

# Git & GitHub



*Aplicado al Business Intelligence*



**Vicent Correcher**

🗨️ Top Data Engineering Voice

# ¿Porqué un taller de



y



?

**Proyecto final  
universidad**

**Proceso de datos  
en empresa**

# ¿Que es Control de Versiones?

GitProy

- Trabajo.r
- Trabajo\_1.r
- Trabajo\_2.r
- Trabajo\_def.r
- Trabajo\_defdef.r
- Trabajo\_definitivo.r

# ¿Que es Git?

Software de control de  
versiones



Estándar



# ¿Para qué un SCV?

## AHORA...

- **Gestión de ficheros a lo largo del tiempo:** Evolución del trabajo de clase
- **Gestión del versionado de los ficheros:** Si un archivo se corrompe o hemos cometido un fallo volvemos atrás
- **Mecanismo para compartir ficheros con otros compañeros**

# ¿Para qué un SCV?

## EN EL FUTURO...

- **Crear copias de seguridad y restaurarlas**
- **Sincronizar (mantener al día) a los desarrolladores respecto a la última versión de desarrollo**
- **Deshacer cambios:** Tanto problemas puntuales, como problemas introducidos hace tiempo
- **Gestionar la auditoría del código**
- **Realizar pruebas (aisladas) :** Simples o utilizando el mecanismo de branches/merges

# ¿Que es GitHub?

The GitHub logo, consisting of the word "GitHub" in a white, bold, sans-serif font, centered within a solid black rectangular box.

**GitHub**

Una plataforma de alojamiento de código que utiliza el sistema de control de versiones Git

Y...

**GitHub** Pages

Websites for you and your projects.

Hosted directly from your [GitHub repository](#). Just edit, push, and your changes are live.



# Y...

## GitHub CodeSpaces + GitHub Copilot

**Secure development  
made simple**

GitHub Codespaces gets you up and coding faster with fully configured, secure cloud development environments native to GitHub.

[Get started for free >](#)

[View plans and pricing >](#)

# Y...

## GitHub Actions

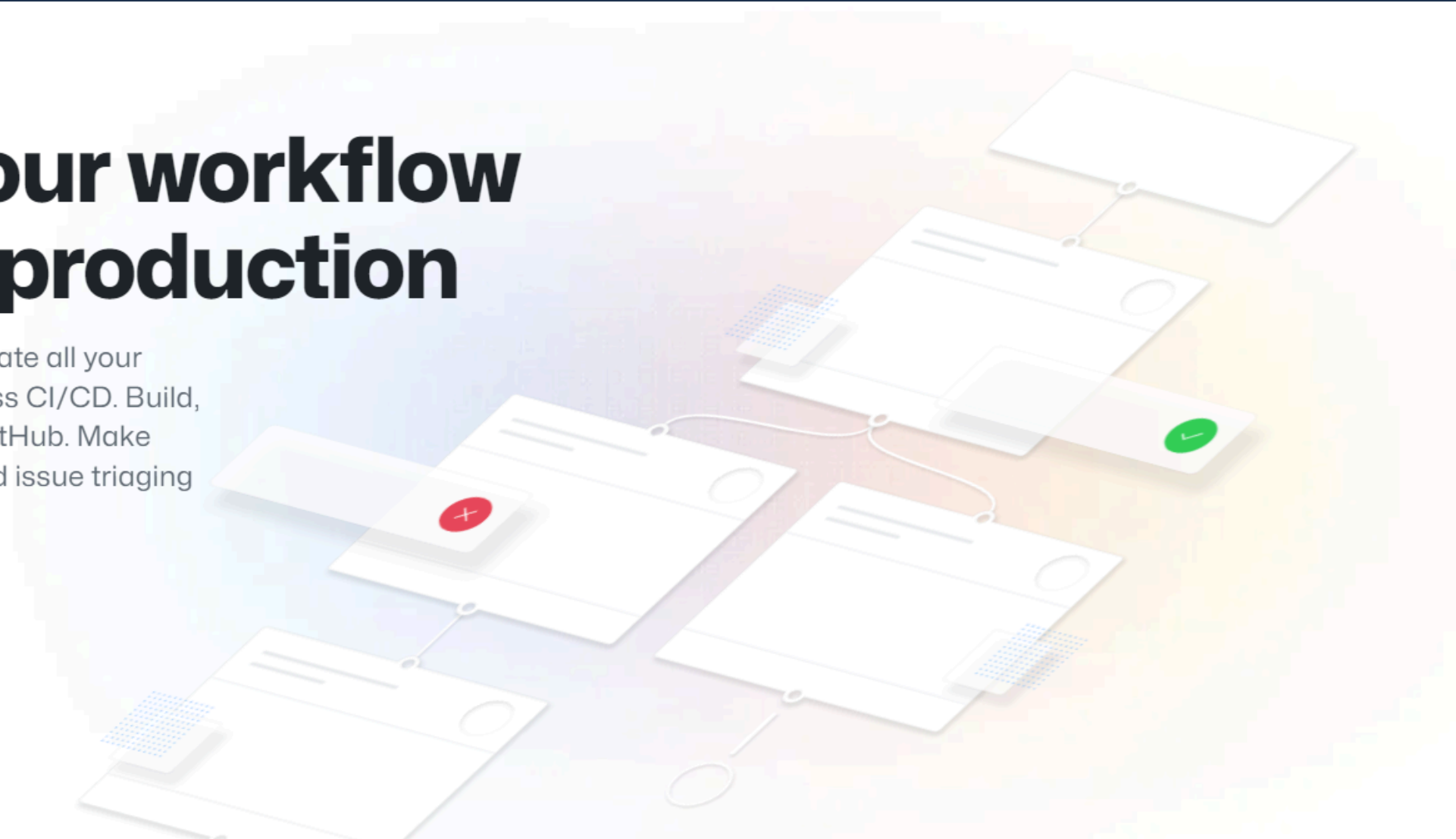


### Automate your workflow from idea to production

GitHub Actions makes it easy to automate all your software workflows, now with world-class CI/CD. Build, test, and deploy your code right from GitHub. Make code reviews, branch management, and issue triaging work the way you want.

