

INTRODUCCIÓN A GIT

PRIMERA SESIÓN

*TOULOUSE LAUTREC
EDUCACIÓN CONTINUA*

**TOULOUSE
LAUTREC**

ALEXANDER VALDEZ PORTOCARRERO

CONTENIDO

Introducción a GIT

SESIÓN - 1

Introducción a GIT

- Instalación y configuración
- Creación de un repositorio local
- Comandos básicos

Introducción a GIT

SESIÓN - 2

Trabajo con Repositorios Remotos

- Repositorios remotos
- Comandos para sincronización
- GitHub
- GitLab

CONTENIDO

Introducción a GIT

SESIÓN - 3

Ramas y fusiones

- Fusiones y conflictos
- Ramas en Git
- Estrategias de fusión

Introducción a GIT

SESIÓN - 4

Flujos de Trabajo en Git

- Flujos de trabajo
- Gitflow
- GitHub Flow
- Ejemplos

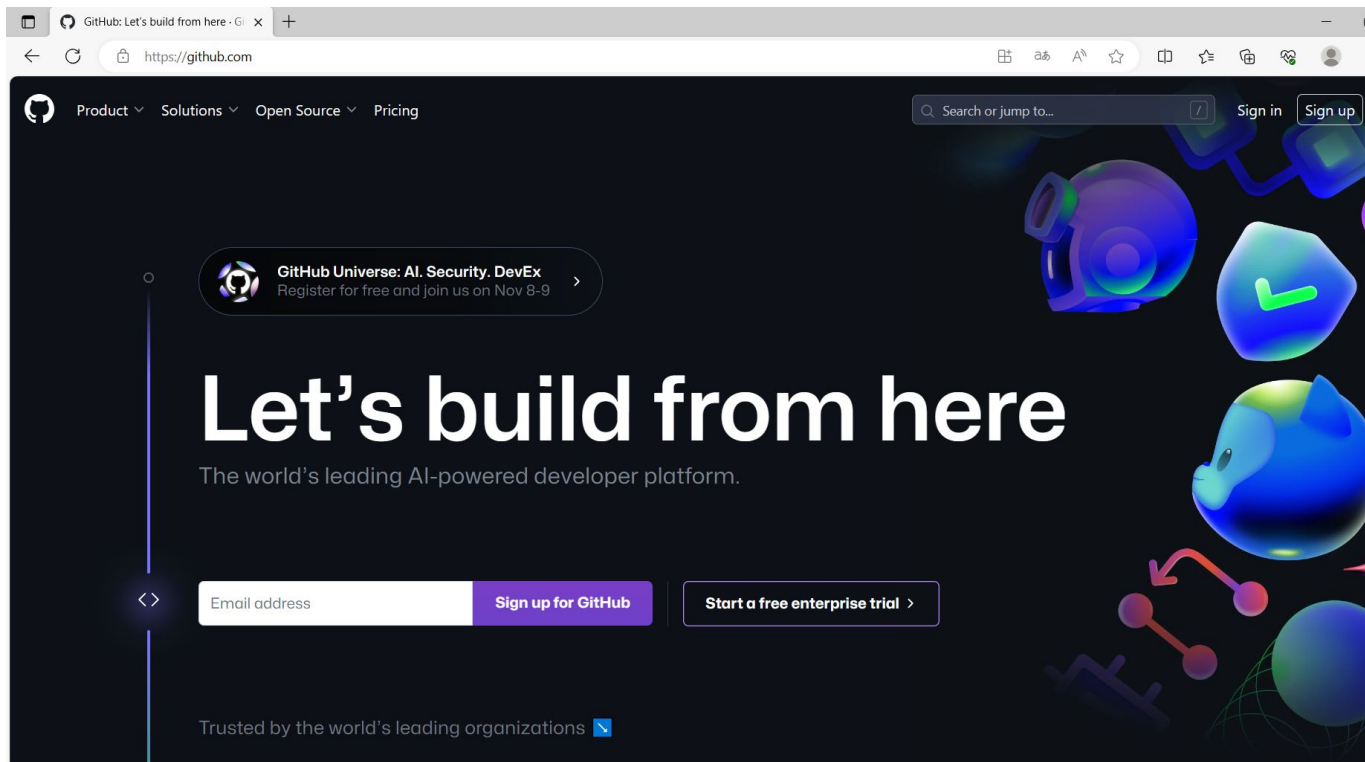
Normas de clase online:

