

# DOCUMENTACIÓN DEL PROYECTO

---

## Preprocesamiento de Datos - Cultura Digital y Sociedad

Nombre: Yoryhi Isaac Cadena Acosta

Unidad 2 - Tema 1

---

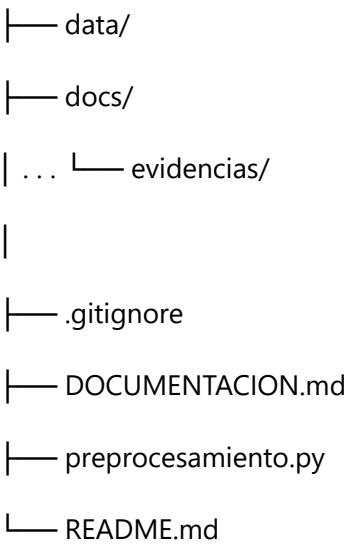
### 1. Descripción del proyecto

Este proyecto busca crear una serie de pasos simples para preparar datos antes de analizarlos, usando el lenguaje de programación Python y la herramienta llamada . Además, se usará Git y GitHub para guardar los cambios del proyecto de forma ordenada y segura.

---

### 2. Estructura del repositorio

preprocesamiento-cienciadatos/



### 3. Funciones implementadas y comandos utilizados

Funciones:

- `cargar_datos(ruta_archivo)` → Carga un dataset CSV.
- `limpiar_datos(df)` → Elimina valores nulos y duplicados.
- `normalizar_datos(df, columnas)` → Normaliza columnas numéricas entre 0 y 1.
- `exportar_datos(df, ruta_salida)` → Exporta los datos procesados a un nuevo CSV.

Comandos:

Comando	¿Qué hace?
<code>git config --global user.name "Tu Nombre"</code>	Guarda nuestro nombre o el que pusimos en Git

Comando	¿Qué hace?
git config --global user.email "yoryhicadena@gmail.com"	Guarda el correo asociado al repositorio
git init	Inicia un nuevo repositorio de Git en la carpeta
git clone URL	Descarga o clona un repositorio de GitHub al computador
git status	Muestra que archivos han cambiado o estan listos para subir
git add .	Prepara todos los archivos nuevos o modificados para el commit
git commit -m "mensaje"	Guarda una versión "commit" del proyecto con un mensaje
git branch	Muestra todas las ramas existentes
git checkout -b nombre-rama	Crea y cambia a una nueva rama
git merge nombre-rama	Une los cambios de otra rama con la rama principal que es main
git push origin main	Sube los cambios de la rama principal al repositorio de GitHub
git pull	Descarga los últimos cambios desde GitHub

## 4. Proceso de trabajo con Git y GitHub

1. Crear el proyecto en GitHub y hacer la configuración inicial, con los archivos importantes: (.gitignore y README.md)
2. Se crea la rama **feature/preprocesamiento**, donde se escribira el código para preparar los datos
3. Se agregan funciones poco a poco y se guarda cada cambio con una nota que explique qué se hizo es decir "commit"
4. Unir el nuevo trabajo "merge" de la rama **feature/preprocesamiento** en **main**
5. Enviar todos los cambios al proyecto en GitHub, para que queden guardados

## 5. Evidencias del proceso

1. Configuración del nombre y correo en Git

```
PS C:\1. UNACH\1.-Issac-Unach-1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git config --local --list
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
user.name=Isaac
user.email=yoryhicadena@gmail.com
remote.origin.url=https://github.com/Issac0805/preprocesamiento-cienciadatos.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
```