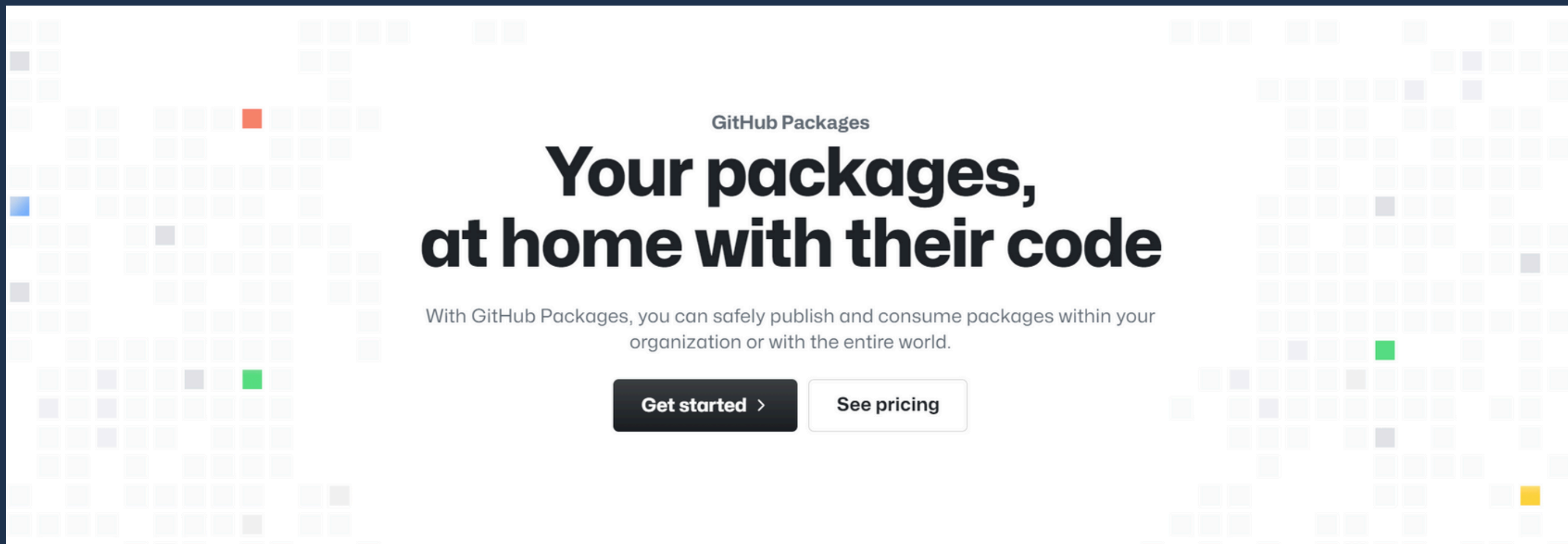
[Get started with Actions >](#)

Questions? [Contact Sales >](#)



# Y...

## GitHub Packages

A mockup of the GitHub Packages landing page. The background is white with a faint, pixelated world map. The text is centered and reads: "GitHub Packages", "Your packages, at home with their code", and "With GitHub Packages, you can safely publish and consume packages within your organization or with the entire world." Below the text are two buttons: "Get started >" and "See pricing".

GitHub Packages

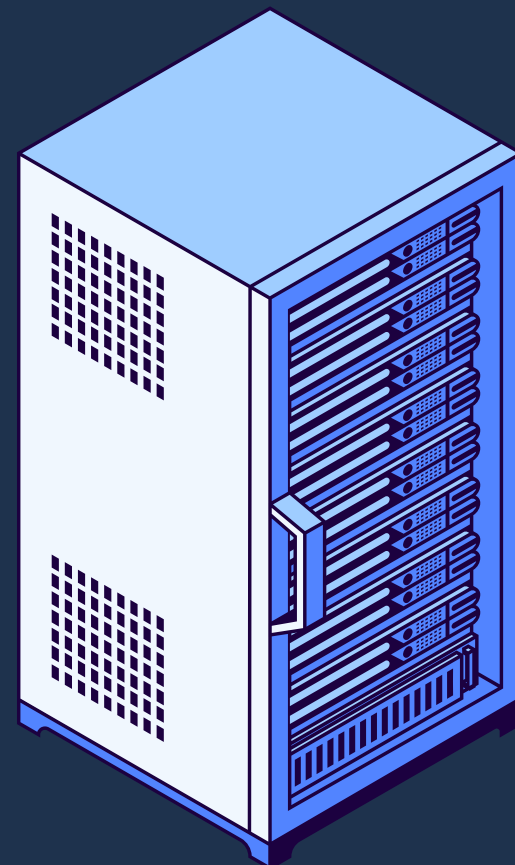
### Your packages, at home with their code

With GitHub Packages, you can safely publish and consume packages within your organization or with the entire world.

[Get started >](#) [See pricing](#)

