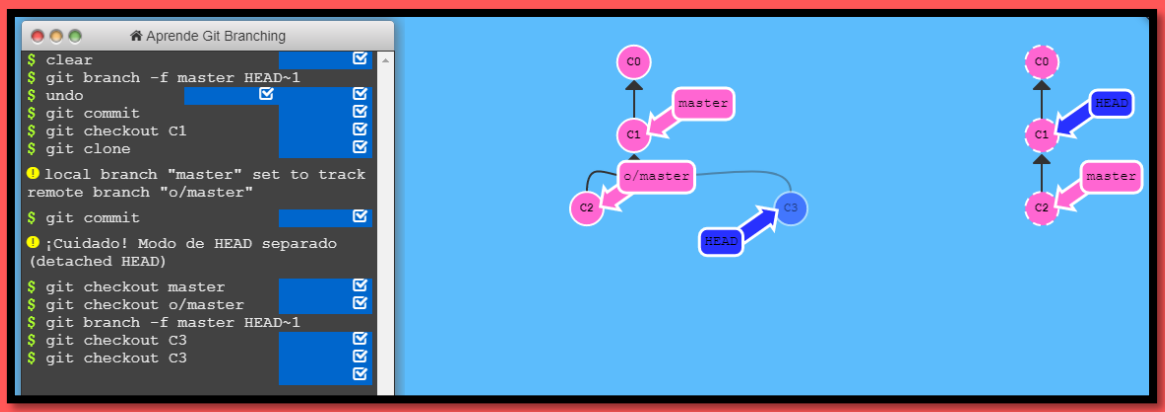
11-


Aprende Git Branching
$ clear
$ reset
$ git commit
$ git commit
$ git commit
$ git commit
$ git checkout C2
$ git branch three
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git cherry-pick C4 C3 C2
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git cherry-pick C5 C4 C3 C2
$ git checkout C2'
$ git branch one
$ git checkout C2''
$ git branch two
Fork me
C0
C1
three
C2
C6
C7
C3
C4'
C5'
C4
C3'
C4''
master
C5
one
C2'
C3''
HEAD
C2''
two