- Habrá un **break de 10 min** después de la parte teórica y antes de la parte práctica en Google Collaboratory.
- La evaluación se realizará **durante la sesión 4 como trabajo final** y obtendrás puntos adicionales en función de las tareas resueltas y enviadas al correo del profesor:

TAREAS PRESENTADAS	PUNTOS ADICIONALES
Ninguna	0
1	+2
2	+4
3	+6

GITHUB

[GitHub: Let's build from here · GitHub](https://github.com)



GIT & GITHUB

GIT	GITHUB
<p>Git es un sistema de control de versiones que originalmente fue diseñado para operar en un entorno de Linux. Actualmente Git es multiplataforma, compatible con Linux, MacOS y Windows</p>	<p>GitHub es un servicio de almacenamiento que ofrece a los desarrolladores repositorios de software usando el sistema de control de versiones de Git</p>



GIT - INICIO

Mi nombre es Alexander
Valdez

Me gusta leer libros de
literatura y en mis tiempos
libre juego football.

Mi comida favorita es el ají
de gallina

V1

Mi nombre es Alexander
Valdez

Me gusta leer libros de
política y en mis tiempos
libre hago natación.

Mi comida favorita es la
jalea mixta.

V2

GIT - INICIO

Mi nombre es Alexander
Valdez

Me gusta leer libros de
literatura y en mis tiempos
libre juego football.

Mi comida favorita es el ají
de gallina

descripcion.txt

Mi nombre es Alexander
Valdez

Me gusta leer libros de
política y en mis tiempos
libre hago natación.

Mi comida favorita es la
jalea mixta.

descripcion2.txt

GIT - INICIO

Mi nombre es Alexander Valdez.

Me gusta leer libros de **literatura** y en mis tiempos libre **juego football**.

Mi comida favorita es **el ají de gallina**

descripcion.txt

Mi nombre es Alexander Valdez.

Me gusta leer libros de **política** y en mis tiempos libre **hago natación**.

Mi comida favorita es **la jalea mixta**.

descripcion2.txt

GIT

CONTROL DE VERSIONES



Mi nombre es Alexander Valdez.

Me gusta leer libros de **literatura** y en mis tiempos libre **juego football**.

Mi comida favorita es **el ají de gallina**

literatura

juego football

el aji de gallina

> **política**

> **hago natación**

> **jalea mixta**

Mi nombre es Alexander Valdez.

Me gusta leer libros de **política** y en mis tiempos libre **hago natación**.

Mi comida favorita es **la jalea mixta**.



Mi nombre es Alexander
Valdez.

Me gusta leer libros de
literatura y en mis tiempos
libre **juego football**.

Mi comida favorita es **el ají de
gallina**

descripcion.txt

Línea de comandos

```
$git init
```



Mi nombre es Alexander Valdez.

Me gusta leer libros de **literatura** y en mis tiempos libre **juego football**.

Mi comida favorita es **el ají de gallina**

descripcion.txt

Línea de comandos:

```
$git init
```

```
$ git add descripcion.txt
```



Mi nombre es Alexander Valdez.

Me gusta leer libros de **literatura** y en mis tiempos libre **juego football**.

Mi comida favorita es **el ají de gallina**

descripcion.txt

Línea de comandos:

```
$git init
```

```
$ git add descripcion.txt
```

```
$git commit -m "Mi primer commit"
```



Mi nombre es Alexander
Valdez.

Me gusta leer libros de **política**
y en mis tiempos libre **hago**
natación.

Mi comida favorita es **la jalea**
mixta.

descripcion.txt

Línea de comandos:

```
$git init  
$ git add descripcion.txt  
$git commit -m "Mi primer commit"  
  
$git add .
```



Mi nombre es Alexander Valdez.

Me gusta leer libros de política y en mis tiempos libre hago natación.