# Conceptos básicos

- Servidor



# Conceptos básicos

- Repositorio

Local

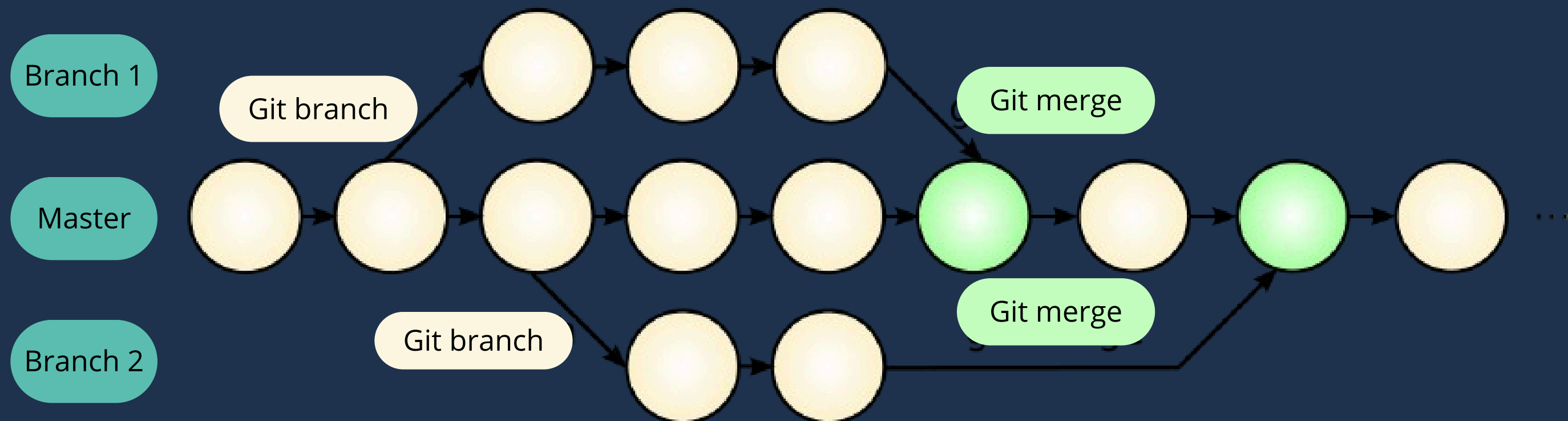


Online



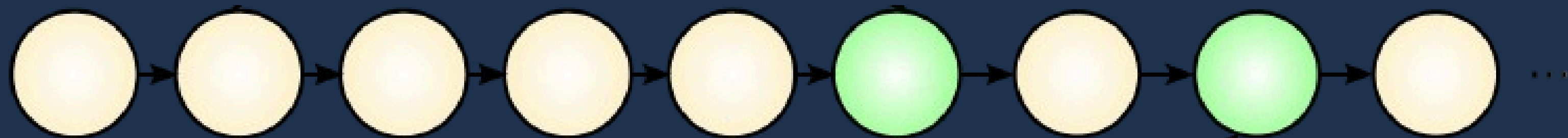
# Conceptos básicos

- Rama



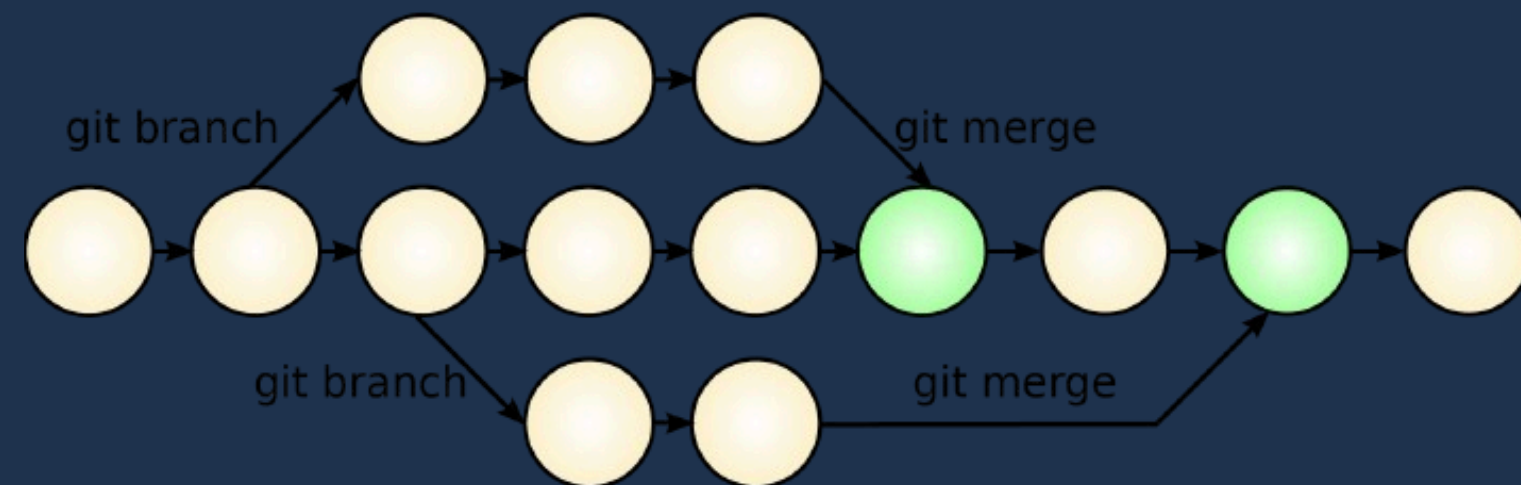
# Conceptos básicos

- Trunk/main/master (rama principal)



