



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 - Programación Avanzada
1^{er} semestre 2017

Comandos Básicos y Uso de Git

Comandos básicos del terminal

- `cd <directorio>`: sirve para cambiar de directorio al directorio especificado. Por ejemplo:

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.10586]
(c) 2015 Microsoft Corporation. Todos los derechos reservados.

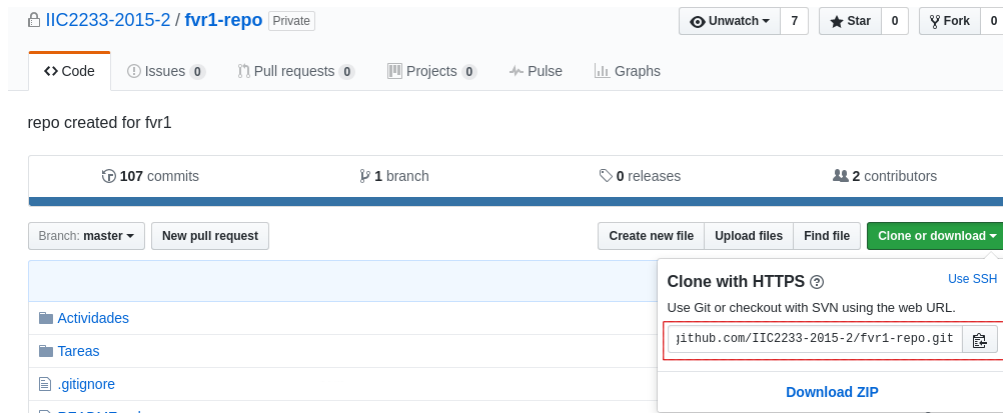
C:\Users\Florencia>
C:\Users\Florencia>cd Desktop\fvr1-repo
C:\Users\Florencia\Desktop\fvr1-repo>
```

Al hacer `cd ..` uno se devuelve a la carpeta "padre".

- `dir`: muestra todos los elementos dentro de la carpeta actual. En Linux o Mac se utiliza el comando `ls`.
- `pwd`: indica el *path* de la carpeta actual.

Clonar repositorio

Para clonar un repositorio en tu computador, debes navegar en el terminal hacia el *path* de la carpeta en donde quieras almacenarlo mediante el comando `cd`. Usa el comando `git clone <link>`. El link que debes utilizar se encuentra en tu repositorio, donde lo indica la siguiente imagen:



Se clonará la carpeta en el lugar indicado, siendo este el *repositorio local*. Si quieres clonar tu repositorio y usar otro nombre para la carpeta, puedes utilizar el comando `git clone <link> <nuevo_nombre>`.

Descargar cambios

El comando `git pull` sirve para actualizar el contenido de tu repositorio local con el contenido nuevo que se encuentra en el directorio de la web. Para esto, debes navegar en el terminal hasta la carpeta de tu repositorio y utilizar el comando `git pull`. Si es un repositorio privado, seguramente deberás autenticarte con tus credenciales de Github.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.10586]
(c) 2015 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Florencia>cd Desktop\IIC2233\2016-2\syllabus

C:\Users\Florencia\Desktop\IIC2233\2016-2\syllabus>git pull
remote: Counting objects: 13, done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 13 (delta 6), reused 10 (delta 3), pack-reused 0
Unpacking objects: 100% (13/13), done.
From https://github.com/IIC2233-2016-02/syllabus
 4cdd976..74d62b8 master -> origin/master
Updating 4cdd976..74d62b8
Fast-forward
Actividades/AC00/main1.pdf | Bin 0 -> 100161 bytes
 README.md                 | 6 +++---
 2 files changed, 3 insertions(+), 3 deletions(-)
 create mode 100644 Actividades/AC00/main1.pdf

C:\Users\Florencia\Desktop\IIC2233\2016-2\syllabus>
```