Mi comida favorita es la jalea mixta.

descripcion.txt

Línea de comandos:

```
$git init
$ git add descripcion.txt
$git commit -m "Mi primer commit"

$git add .
$git commit -m "cambios a primer commit"
```




Mi nombre es Alexander Valdez.

Me gusta leer libros de política y en mis tiempos libre hago natación.

Mi comida favorita es la jalea mixta.

descripcion.txt

Línea de comandos:

```
$git init
```

```
$ git add descripcion.txt
```

```
$git commit -m "Mi primer commit"
```

```
$git add .
```

```
$git commit -m "cambios a primer commit"
```

```
$git status
```



Mi nombre es Alexander Valdez.

Me gusta leer libros de política y en mis tiempos libre hago natación.

Mi comida favorita es la jalea mixta.

descripcion.txt

Línea de comandos:

```
$git init
```

```
$ git add descripcion.txt
```

```
$git commit -m "Mi primer commit"
```

```
$git add .
```

```
$git commit -m "cambios a primer commit"
```

```
$git status
```

```
$git show
```



Mi nombre es Alexander Valdez.

Me gusta leer libros de política y en mis tiempos libre hago natación.

Mi comida favorita es la jalea mixta.

descripcion.txt

Línea de comandos:

```
$git init
```

```
$ git add descripcion.txt
```

```
$git commit -m "Mi primer commit"
```

```
$git add .
```

```
$git commit -m "cambios a primer commit"
```

```
$git status
```

```
$git show
```

```
$git log descripcion.txt
```



Mi nombre es Alexander Valdez.

Me gusta leer libros de política y en mis tiempos libre hago natación.

Mi comida favorita es la jalea mixta.

descripcion.txt

Línea de comandos:

```
$git init  
$ git add descripcion.txt  
$git commit -m "Mi primer commit"  
  
$git add .  
$git commit -m "cambios a primer commit"  
$git status  
$git show  
$git log descripcion.txt  
$git push
```



RESUMEN



- Un **sistema de control de versiones como Git** nos ayuda a guardar el historial de cambios y crecimiento de los archivos de nuestro proyecto.
- Los cambios y diferencias entre las versiones de nuestros proyectos pueden tener similitudes, algunas veces los cambios pueden ser solo una palabra o una parte específica de un archivo específico. Git está optimizado para guardar todos estos cambios de forma atómica e incremental, o sea, aplicando cambios sobre los últimos cambios, estos sobre los cambios anteriores y así hasta el inicio de nuestro proyecto.



- El comando para iniciar nuestro repositorio, o sea, indicar a Git que queremos usar su sistema de control de versiones en nuestro proyecto, es **git init**.
- El comando para que nuestro repositorio sepa de la existencia de un archivo o sus últimos cambios es **git add**. Este comando no almacena las actualizaciones de forma definitiva, únicamente las guarda en algo que conocemos como “Staging Area” (área de montaje o ensayo).
- El comando para almacenar definitivamente todos los cambios que por ahora viven en el staging area es **git commit**. También podemos guardar un mensaje para recordar muy bien qué cambios hicimos en este commit con el argumento **-m "Mensaje del commit"**.
- Por último, si queremos mandar nuestros commits a un servidor remoto, un lugar donde todos podamos conectar nuestros proyectos, usamos el comando **git push**.



COMANDOS BÁSICOS DE GIT

- **git init:** inicializa un repositorio de GIT en la carpeta donde se ejecute el comando.
- **git add:** añade los archivos especificados al área de preparación (staging).
- **git commit -m "commit description":** confirma los archivos que se encuentran en el área de preparación y los agrega al repositorio.
- **git commit -am "commit description":** añade al staging area y hace un commit mediante un solo comando. (No funciona con archivos nuevos)
- **git status:** ofrece una descripción del estado de los archivos (untracked, ready to commit, nothing to commit).
- **git rm (. -r, filename) (--cached):** remueve los archivos del index.
- **git config --global user.email tu@email.com:** configura un email.
- **git config --global user.name :** configura un nombre.
- **git config --list:** lista las configuraciones.