# Conceptos básicos

- Working copy / working set



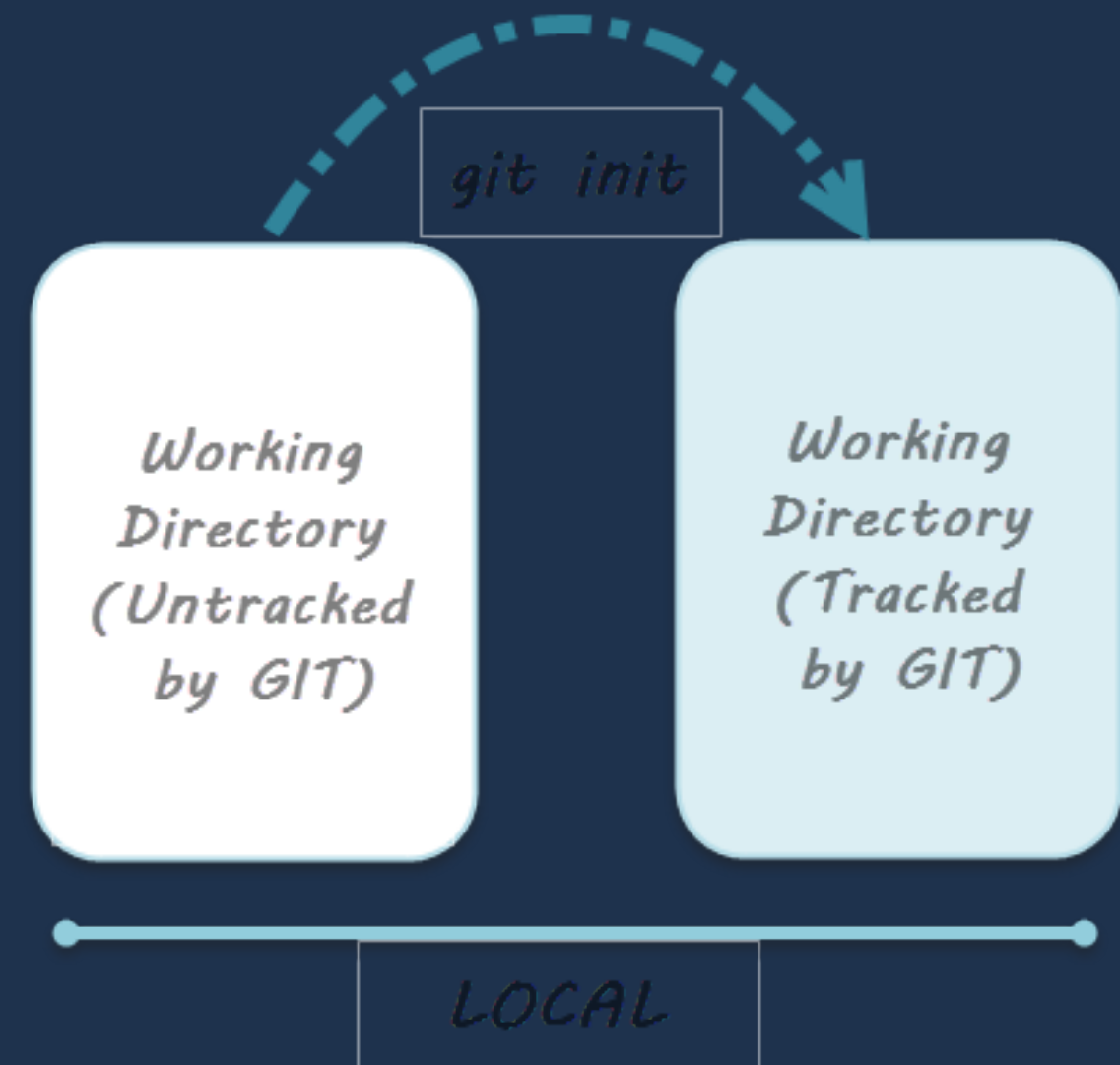


# Comandos básicos

## Git Init

Se usa para crear un nuevo repositorio de Git.