## 2. Creación del repositorio y archivos iniciales (.gitignore, README.md)

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git add .
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git commit -m "Commit inicial: agregar .gitignore y README"
[main c540c79] Commit inicial: agregar .gitignore y README
3 files changed, 0 insertions(+), 0 deletions(-)
3 files changed, 0 insertions(+), 0 deletions(-)
3 files changed, 0 insertions(+), 0 deletions(-)
3 files changed, 0 insertions(+), 0 deletions(-)
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 src/preprocesamiento.py
3 files changed, 0 insertions(+), 0 deletions(-)
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 src/preprocesamiento.py
```

## 3. Commit inicial y push a GitHub

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 20 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 1.02 KiB | 1.02 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Issac0805/preprocesamiento-cienciadatos.git
54df053..c540c79 main -> main
branch 'main' set up to track 'origin/main'.
```

## 4. Creación de la rama feature/preprocesamiento

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git checkout -b feature/preprocesamiento
Switched to a new branch 'feature/preprocesamiento'
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git branch
* feature/preprocesamiento
main
```

## 5. Commit con función preprocesamiento completo de datos

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git add preprocesamiento.py
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git commit -m "Agregar función para preprocesamiento completo de datos"
[feature/preprocesamiento dd56aa1] Agregar función para preprocesamiento completo de datos
1 file changed, 21 insertions(+)
create mode 100644 preprocesamiento.py
```

## 6. Commit con función limpiar\_datos()

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git add .
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git commit -m "Agregar función limpiar_datos() y actualizar estructura del proyecto"
[feature/preprocesamiento e178b30] Agregar función limpiar_datos() y actualizar estructura del proyecto
2 files changed, 0 insertions(+), 0 deletions(-)
rename src/preprocesamiento.py => DOCUMENTACION.md (100%)
```

## 7. Commit con función normalizar\_datos()

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git add preprocesamiento.py
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git commit -m "Agregar función normalizar_datos()"
[feature/preprocesamiento 65d919c] Agregar función normalizar_datos()
1 file changed, 24 insertions(+), 4 deletions(-)
```

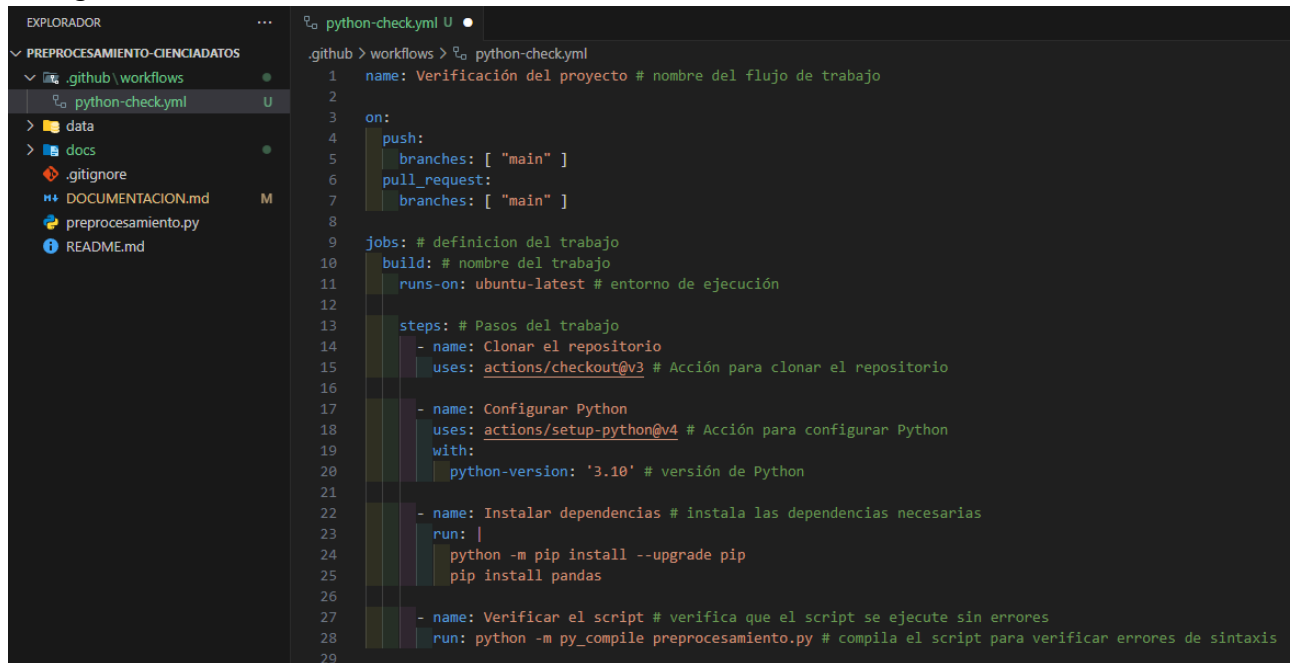
## 8. Commit con función exportar\_datos()

