



RoboTech URJC

INTRODUCCIÓN A SHELL+ GIT

<https://robotech-urjc.github.io/mainweb/>





RoboTech URJC

¿Qué es la shell?

Es el intérprete de comandos de Linux

Permite ejecutar comandos para interactuar con el PC

Hay distintos tipos según cada sistema operativo, nosotros utilizamos BASH (default en Ubuntu)

Ctrl + Alt + t para abrir una nueva terminal.

A screenshot of a terminal window titled 'claudia-elez@claudia-elez-katana: ~'. The terminal shows two lines of text: 'claudia-elez@claudia-elez-katana:~\$ touch fichero' and 'claudia-elez@claudia-elez-katana:~\$'. A blue arrow points from the word 'Comando' below to the command 'touch fichero'. Another blue arrow points from the word 'Prompt' below to the prompt 'claudia-elez@claudia-elez-katana:~\$'.

Prompt

Comando



Uso básico de comandos

```
claudia-elez@claudia-elez-katana: ~/Desktop/taller_shell
claudia-elez@claudia-elez-katana:~/Desktop/taller_shell$ ls
prueba1.txt
claudia-elez@claudia-elez-katana:~/Desktop/taller_shell$ ls -last
total 8
4 drwxrwxr-x 2 claudia-elez claudia-elez 4096 Aug 24 13:10 .
0 -rw-rw-r-- 1 claudia-elez claudia-elez 0 Aug 24 13:08 prueba1.txt
4 drwxr-xr-x 4 claudia-elez claudia-elez 4096 Aug 24 13:08 ..
claudia-elez@claudia-elez-katana:~/Desktop/taller_shell$ touch prueba 2.txt
claudia-elez@claudia-elez-katana:~/Desktop/taller_shell$ ls prueba 2.txt
2.txt  prueba
claudia-elez@claudia-elez-katana:~/Desktop/taller_shell$ ls -last
total 8
4 drwxrwxr-x 2 claudia-elez claudia-elez 4096 Aug 24 13:10 .
0 -rw-rw-r-- 1 claudia-elez claudia-elez 0 Aug 24 13:10 2.txt
0 -rw-rw-r-- 1 claudia-elez claudia-elez 0 Aug 24 13:10 prueba
0 -rw-rw-r-- 1 claudia-elez claudia-elez 0 Aug 24 13:08 prueba1.txt
4 drwxr-xr-x 4 claudia-elez claudia-elez 4096 Aug 24 13:08 ..
claudia-elez@claudia-elez-katana:~/Desktop/taller_shell$
```

Comando

Flag(s) / Opciones

Argumentos
(Ojo con los espacios)

Salida del comando

username@pcname

Ruta de trabajo



El manual: comando man

```
claudia-elez@claudia-elez-katana: ~/Desktop/taller_shell
LS(1) User Commands LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default).
  Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
  fied.

  Mandatory arguments to long options are mandatory for short options
  too.

  -a, --all
        do not ignore entries starting with .

  -A, --almost-all
        do not list implied . and ..

  --author
```

Manual page ls(1) line 1 (press h for help or q to quit)

Para consultar el manual:
man sección asunto

Secciones de interés:
1 para comandos
2 para llamadas al sistema
3 para llamadas a biblioteca

Para buscar sobre una palabra:
apropos palabra



Comandos básicos

- **cd:** cambia el directorio actual
- **echo:** escribe los argumentos en la salida
- **cp:** copia ficheros
- **ls:** lista el contenido de un directorio
- **ls:** lista el contenido de un directorio
- **mv:** mueve ficheros
- **rm:** elimina ficheros
- **mkdir:** crea nuevos directorios
- **rmdir:** elimina directorios
- **touch:** cambia la fecha de modificación de un fichero (lo crea si no existe)
- **date:** muestra la fecha
- **who:** muestra los usuarios del sistema
- **whoami:** muestra tu nombre de usuario
- **grep:** busca cadenas dentro de ficheros
- **sort:** ordena las líneas de un fichero
- **wc:** cuenta caracteres, palabras y líneas de un fichero



Comandos básicos

➤ **diff:** compara ficheros

➤ **cat:** escribe en su salida el contenido de uno o varios ficheros

➤ **less:** permite leer un fichero de texto en la terminal usando scroll

➤ **file:** da “pistas” sobre el contenido de un fichero

➤ **head:** escribe las primeras filas de un fichero

➤ **tail:** escribe las últimas filas de un fichero

➤ **chmod:** modifica los permisos del fichero

➤ **sudo:** otorga permisos de administrador

➤ **top:** muestra los procesos y el estado del sistema

➤ **htop:** igual que top pero de forma interactiva

➤ **exit:** termina la ejecución de la shell

➤ **tar:** crea un fichero con múltiples dentro (comprimidos o no)