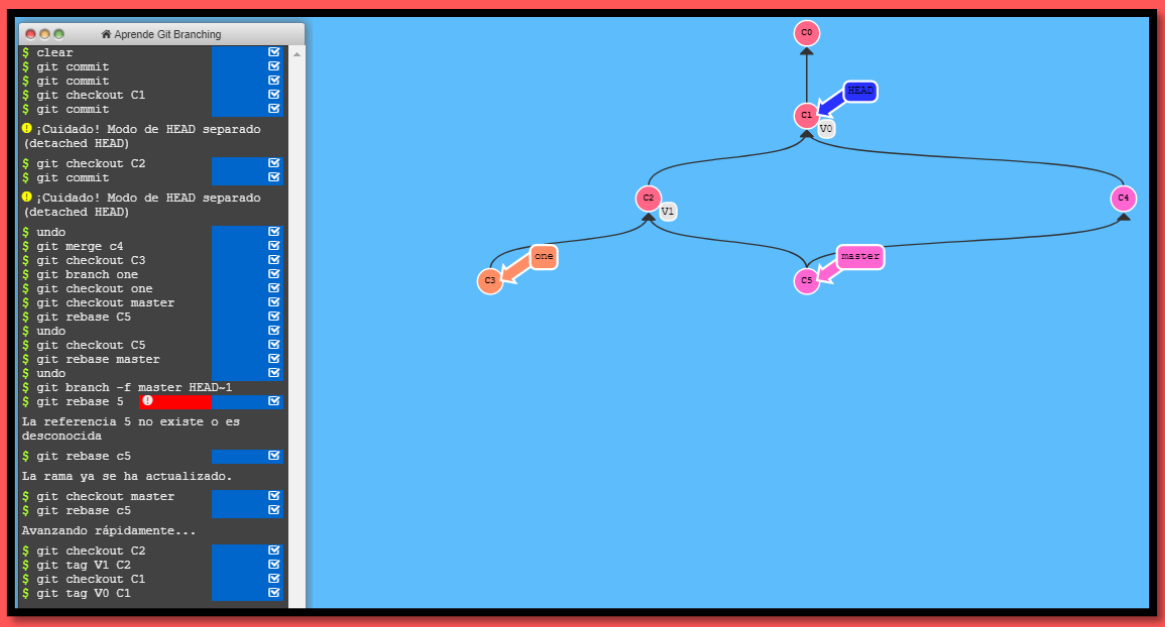
12-


Aprende Git Branching
$ git commit
$ git commit
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout C2
$ git branch n2
$ git checkout master
$ git branch n1
$ git checkout C4
$ git checkout master
$ git branch -f master HEAD~2
$ git rebase C4
Avanzando rápidamente...
$ git checkout C0
$ git merge C3
Avanzando rápidamente...
$ git checkout C0
$ git cherry-pick c3
$ git checkout C4
$ undo
$ git merge C4
$ git rebase C5
La rama ya se ha actualizado.
$ git checkout master
$ git rebase C5
C0
C1
C3'
n2
C2
C4
n1
C3
C5
master*

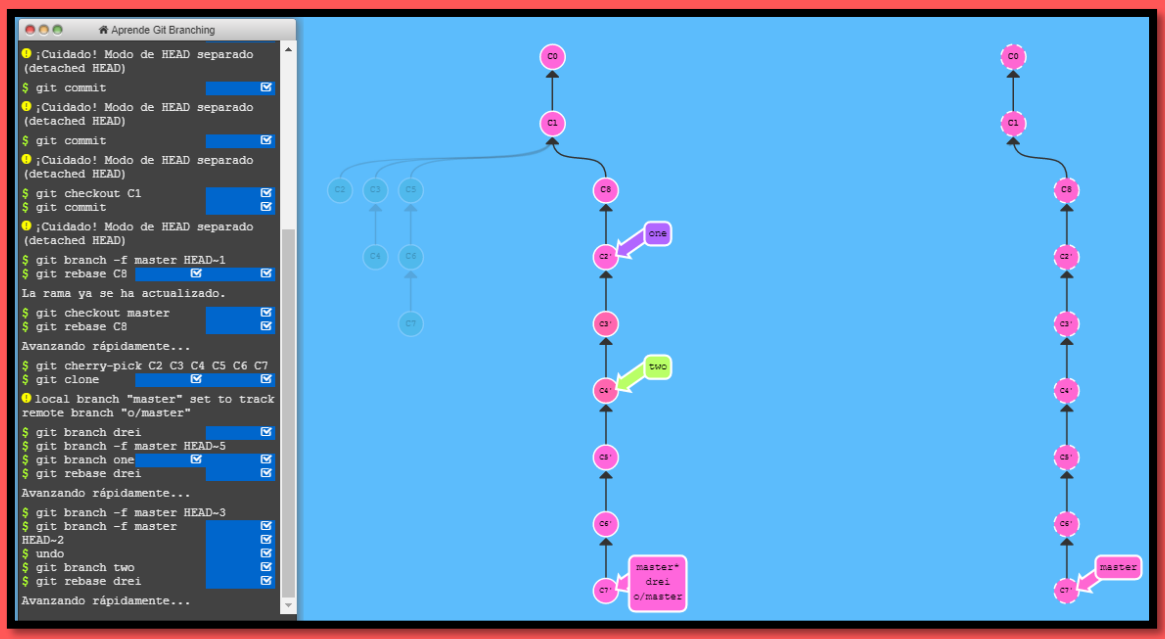
13-


Aprende Git Branching
$ clear
$ git branch -f master HEAD~1
$ undo
$ git commit
$ git checkout C1
$ git clone
local branch "master" set to track remote branch "o/master"
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout master
$ git checkout o/master
$ git branch -f master HEAD~1
$ git checkout C3
$ git checkout C3
C0
C1
C2
C3
master
o/master
HEAD
C0
C1
C2
HEAD
master

