

Sustitución de comandos

El sistema operativo GNU/Linux permite realizar sustitución de comandos. Lo explicaremos con un ejemplo.

El comando **whoami** devuelve el nombre del usuario actual. Pero supongamos que queremos poner un cartel que diga “Yo soy fulano” (suponiendo que “fulano” sea nuestro nombre de usuario).

Una manera es tipear los comandos: **echo "Yo soy "; whoami** Pero esto nos devolvería la salida:

```
Yo soy
fulano
```

Puesto que se trata de dos comandos y no de uno solo.

Con sustitución de comandos, esto se realizaría así: **echo "Yo soy \$(whoami) "**

El comando anterior se interpreta del siguiente modo: primero, se ejecuta el comando entre \$(...).

Así, se sustituye esta expresión por la salida del comando:

```
echo "Yo soy $(whoami)" → echo "Yo soy fulano"
```

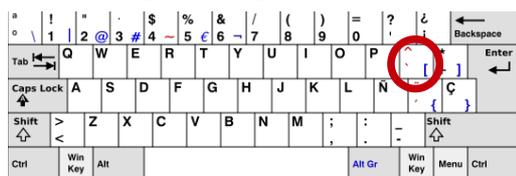
Luego, se ejecuta el comando resultante, mostrándose la salida: **Yo soy fulano**

La sintaxis alternativa para \$(...) es encerrar la expresión a sustituir entre *tildes graves* (como las que se usan en francés). El equivalente al ejemplo anterior sería:

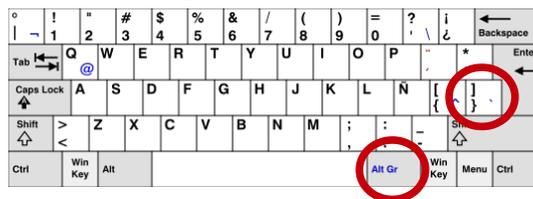
```
echo "Yo soy `whoami` "
```

La tilde grave está en distintos lugares del teclado, dependiendo de la distribución del mismo:

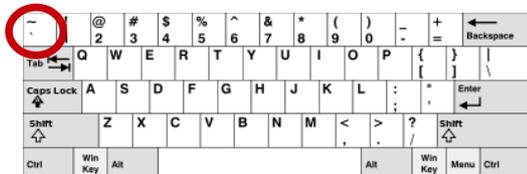
Teclado español-España:



Teclado español-Latinoamérica:



Teclado EEUU-Internacional:



Código ASCII: 96

Los comandos sustituidos pueden anidarse, es decir incluirse uno dentro de otro, sin límite. Ej:

```
juan@PC ~ $ whoami
```

```
juan
```

```
juan@PC ~ $ who
```

```
juan tty1 2016-04-19 09:21
```

```
ana tty8 2016-04-19 08:08 (:0)
```

```
luis pts/2 2016-04-19 09:51 (:0.0)
```

```
juan@PC ~ $ who | grep $(whoami) → who | grep juan
```

```
juan tty1 2016-04-19 09:21
```

```
juan@PC ~ $ echo "Mis datos de conexión son: $(who | grep $(whoami))"
```

```
→ echo "Mis datos de conexión son: $(who | grep juan)"
```

```
→ echo "Mis datos de conexión son: juan tty1 2016-04-19 09:21"
```

```
Mis datos de conexión son: juan tty1 2016-04-19 09:21
```

Si se utilizan las tildes graves, debemos “escaparlas” con \ (barra invertida) en cualquier nivel de anidación excepto el más externo. El ejemplo anterior sería:

```
juan@PC ~ $ echo "Mis datos de conexión son: `who | grep `whoami``"
```