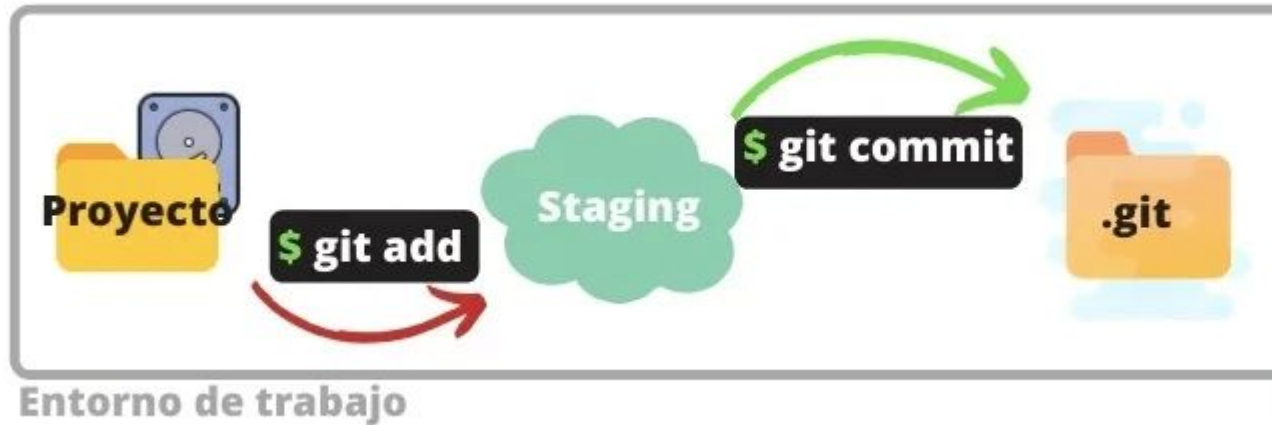


COMANDOS

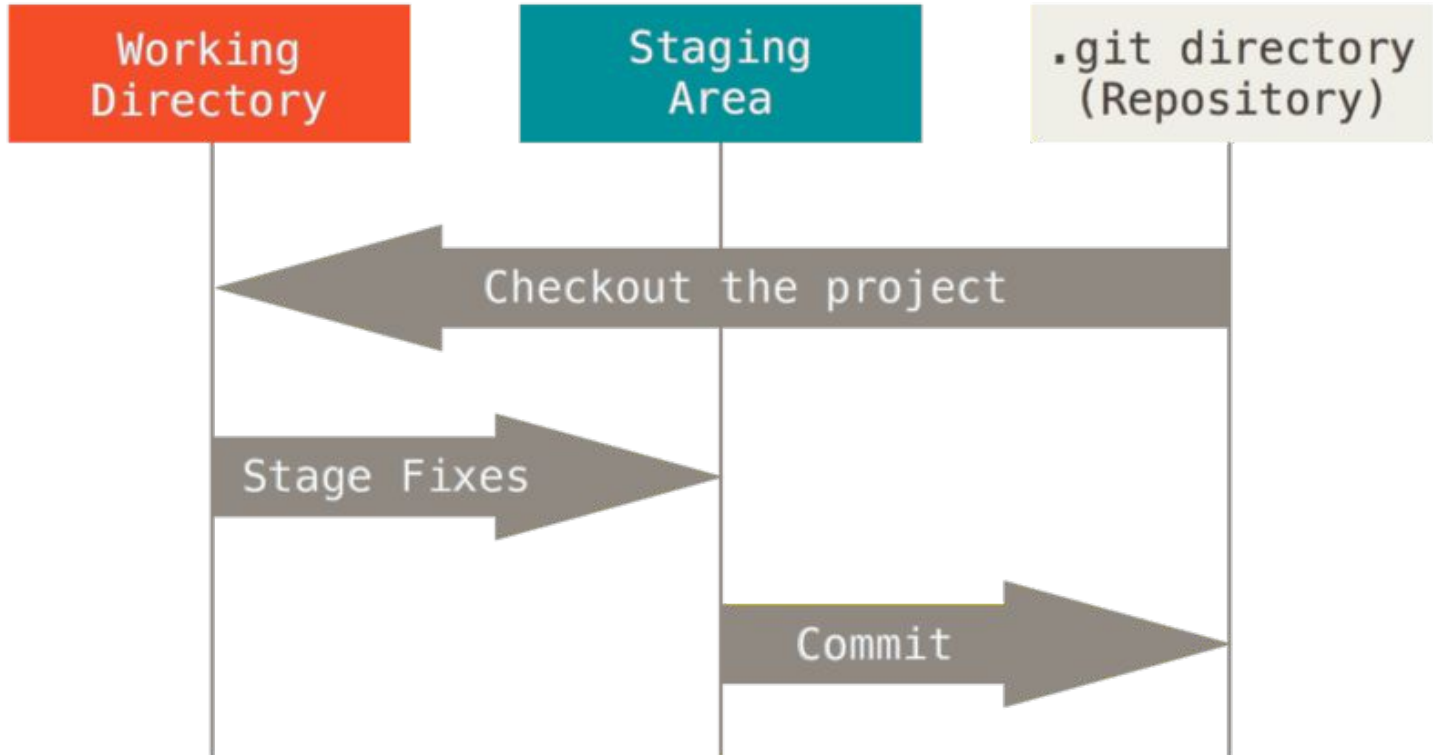
- **git log**: lista de manera descendente los commits realizados.
- **git log --stat**: además de listar los commits, muestra la cantidad de bytes añadidos y eliminados en cada uno de los archivos modificados.
- **git log --all --graph --decorate --oneline**: muestra de manera comprimida toda la historia del repositorio de manera gráfica y embellecida.
- **git show filename**: permite ver la historia de los cambios en un archivo.
- **git diff** : compara diferencias entre en cambios confirmados.