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git add preprocesamiento.py
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git commit -m "Agregar función exportar_datos()"
[feature/preprocesamiento 2ca40fe] Agregar función exportar_datos()
1 file changed, 10 insertions(+)
```

## 9. Fusión y push final hacia main

```
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git merge feature/preprocesamiento
Updating c540c79..2ca40fe
Fast-forward
 src/preprocesamiento.py => DOCUMENTACION.md | 0
 README.md | Bin 956 -> 956 bytes
 preprocesamiento.py | 51 ++++++
3 files changed, 51 insertions(+)
rename src/preprocesamiento.py => DOCUMENTACION.md (100%)
create mode 100644 preprocesamiento.py
PS C:\1. UNACH\1.-Issac-Unach-\1. AUTONOMOS UNACH\Semestre 3\Cultura_Digital_y_Sociedad\preprocesamiento-cienciadatos> git push origin main
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 20 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (13/13), 1.97 KiB | 1009.00 KiB/s, done.
Total 13 (delta 5), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (5/5), completed with 1 local object.
To https://github.com/Issac0805/preprocesamiento-cienciadatos.git
c540c79..2ca40fe main -> main
```

## 10. Configuración del workflow de GitHub Actions



```

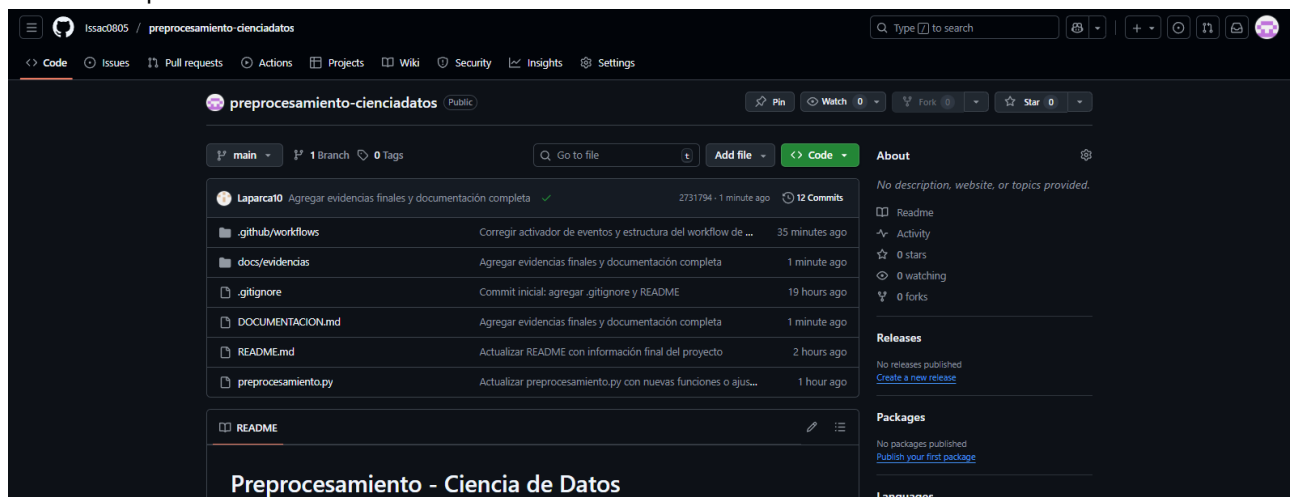
EXPLORADOR
PREPROCESAMIENTO-CIENCIADATOS
  .github\workflows
    python-check.yml
  data
  docs
  .gitignore
  DOCUMENTACION.md
  preprocesamiento.py
  README.md