RoboTech URJC

Editores de texto

A screenshot of the nano text editor running in a terminal window. The window title is 'claudia-elez@claudia-elez-katana: ~/Desktop/taller_shell'. The editor shows a file named 'fichero *' with the text 'este es mi nuevo fichero...' on the first line. The bottom status bar displays various keyboard shortcuts: ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Location, ^X Exit, ^R Read File, ^\ Replace, ^U Paste, ^J Justify, and ^/ Go To Line.

```
claudia-elez@claudia-elez-katana: ~/Desktop/taller_shell
GNU nano 7.2 fichero *
este es mi nuevo fichero...

^G Help    ^O Write Out  ^W Where Is  ^K Cut      ^T Execute   ^C Location
^X Exit    ^R Read File  ^\ Replace   ^U Paste    ^J Justify   ^/ Go To Line
```

nano fichero

A screenshot of the gedit text editor window. The window title is 'fichero2 ~/Desktop/taller_shell'. The editor shows a file named 'fichero2' with a single line of text '1' on the first line. The bottom status bar displays 'Plain Text', 'Tab Width: 8', 'Ln 1, Col 1', and 'INS'.

```
fichero2
~/Desktop/taller_shell
1

Plain Text Tab Width: 8 Ln 1, Col 1 INS
```

gedit fichero



RoboTech URJC

¿Qué es git ?

Es un sistema de control de versiones especializado en código

Es un protocolo que necesita comunicarse con una plataforma: GitHub y/o GitLab

Nosotros vamos a utilizar GitHub, pero en otras asignaturas puede que utiliceis GitLab, ambas son de gran utilidad





RoboTech URJC

Creación cuenta en GitHub



Accedemos a github.com



Sign up



Introducimos nuestros datos

Welcome to GitHub!
Let's begin the adventure

Enter your email*

✗ EMAIL@EMAIL.COM

Create a password*

✓ CONTRASEÑA1234++++

Enter a username*

✓ USERNAME12345678-DOTCO

Email preferences

☐ Receive occasional product updates and announcements.

Continue



RoboTech URJC

Creamos o copiamos nuestras claves SSH

Las claves ssh son “llaves” que utilizaremos para autenticarnos y poder utilizar git sin necesidad de contraseñas

Hay una pública y una privada, NUNCA compartas la privada. Para utilizar git usaremos la pública.

Puede que ya las tengas creadas, si no es así se pueden generar

A terminal window titled 'claudia-elez@claudia-elez-katana: ~' with search, menu, and window control icons in the title bar. The terminal shows two commands: 'cat ~/.ssh/id_ed25519.pub' followed by 'ssh-' on the next line, and 'ssh-keygen -t rsa' on the next line. A blue arrow points from the text 'Con este comando se generan nuevas claves.' to the 'ssh-keygen -t rsa' command.

```
claudia-elez@claudia-elez-katana: ~  
claudia-elez@claudia-elez-katana:~$ cat ~/.ssh/id_ed25519.pub  
ssh-  
claudia-elez@claudia-elez-katana:~$ ssh-keygen -t rsa
```

Con este comando se generan nuevas claves.



RoboTech URJC

Vinculamos nuestra clave a git

Click en el perfil → Settings → SSH and GPG keys → Add new SSH key

Le damos el nombre deseado (intentad que sea reconocible puesto que necesitareis vincular vuestro PC y el del lab como mínimo)

Add new SSH Key

Title

Key type

Authentication Key ↕

Key

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'

Add SSH key



RoboTech URJC

Comprobamos y configuramos...

```
claudia-elez@claudia-elez-katana: ~  
claudia-elez@claudia-elez-katana:~$ ssh -T git@github.com  
Hi celezm! You've successfully authenticated, but GitHub does not provide shell access.  
claudia-elez@claudia-elez-katana:~$ git config --global user.name "celezm"  
claudia-elez@claudia-elez-katana:~$ git config --global user.email "tu email de la uni"
```