Se ejecuta en el directorio donde quieres inicializar el repositor



# Comandos básicos

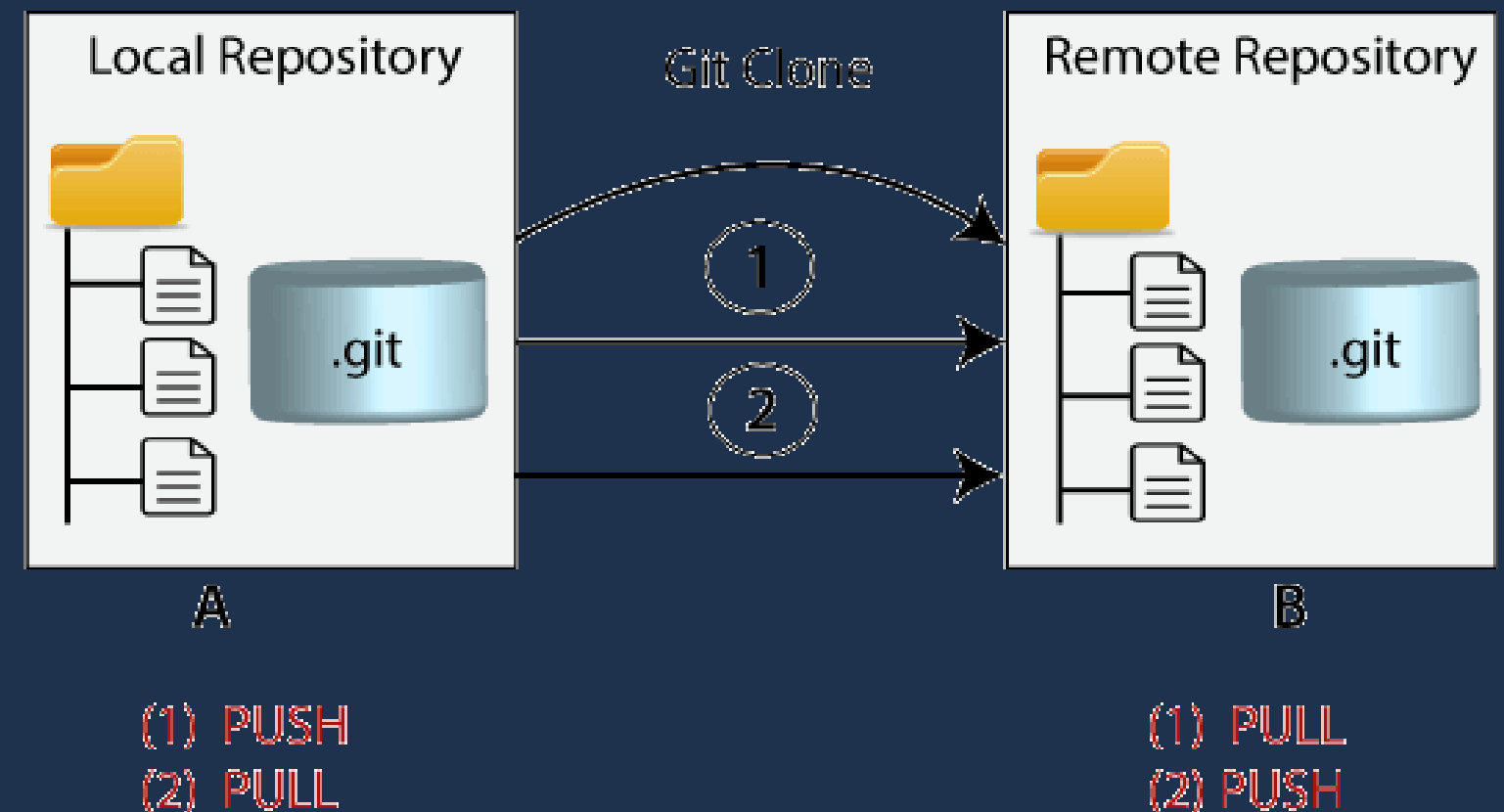
## Git Clone

**Clona un repositorio existente desde una URL.**

**Se usa para obtener una copia de un repositorio remoto.**

**Ejemplo:** git clone

<https://github.com/usuario/repositorio>.  
git

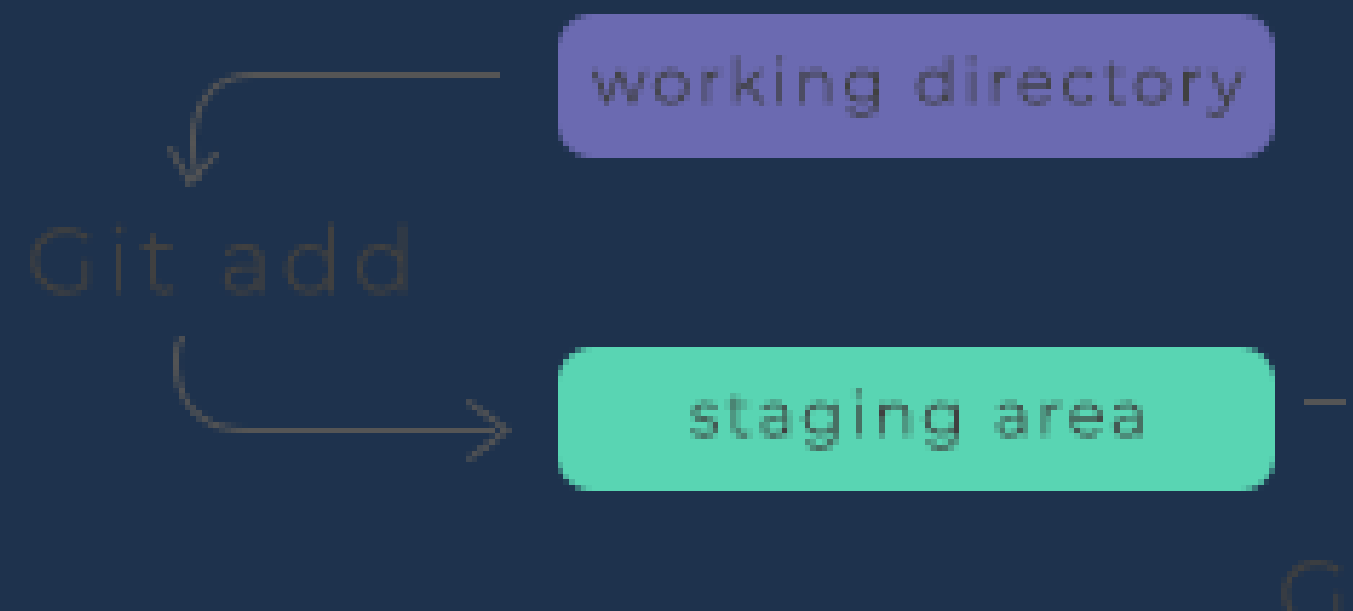


# Comandos básicos

## Git Add

Se usa para añadir cambios al área de preparación (staging area) antes de hacer un commit.

**Ejemplo:** `git add archivo.txt`  
o  
`git add .`

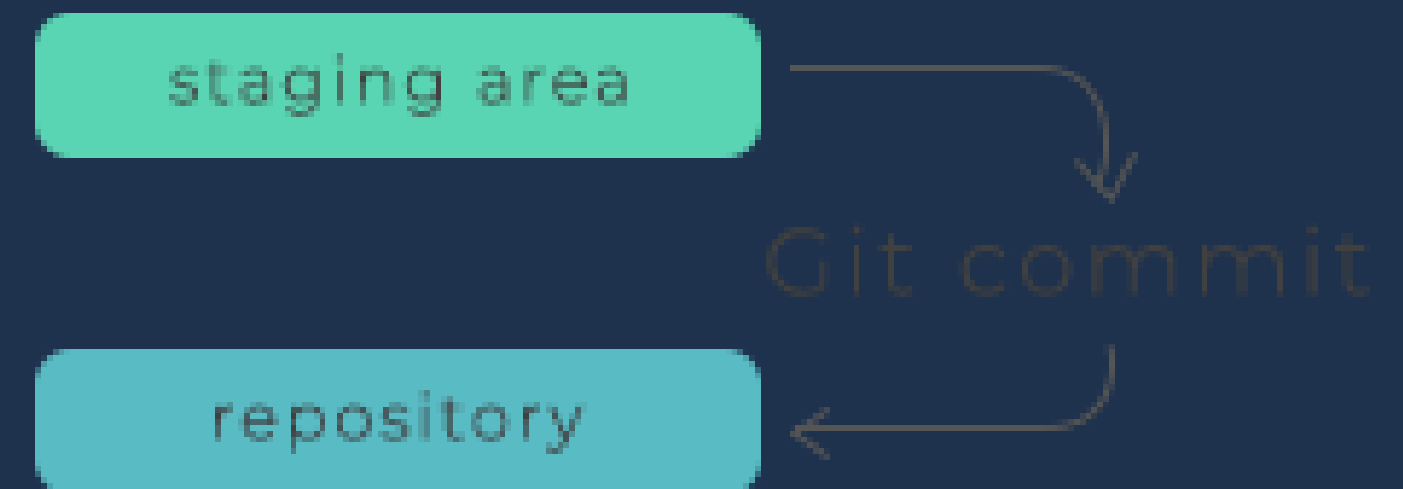


# Comandos básicos

## Git Commit

**Crea un snapshot de los cambios en el repositorio.**

**Ejemplo:** `git commit -m "Mensaje del commit"`



# Comandos básicos

## Git Status

Muestra el estado actual del repositorio, incluyendo archivos que están en staging, no rastreados o modificados.

```
@Harish MINGW64 /e/ToolsQA/First Project (master)
$ git add ABC.txt

@Harish MINGW64 /e/ToolsQA/First Project (master)
$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   ABC.txt

@Harish MINGW64 /e/ToolsQA/First Project (master)
$ |
```

