**Desde un repositorio existente realizar una copia local**

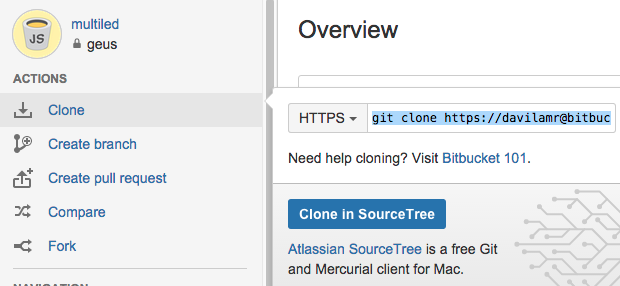
**Paso 1 - Crear el directorio local e ingresar en el.**

$ mkdir repositorio

$ cd repositorio

**Paso 2 - Obtener el comando para clonar el repositorio desde el sitio.**

Ir al sitio de Bitbucket <https://bitbucket.org> e ingresar al repositorio que se desea clonar, luego hacer click en el boton Clone y copiar el comando:



**Paso 3 - Clonar repositorio remoto**

Para ello, se debe pegar el comando en la terminal e ingresar el Password cuando el mismo es solicitado (es el mismo que utiliza para loguearse en Bitbucket)

$ git clone https://usuario@bitbucket.org/dueño/repositorio.git

Cloning into 'bitbucketstationlocations'...

Password

**¡Listo el repositorio ya está clonado en nuestro equipo!**

**Inicializando un repositorio en un proyecto existente**

**Paso 0 - Realizar el Clone**

**Paso 1 - Ingresar en el directorio e inicializar git**

$ cd directorio\_del\_proyecto

$ git init

**Paso 2 -Especificar qué archivos se desea controlar**

$ git add \*

**Paso 3 -Realizar un commit para confirmar los cambios**

$ git commit -m 'Descripción del primer commit'

**Paso 4 - Enviar a un repositorio remoto.**

Cuando tu proyecto se encuentra en un estado que quieres compartir, tienes que enviarlo a un repositorio remoto. El comando que te permite hacer esto es: **git push** [nombre-remoto] [nombre-rama]. Para enviar tu branch (master) a tu servidor origen (origin), ejecutamos:

$ git push [origin] [master]

**Importante:**

Si en este momento realizamos un cambio en un archivo y consultamos el estatus de git con el comando “git status” veremos lo siguiente:

$ git status

On branch master

Your branch is up-to-date with 'origin/master'.

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

modified: www/index.html

no changes added to commit (use "git add" and/or "git commit -a")

Esto se debe a que cuando modificamos un archivo este pasa al estado de modificado y deberemos al momento de realizar el commit del mismo utilizar el siguiente comando:

$ git commit **-a** -m 'Descripción del segundo commit'

Para luego repetir el paso número cuatro (push).

**Crear un nuevo branch**

$ git checkout -b nuevo\_branch

**Ver los branch remotos**

$ git remote show origin

**Traer cambios remotos**

$ git pull origin nombre\_branch

**Cambiar de branch**

$ git checkout otro\_branch

**Si no reconoce el nuevo branch, probar fetch para que tome los cambios remotos**

**$ git fetch**

**Eliminar un branch**

Remoto:

$ git push origin --delete <branchName>

Local:

$ git branch -d <branchName>

[**http://pcottle.github.io/learnGitBranching/**](http://pcottle.github.io/learnGitBranching/)