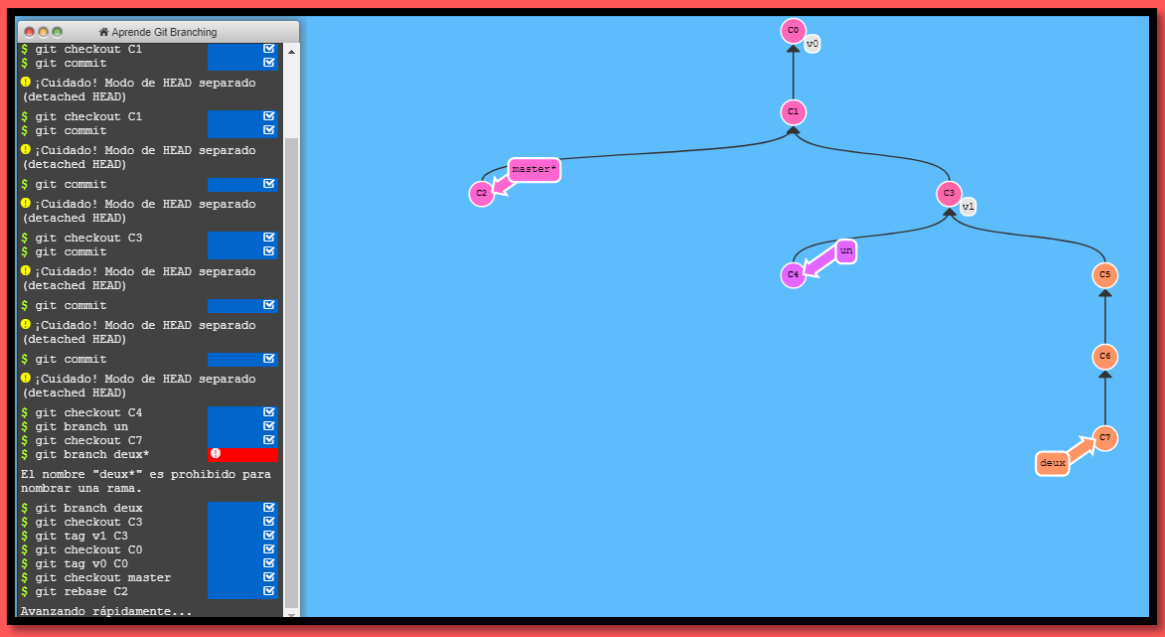
14-


Aprende Git Branching
$ clear
$ git commit
$ git commit
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout C2
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ undo
$ git merge c4
$ git checkout C3
$ git branch one
$ git checkout one
$ git checkout master
$ git rebase C5
$ undo
$ git checkout C5
$ git rebase master
$ undo
$ git branch -f master HEAD~1
$ git rebase 5
La referencia 5 no existe o es desconocida
$ git rebase c5
La rama ya se ha actualizado.
$ git checkout master
$ git rebase c5
Avanzando rápidamente...
$ git checkout C2
$ git tag V1 C2
$ git checkout C1
$ git tag V0 C1
C0
HEAD
C1
V0
C2
V1
C4
one
C3
master
C5

15-


Aprende Git Branching
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git branch -f master HEAD~1
$ git rebase C8
La rama ya se ha actualizado.
$ git checkout master
$ git rebase C8
Avanzando rápidamente...
$ git cherry-pick C2 C3 C4 C5 C6 C7
$ git clone
local branch "master" set to track remote branch "o/master"
$ git branch drei
$ git branch -f master HEAD~5
$ git branch one
$ git rebase drei
Avanzando rápidamente...
$ git branch -f master HEAD~3
$ git branch -f master HEAD~2
$ undo
$ git branch two
$ git rebase drei
Avanzando rápidamente...
C0
C1
C8
one
two
master*
drei
o/master
master

16-


Aprende Git Branching
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout C1
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout C3
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git commit
¡Cuidado! Modo de HEAD separado (detached HEAD)
$ git checkout C4
$ git branch un
$ git checkout C7
$ git branch deux*
El nombre "deux*" es prohibido para nombrar una rama.
$ git branch deux
$ git checkout C3
$ git tag v1 C3
$ git checkout C0
$ git tag v0 C0
$ git checkout master
$ git rebase C2
Avanzando rápidamente...
C0
v0
C1
master*
C2
C3
v1
un
C4
C5
C6
C7
deux