BREAK

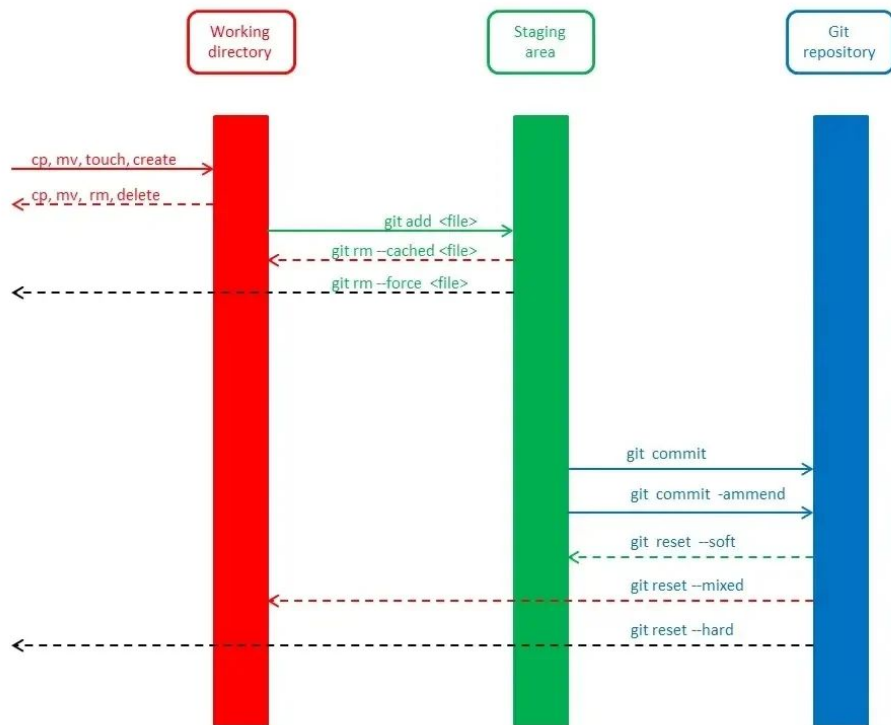
GIT - LOS TRES ESTADOS



GIT- LOS TRES ESTADOS



GIT- LOS TRES ESTADOS



GIT- CICLO DE VIDA DE LOS ARCHIVOS

- Archivos tracked.
- Archivos staged
- Archivos unstaged
- Archivos untracked

APRENDIZAJE AUTOMÁTICO CON PYTHON

Gracias por su atencion.