# Comandos básicos

## Git Push

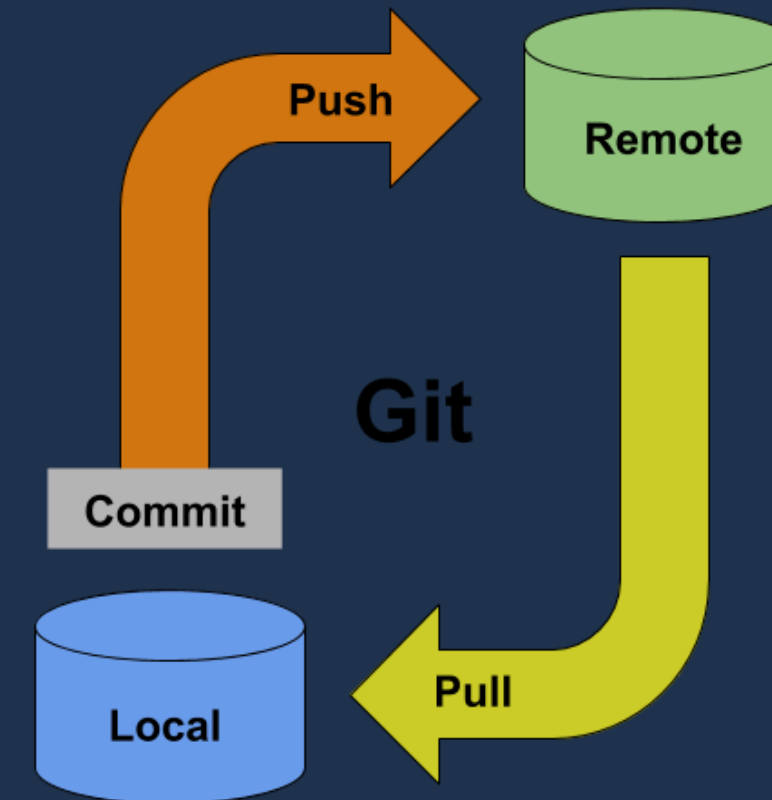
**Empuja los commits locales al repositorio remoto.**

**Ejemplo:** `git push origin main`

## Git Pull

**Descarga y fusiona cambios desde un repositorio remoto.**

**Ejemplo:** `git pull origin main`



# Comandos básicos

## Git Branch

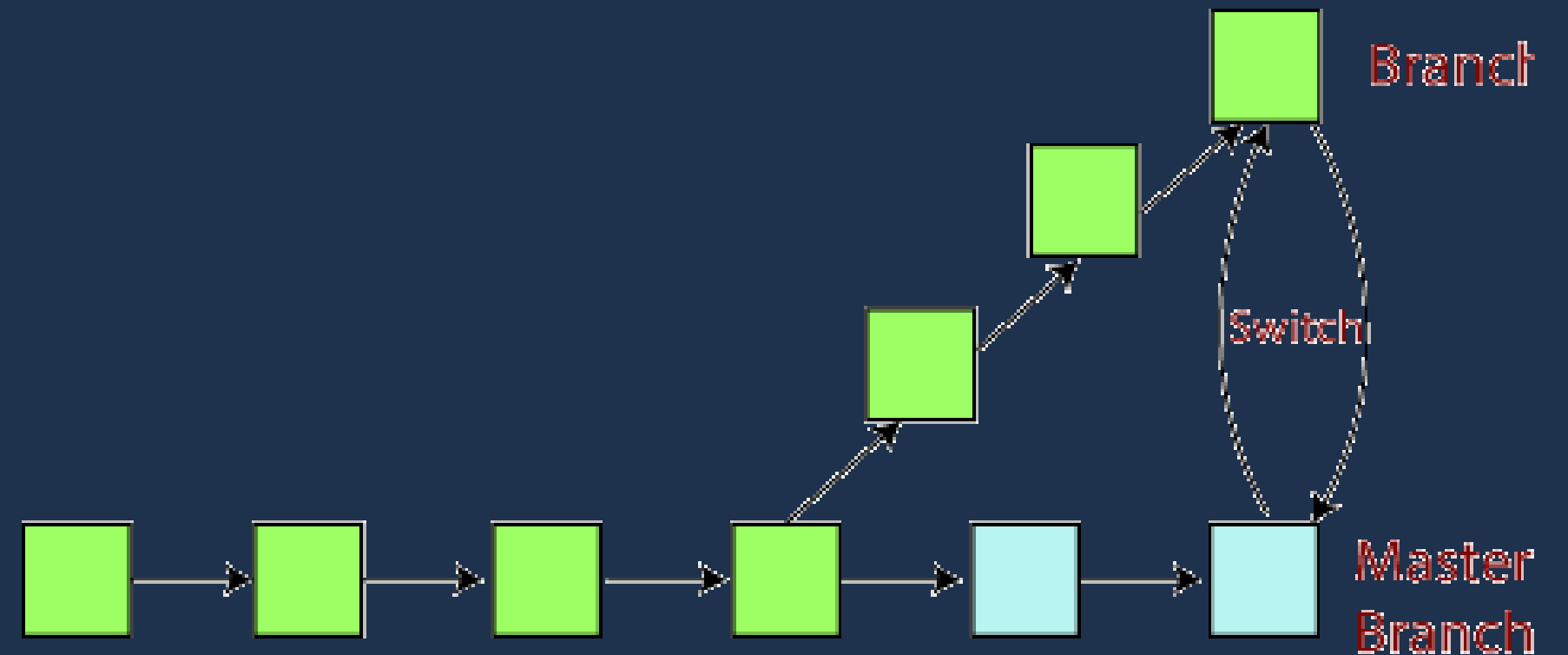
Se usa para listar, crear o eliminar ramas.

Ejemplo: `git branch nueva-rama`

## Git Checkout

Se usa para cambiar de rama.

