



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

IIC2233 - Programación Avanzada
2^{do} semestre 2016

Comandos Básicos y Uso de Git

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'. Puedes utilizar el comando 'dir' en Windows o 'ls' en Mac y Linux para mostrar el contenido en el directorio actual. Por ejemplo:

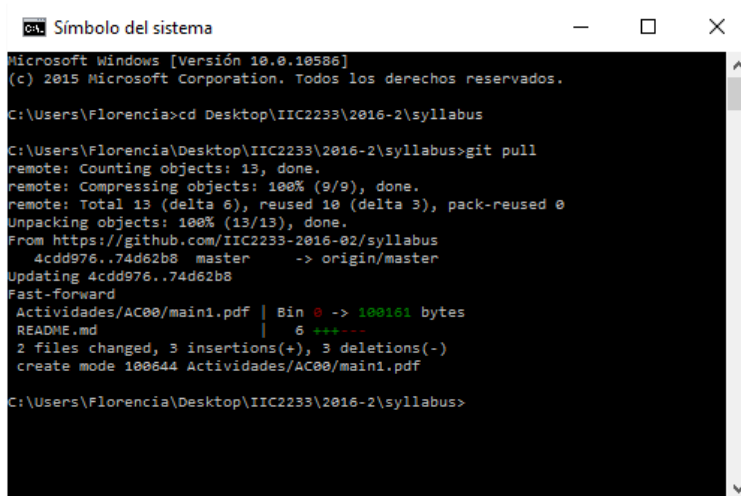
```
Símbolo del sistema
Microsoft Windows [Versión 10.0.10586]
(c) 2015 Microsoft Corporation. Todos los derechos reservados.

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

Luego, usa el comando 'git clone <link>', donde el link que debes utilizar se encuentra en la URL de tu repositorio. Se clonará la carpeta en el lugar indicado siendo esta el repositorio local.

Bajar Cambios

El comando 'git pull' sirve para actualizar el contenido de tu repo 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 repo y utilizar el comando 'git pull'. Si es un repositorio privado seguramente deberás autenticarte con tus credenciales de Github.



```
CA 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

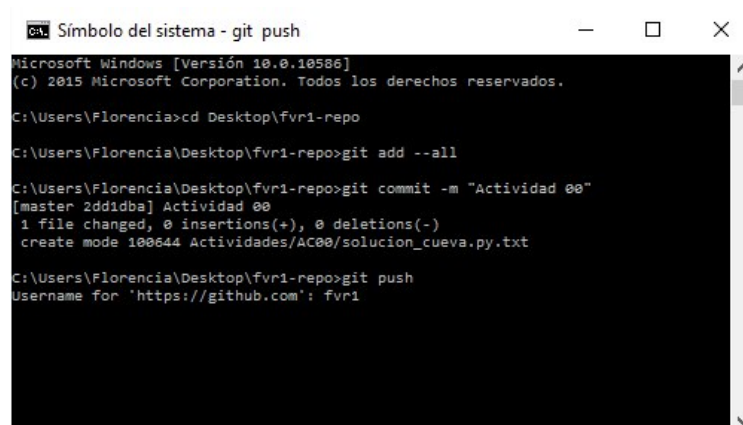
C:\Users\Florencia\Desktop\IIC2233\2016-2\syllabus>
```

Subir Cambios

El comando 'git push' se podría considerar como el inverso de 'git pull', es decir, sirve para enviar las actualizaciones de tu repo local al del directorio de la web. Para saber cuáles cambios se realizaron puedes usar el comando 'git status'. 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, ejecutar el comando "git add --all" (fijarse en la sintaxis) para agregar todos los cambios hechos en tu repositorio, es decir, todos los archivos modificados, creados o eliminados. También puedes utilizar "git add <nombre_archivo >" para subir o actualizar un archivo en específico.
3. Escribir 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, ejecutar '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\fvri-repo

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

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

TIP IMPORTANTE

Antes de hacer 'push', asegurate que no existan cambios no sincronizados entre el directorio de la web y el directorio local, ya que en ese caso te 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 repo local.

Cómo subir tareas y actividades

Las actividades en cátedra las tendrás que subir en el directorio Actividades/ACXX, siendo XX el número que corresponde a la actividad. El enunciado no debe ser subido y asegúrate 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 **no será recolectada tu actividad**.

Para las actividades que se realicen en pareja, basta con que uno suba la actividad y si la llegasen a subir ambos, tiene que ser la misma actividad o si no el corrector decidirá cuál corregir.

Las tareas se tienen que subir en un formato parecido a las actividades, pero debe ser en la carpeta de 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 éste link <https://github.com/IIC2233-2016-1/syllabus/issues/143> .