Subir cambios

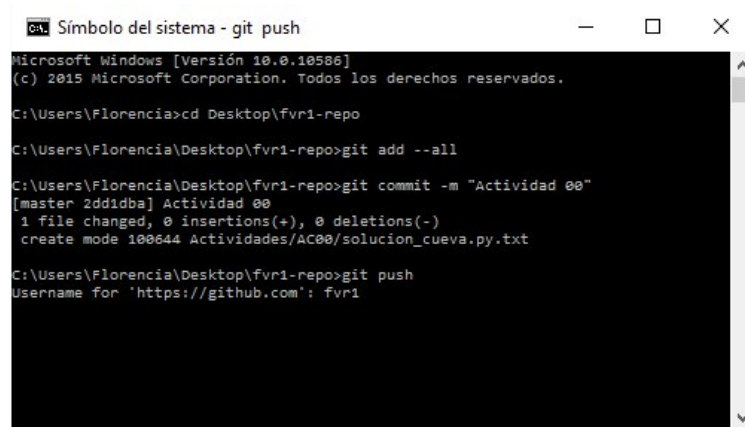
El comando `git push` se podría considerar como el inverso de `git pull`, es decir, sirve para enviar las actualizaciones de tu repositorio local al directorio de la web. Para esto debes:

1. Abrir la carpeta de tu repositorio en tu computador desde el terminal, navegando hacia el directorio correspondiente.
2. Desde el terminal, utiliza el comando `git add <nombre_archivo>` para subir o actualizar un archivo en específico. Asimismo, puedes utilizar `git add <carpeta> /*` para agregar todos los archivos modificados en cierto directorio, incluyendo los archivos creados o eliminados. También, puedes ejecutar

el comando `git add --all` (fijarse en la sintaxis) para agregar todos los cambios hechos en todo tu repositorio.

3. Escribe el comando `git commit -m "<mensaje_para_identificar_actualización>"`. Notar que el mensaje va entre comillas dobles. Un *commit* es básicamente el estado de un repositorio en ese instante de tiempo.
4. Finalmente, ejecuta `git push` para enviar tus cambios al directorio de trabajo. Seguramente, deberás autenticarte en Github para poder realizar esta acción.

En resumen:



```
Símbolo del sistema - git push
Microsoft Windows [Versión 10.0.10586]
(c) 2015 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Florencia>cd Desktop\fvr1-repo

C:\Users\Florencia\Desktop\fvr1-repo>git add --all

C:\Users\Florencia\Desktop\fvr1-repo>git commit -m "Actividad 00"
[master 2dd1dba] Actividad 00
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Actividades/AC00/solucion_cueva.py.txt

C:\Users\Florencia\Desktop\fvr1-repo>git push
Username for 'https://github.com': fvr1
```

Para saber qué cambios se realizaron puedes usar el comando `git status`.

TIP IMPORTANTE

Antes de hacer `push`, asegúrate de que no existan cambios no sincronizados entre el directorio de la web y el directorio local, ya que en ese caso se lanzará un error al ejecutar el comando (y puede ser uno bien feo). Por esto, antes de hacer `push` se recomienda hacer `pull`. Notar que al hacer `git pull` no se van a ver afectados los nuevos cambios que hayas hecho en el repositorio local.

Cómo subir tareas y actividades

Las actividades en cátedra las tendrás que subir al directorio `Actividades/ACXX`, siendo `XX` el número que corresponde a la actividad. El enunciado no debe ser subido. Asegúrate también de subir todo antes de las 16:50 hrs. además de hacerlo en la carpeta con el formato indicado, ya que en el caso contrario **tu actividad no será recolectada**.

Para las actividades realizadas en pareja, basta con que un miembro suba la actividad. Si ambos miembros llegasen a subir la actividad, tiene que ser el mismo archivo, de lo contrario el corrector decidirá cuál corregir.

Las tareas se suben similarmente a las actividades, pero en la carpeta `Tareas/TXX` siendo `'XX'` el número de la tarea. En cada tarea debes hacer un archivo llamado `README.md` que contenga información referente a cómo hiciste la tarea, y todas las consideraciones que el corrector debe tomar al revisarla. Si quieres saber más detalles sobre cómo hacer el `README` ingresa a este link.