Ejemplo: `git checkout nueva-rama`

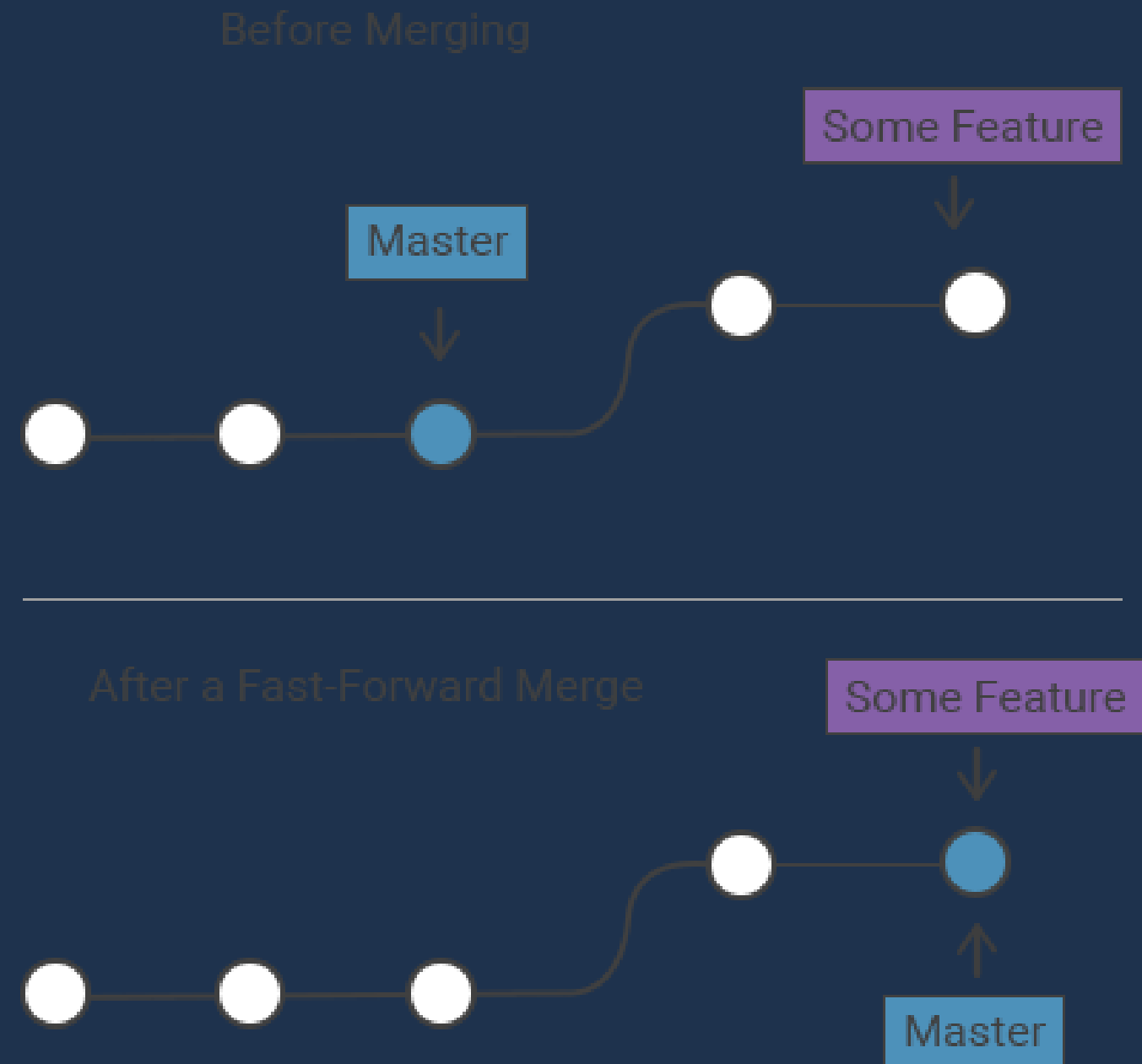


# Comandos básicos

## Git Merge

Combina cambios de una rama en otra.

**Ejemplo:** git merge rama-dos

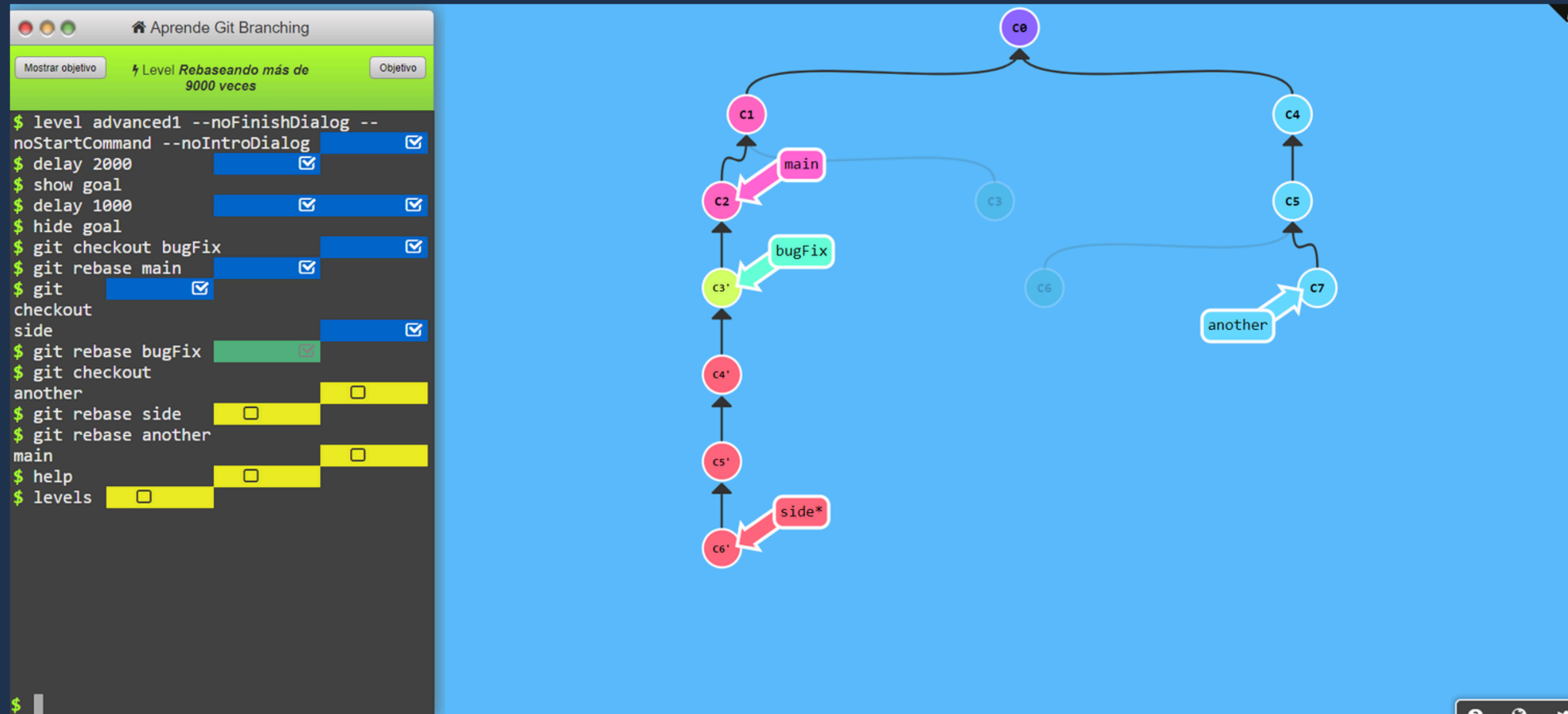




# Otros comandos útiles

- **git log:** Ver el historial de commits.
- **git revert:** Revertir un commit.
- **git reset:** Resetear un commit
- **git stash:** Guardar cambios temporalmente.
- **git rebase:** Reorganizar commits.
- **git diff:** Muestra diferencias entre versiones, ramas...