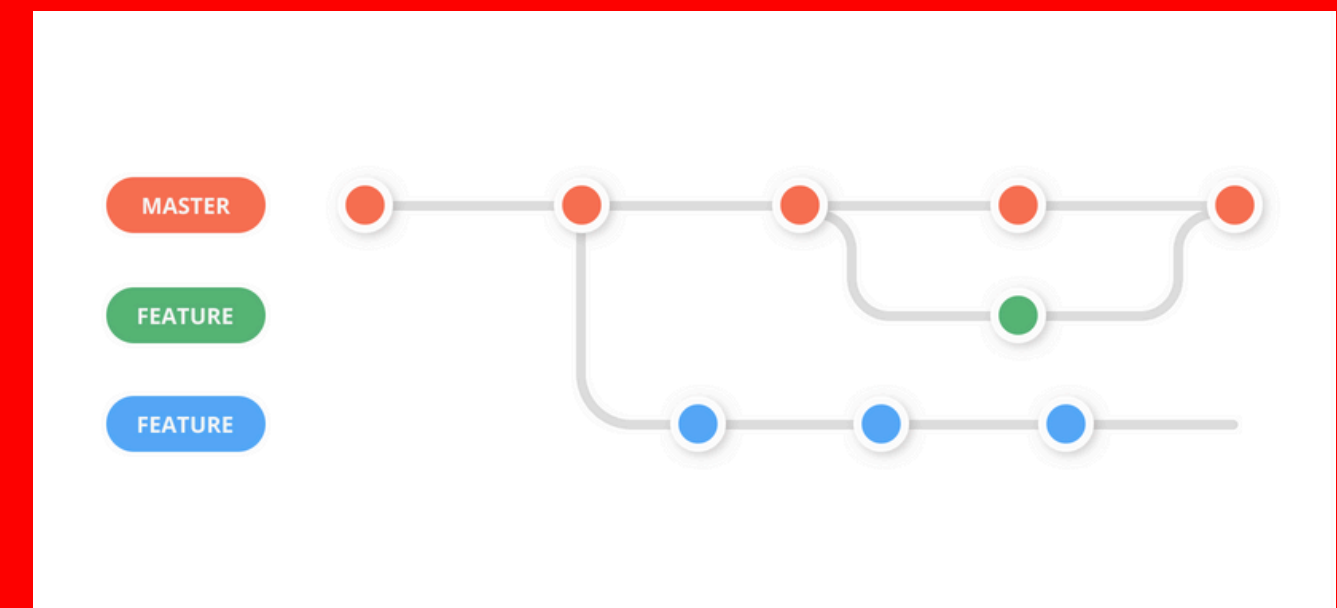
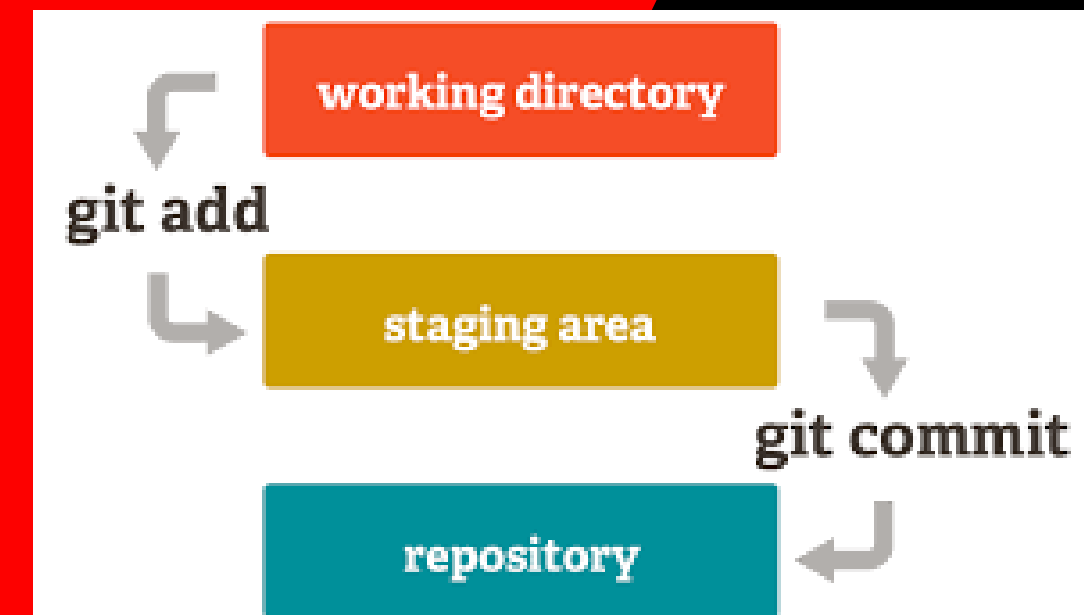
RoboTech URJC

Conceptos clave en Git

Repositorio/Repo: directorio que contiene los archivos del proyecto. Local y/o remoto.

Commit: Guarda el estado del proyecto en ese momento y lo puedes volver a él en cualquier momento.

Ramas/Branches: Son líneas de trabajo en las que puedes trabajar sin afectar a la principal, la rama main/master por defecto.





RoboTech URJC

Comandos básicos Git

- **git add [fichero(s) / directorio(s)]** : agrega los cambios
- **git commit -m "version x.x"**: registra los cambios como una versión
- **git push**: sube los cambios a la plataforma
- **git pull**: "trae" los cambios más actualizados
- **git clone**: clona un repo remoto a uno local
- **git log**: muestra el historial de commits
- **git -h / --help** : manual / ayuda
- **git branch [nombre]**: crea una rama
- **git checkout / switch [rama]**: cambia la rama del repo local
- **git checkout / restore [commit hash]**: cambia al estado de ese commit
- **git merge [rama]**: Fusiona la rama actual con la introducida



RoboTech URJC

Fin de la **Explicación** ¡Turno del escape room!

Conócenos un poco más ...



asociacion.robotech@urjc.es



robotech-urjc.github.io/mainweb



[@robotech_urjc](https://www.instagram.com/robotech_urjc)



[@robotech_urjc](https://twitter.com/robotech_urjc)

