

Estos ejercicios se desarrollarán tras haber hecho el primer push, por tanto ya debes tener establecida correctamente la conexión entre tu repositorio local y remoto.

1. **Verifica el estado de tu proyecto**
2. **Es posible que aún tengamos pendientes archivos por añadir al staging y hacer el commit. Haz esos dos pasos.**
3. **Sube los cambios a GitHub (push)**
4. **Crea y activa una nueva rama para probar mejoras en el proyecto con este comando: `git checkout -b "nombre_rama"`**
5. **Ahora que estás trabajando en la nueva rama, en el script `app_abandono.py` incorpora el siguiente fragmento de código:**

```
df['segmento_valor'] = pd.qcut(df['valor'], 3, labels=['bajo', 'medio', 'alto'])
```
6. **Añade el cambio al Staging y haz un commit comentando los cambios**
7. **Sube la nueva rama a remoto con los cambios realizados**
8. **Revisa tu repositorio, tendrás una indicación de solicitud de fusión “Compare & pull request” que debes resolver**
 - a) Accede y revisa los cambios, si estás de acuerdo puedes integrarlos al main con “Pull request”
 - b) Realiza un “Merge pull request” (confirma el merge), a partir de ese momento ambas ramas tendrán el mismo contenido.
9. **Edita directamente en el repositorio el `README.md`, añadiendo la siguiente línea:**
`## Proyecto piloto en Data Science donde se crea una App para prevenir el abandono de clientes`

10. Vuelve a la terminal, revisa en que rama estás actualmente, si no estás en la rama la principal, debes volver a ella con este comando:
`git checkout main`
11. Ahora tienes que traer el cambio que hiciste en el repositorio remoto al local, para ello ejecuta el comando: `git pull origin main`
12. Vuelve a GitHub y crea desde allí una nueva rama, esta se llamará “mejoras_pipeline” (simularemos que la ha creado otro compañero del equipo)

 - a) En la rama recién creada añade un nuevo archivo, se llamará `pipeline_test.py` y contendrá la siguiente línea de código:
`print("Esta es una prueba de pipeline")`
13. Nadie del equipo te ha mencionado aún que está esa rama en el repositorio. Usa el comando: `git Branch -a` ¿Qué aparece?
14. No aparece la rama de la que hablan en la oficina, comprueba ahora con este comando: `git fetch origin` ¿Qué aparece?
15. Ya tienes la rama localizada, ahora haz una copia en local para trabajar con ella con el comando:
`git checkout -b mejoras_pipeline origin/mejoras_pipeline`