# Juguemos...



# Caso 1 - Trabajo individual para analizar el PIB

Estás trabajando en un proyecto en análisis exploratorio de datos para analizar la evolución del PIB (Producto Interno Bruto) de varios países.

## Tareas:

- Inicializas el repositorio con el fichero en el que vas a realizar tu trabajo: Nombre\_trabajo.r
- En él te guardas los datos sobre los que vas a trabajar
- Haces la recopilación, limpieza, transformación de datos...
- Haces la parte de visualización de datos con ggplot2

Horas antes de la entrega se te ocurre probar a hacer la visualización con plotly y commiteas cuando... Oh oh



No te has guardado un backup y tu código acaba de dejar de funcionar.

Te das cuenta que para poder graficar con plotly necesitas cambiar el formato de tus datos y no te da tiempo a reescribir todo el código.

# Caso 2 - Trabajo en grupo para predicciones temporales en bolsa

Os mandan el proyecto final  
en Predicción de series  
temporales para predecir el  
precio de las acciones de  
Apple



## Tareas:

- Inicializas el repositorio remoto en GitHub
- Tu compañero y tú decidís dividiros el trabajo en:
  - Extracción diaria de los precios medios diarios
  - Modelos predictivos y visualización
- Cada uno os creáis vuestra rama y empezáis el desarrollo en local
- Tras finalizar el proyecto llega el momento crítico, quedan escasos días para la entrega y debéis unir ambas ramas y pasarlas a producción en la rama master en GitHub.

# Caso 3 - Gestión de un modelo de IA en producción con múltiples equipos

Imagina un departamento de datos en una empresa del sector automovilístico.

Tienen un modelo de machine learning en producción que predice averías (detección de anomalías).

El modelo es constantemente mejorado y actualizado, pero es utilizado por otros equipos para realizar experimentos y personalizaciones.



**data** → raw, processed  
**models** → trained\_models, experiments

**src** → data\_preprocessing.py, model\_training.py, model\_monitoring.py

**notebooks** → exploratory\_data\_analysis.ipynb, model\_evaluation.ipynb

**Dockerfile**  
**requirements.txt**

- **data:** datos crudos y transformados.
- **models:** Custodia los modelos entrenados y los resultados de los experimentos.
- **src:** Conserva el código fuente de las diversas fases del pipeline.
- **notebooks:** Reúne los cuadernos de Jupyter para el análisis exploratorio y la evaluación de modelos.
- **Dockerfile:** Esboza la imagen de Docker para asegurar la reproducibilidad del entorno.
- **requirements.txt:** Enumera las dependencias del proyecto.

# Flujo de trabajo con Git y GitHub:

## Seguimiento de los cambios:

C:\Proyectos\IA>

```
git add .  
git commit -m "Inicialización del proyecto"
```



## Creación de ramas:

- **main:** Rama principal para la versión estable del modelo.
- **develop:** Rama para el desarrollo activo.
- **feature:** Ramas para nuevas funcionalidades.
- **bugfix:** Ramas para corregir errores.

C:\Proyectos\IA>

```
git branch develop  
git checkout develop
```



## Desarrollo diario:

Cada desarrollador trabaja en su propia rama.  
Al finalizar las tareas, se crean pull requests hacia la rama develop.

C:\Project\IA>

```
git checkout -b feature  
git add .  
git commit -m "Nueva característica implementada"  
git push origin feature
```



# Flujo de trabajo con Git y GitHub:

## Reentrenamiento:

Se crea una nueva rama a partir de develop.  
Se reentrena el modelo con los nuevos datos.  
Se evalúa el modelo y se comparan los resultados con el modelo anterior.

```
C:\Proyectos\IA> git checkout -b retrain_model
git add .
git commit -m "Modelo reentrenado con datos de la semana X"
git push origin retrain_model
```

## Puesta en producción:

Una vez aprobados los cambios, se fusionan en la rama main.  
Se despliega el modelo en producción.

```
C:\Proyectos\IA> git checkout main
git merge develop
git push origin main
```


## Monitorización continua:

Se utiliza una herramienta de monitorización para rastrear el rendimiento del modelo en producción.  
Los resultados de la monitorización se almacenan en el repositorio.

```
C:\Proyectos\IA> Bash
git add monitoring_results.csv
git commit -m "Resultados de monitorización de la semana X"
```



# ¿Creamos nuestro primer portfolio?



**Vicent Muñoz Correcher**  
vicentcorrecher

BI & AI | DataOps

Edit profile

2 followers • 2 following

Volkswagen Group  
Barcelona  
in/vicentmc

vicentcorrecher / README.md

```
print("Hello World ") ...soy Vicent Muñoz! 🙌
```

BI & AI | DataOps

¡Bienvenido a mi portfolio de GitHub! Aquí encontrarás un resumen de mi trayectoria, proyectos destacados (o no porque soy un desastre con estas cosas... 😅) y cómo ponerte en contacto conmigo.

[LinkedIn](#)

**Tech Stack:**

python, R, AWS, azure, GoogleCloud, Apache Spark, Apache Airflow, MongoDB, mysql, Neo4j, sqlite, Teradata, cassandra, Keras, TensorFlow, PyTorch, mlflow, scikit-learn, power bi

**Acerca de mí**

Graduado en Business Intelligence and Analytics y estudiante del máster de Inteligencia Artificial en la Universidad de Valencia. Actualmente cubro una posición de analista de datos en Volkswagen Group Services.

**Proyectos Profesionales en curso...**

Data Analyst Specialist for Industrial Automation en Volkswagen Group

**VOLKSWAGEN**  
GROUP SERVICES

- Barcelona y alrededores
- Estrategia y gobernanza del dato - DataOps
- Elaboración de dashboards - Power BI, Tableau
- Creación, administración y gestión de bases de datos - Azure Data Services
- Planes de automatización ETL - Azure Data Factory, Power Automate, Polars
- Orquestación ETL - Apache AirFlow
- Creación de aplicaciones de negocio - Power Apps

Event Manager en Economía DataTech

**DataTech**

- Valencia/València, Comunidad Valenciana, España
- Dirección, organización y gestión de eventos relacionados con datos, IA y nuevas tecnologías en la Universitat de València



# Herramientas útiles para crear nuestro portfolio

