

```
$ gitclick
```

Usage:

```
gitclick create [<repository>] [on <account>] [--no-issues] [--no-wiki] [--private]
```

Creates a public repository with issues and wiki by default

<repository> default: the name of the current folder

<account> default: the default account

```
gitclick list                               List existing accounts
```

```
gitclick use <account>                    Set <account> as the default
```

```
gitclick add                               Interactive prompt for creating a new account
```

```
gitclick remove <account>                 Remove <account>
```

```
gitclick -v, --version                     Output version number
```

```
gitclick -h, --help                       Output usage information
```

```
$ █
```

Activitat pràctica 3 GIT



git

Contenidos

1	Act practica GIT:	3
1.1	Ejercicio 1:	3
1.2	Ejercicio 2:	3
1.3	Ejercicio 3:	4
1.4	Ejercicio 4:	5

1. Act practica GIT:

Creación y actualización de repositorios.

Ejercicios de deshacer cambios Para hacer estos ejercicios es necesario haber hecho antes los ejercicios sobre historial de cambios.

1.1. Ejercicio 1:

1. Eliminar la última línea del fichero indice.txt y guardarlo.

```
1 - nano indice.txt
2 - Eliminar la ultima linea.
```

2. Comprobar el estado del repositorio.

```
1 - git status
```

3. Deshacer los cambios realizados en el fichero indice.txt para volver a la versión anterior del fichero.

```
1 -git checkout indice.txt
```

4. Volver a comprobar el estado del repositorio.

```
1 - git status
```

1.2. Ejercicio 2:

1. Eliminar la última línea del fichero indice.txt y guardarlo.

```
1 - nano indice.txt
2 - Eliminar la ultima linea.
3 - git status
```

2. Añadir los cambios a la zona de intercambio temporal.

```
1 - git add .
```

3. Comprobar de nuevo el estado del repositorio.

```
1 - git status
```

4. Quitar los cambios de la zona de intercambio temporal, pero mantenerlos en el directorio de trabajo.

```
1 - git restore --staged index.txt
2 - git reset index.txt
```

5. Comprobar de nuevo el estado del repositorio.

```
1 - git status
```

6. Deshacer los cambios realizados en el fichero indice.txt para volver a la versión anterior del fichero.

```
1 - git checkout --indice.txt
```

7. Volver a comprobar el estado del repositorio.

```
1 - git status
```

1.3. Ejercicio 3:

1. Eliminar la última línea del fichero indice.txt y guardarlo.

```
1 - nano indice.txt
2 - eliminar la ultima linea
3 - git status
```

2. Eliminar el fichero capitulos/capitulo3.txt.

```
1 - rm capitulos/capitulo3.txt
```

3. Añadir un fichero nuevo capitulos/capitulo4.txt vacío.

```
1 - nano capitulos/capitulo4.txt
2 - git status
```

4. Añadir los cambios a la zona de intercambio temporal.

```
1 - git add .
```

5. Comprobar de nuevo el estado del repositorio.

```
1 - git status
```

6. Quitar los cambios de la zona de intercambio temporal, pero mantenerlos en el directorio de trabajo.

```
1 - git reset .
```

7. Comprobar de nuevo el estado del repositorio.

```
1 - git status
```

8. Deshacer los cambios realizados para volver a la versión del repositorio.

```
1 - git checkout -- .
```

9. Volver a comprobar el estado del repositorio.

```
1 - git status
```

1.4. Ejercicio 4:

1. Eliminar la última línea del fichero indice.txt y guardarlo.

```
1 - nano indice.txt
```

2. Eliminar el fichero capitulos/capitulo3.txt.

```
1 - rm capitulos/capitulo3.txt
```

3. Añadir los cambios a la zona de intercambio temporal y hacer un commit con el mensaje “Borrado accidental.”

```
1 - git commit -m "Borrado accidental"  
2 - git log --oneline
```

4. Comprobar el historial del repositorio.

```
1 - git status
```

5. Deshacer el último commit pero mantener los cambios anteriores en el directorio de trabajo y la zona de intercambio temporal.

```
1 - git reset --soft HEAD-1  
2 - git log --oneline
```

6. Comprobar el historial y el estado del repositorio.

```
1 - git status
```

7. Volver a hacer el commit con el mismo mensaje de antes.

```
1 - git commit -m "Borrado accidental"  
2 - git log --oneline
```

8. Deshacer el último commit y los cambios anteriores del directorio de trabajo volviendo a la versión anterior del repositorio.

```
1 - git reset --hard HEAD~1  
2 - git log --oneline
```

9. Comprobar de nuevo el historial y el estado del repositorio.

```
1 - git status
```