



ENTORNOS DE DESARROLLO

UD4.2 "Mastering Git Commit!"

Contenidos

- ◆ ¿Qué es `git commit`?
- ◆ Mastering Git Commit!

¿Qué es `git commit`?

`git commit` es un comando de Git que se utiliza para guardar cambios en el repositorio local.

Cuando realizas modificaciones en los archivos de tu proyecto y deseas guardar esos cambios en el sistema de control de versiones, utilizas el comando `git commit`.

Mastering Git Commit!

1. Basic Commit

```
git commit -m "Initial commit"
```

Confirmar cambios con un mensaje conciso. El indicador `-m` sirve para agregar un mensaje de confirmación directamente desde la línea de comando.

2. Adding Changes

```
git commit -a -m "Add new feature"
```

El indicador `-a` prepara y confirma automáticamente los archivos rastreados, guardando un paso en el proceso.

Mastering Git Commit!

3. Amending the last commit

```
git commit --amend -m "Improved feature"
```

Modifica la última confirmación modificando los cambios. Útil para agregar archivos olvidados o corregir el mensaje de confirmación.

4. Interactive Staging

```
git add -i
```

```
git commit -m "Interactive commit"
```

Inicia la preparación interactiva, lo que le permite preparar cambios de forma selectiva antes de confirmarlos.

Mastering Git Commit!

5. Committing Specific Files

```
git commit file1.txt file2.txt -m "Commit specific files"
```

Confirma solo archivos específicos, lo que resulta útil cuando desea agrupar cambios de forma selectiva.

6. Committing with Date

```
git commit --date="YYYY-MM-DD HH:MM:SS" -m "Commit with custom date"
```

Establece una fecha y hora de confirmación personalizadas.

Mastering Git Commit!

7. Committing All Changes with Date

```
git commit -a --date="YYYY-MM-DD HH:MM:SS" -m "Commit all with custom date"
```

Confirma todos los cambios con una fecha y hora personalizadas.

8. Signing Commits

```
git commit -S -m "Signed commit"
```

Agrega una firma GPG a la confirmación para mejorar la seguridad.

Mastering Git Commit!

9. Committing with Author Information

```
git commit --author="John Doe <john@example.com>" -m "Commit with custom author"
```

Establece información de autor personalizada para la confirmación.

10. Committing Changes to a Different Branch

```
git commit -m "Commit on feature branch" feature-branch
```

Confirma directamente los cambios en una rama específica.

Mastering Git Commit!

11. Committing and Tagging

```
git commit -m "Release v1.0" && git tag v1.0
```

Confirma los cambios y etiqueta la confirmación para el control de versiones.

12. Committing Changes from Another Branch

```
git commit -m "Merge feature branch" feature-branch
```

Confirma los cambios de una rama fusionada a la rama actual.

Mastering Git Commit!

13. Committing Changes with Custom Message File

```
git commit -F commit_message.txt
```

Especifica un archivo que contiene el mensaje de confirmación.

14. Committing Changes with Custom Editor

```
git commit -e
```

Abre el editor predeterminado para escribir el mensaje de confirmación.

Mastering Git Commit!

15. Committing Changes without a Newline at the End

```
git commit --no-edit --allow-empty-message
```

Permite confirmar sin una nueva línea al final del mensaje.

16. Committing Changes with Emoji

```
git commit -m ":rocket: Add new feature"
```

Agrega emoji para confirmar mensajes para darle un toque creativo.

Mastering Git Commit!

17. Committing Changes with Multiple Messages

```
git commit -m "First line" -m "Second line" -m "Third line"
```

Agrega varias líneas al mensaje de confirmación.

18. Committing Changes with Jira Issue Reference

```
git commit -m "PROJ-123: Implement new feature"
```

Incluye una referencia al problema de Jira en el mensaje de confirmación.

Mastering Git Commit!

19. Committing Changes with Co-author

```
git commit -m "Fix issue" -m "Co-authored-by: Rakesh<rakesh@example.com>"
```

Acredita a los coautores en el mensaje de confirmación.

20. Committing Changes in Dry Run Mode

```
git commit --dry-run -m "Test commit"
```

Comprueba lo que se confirmaría sin realizar cambios. Útil para repasar.

Bibliografía

- https://twitter.com/devops_tech/status/1728291500671529345