.python-check.yml
.github > workflows > python-check.yml
1  name: Verificación del proyecto # nombre del flujo de trabajo
2
3  on:
4    push:
5      branches: [ "main" ]
6    pull_request:
7      branches: [ "main" ]
8
9  jobs: # definicion del trabajo
10    build: # nombre del trabajo
11      runs-on: ubuntu-latest # entorno de ejecución
12
13      steps: # Pasos del trabajo
14        - name: Clonar el repositorio
15          uses: actions/checkout@v3 # Acción para clonar el repositorio
16
17        - name: Configurar Python
18          uses: actions/setup-python@v4 # Acción para configurar Python
19          with:
20            python-version: '3.10' # versión de Python
21
22        - name: Instalar dependencias # instala las dependencias necesarias
23          run: |
24            python -m pip install --upgrade pip
25            pip install pandas
26
27        - name: Verificar el script # verifica que el script se ejecute sin errores
28          run: python -m py_compile preprocesamiento.py # compila el script para verificar errores de sintaxis
29

```

on: indica cuándo se ejecuta es decir cada que se hace push a main y jobs: define lo que hará el flujo, clona el repositorio, instala Python y pandas, luego verifica que el script preprocesamiento.py se ejecute sin errores de sintaxis.

## 11. Vista del repositorio en GitHub con todos los archivos subidos



The screenshot shows the GitHub repository page for 'preprocesamiento-cienciadatos'. The repository is public and has 12 commits. The commit history is as follows:

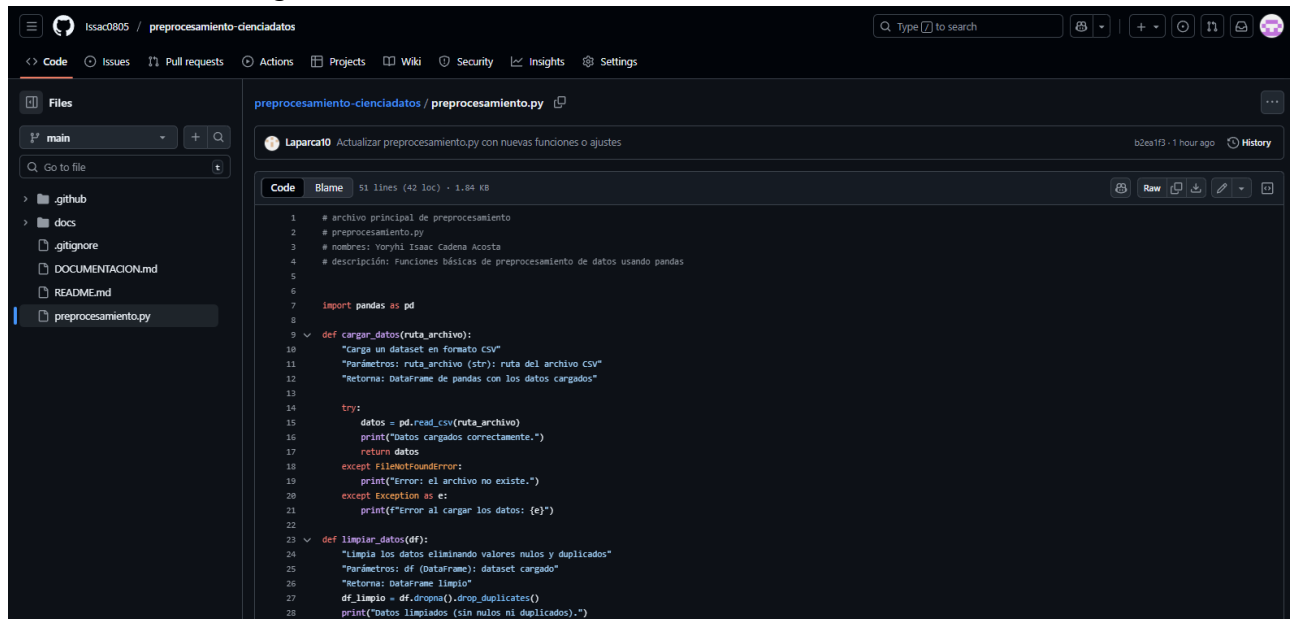
Commit	Message	Time
Laparc10	Agregar evidencias finales y documentación completa	2731794 - 1 minute ago
	Corregir activador de eventos y estructura del workflow de ...	35 minutes ago
	Agregar evidencias finales y documentación completa	1 minute ago
	Commit inicial: agregar .gitignore y README	19 hours ago
	Agregar evidencias finales y documentación completa	1 minute ago
	Actualizar README con información final del proyecto	2 hours ago
	Actualizar preprocesamiento.py con nuevas funciones o ajus...	1 hour ago

The repository contains the following files:

- .github/workflows
- docs/evidencias
- .gitignore
- DOCUMENTACION.md
- README.md
- preprocesamiento.py

The repository is named 'Preprocesamiento - Ciencia de Datos'.

## 12. Visualización del código en GitHub

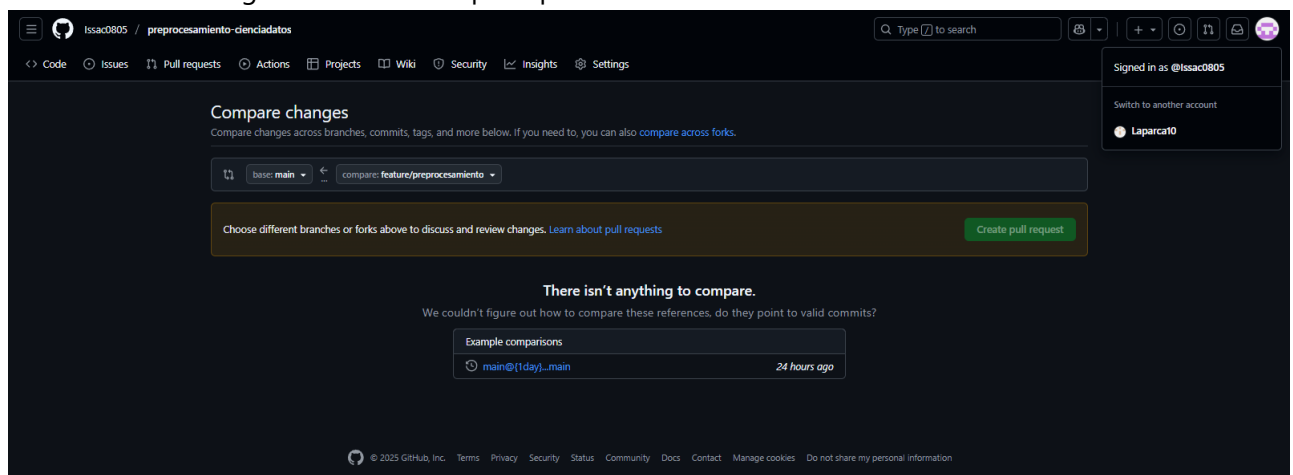


The screenshot shows the GitHub interface for the repository 'preprocesamiento-cienciadatos'. The file 'preprocesamiento.py' is selected, showing its code. The file is 51 lines long, 42 loc, and 1.04 KB. The code is in Python and uses pandas for data processing. The file was updated by 'Laparcit0' 1 hour ago.

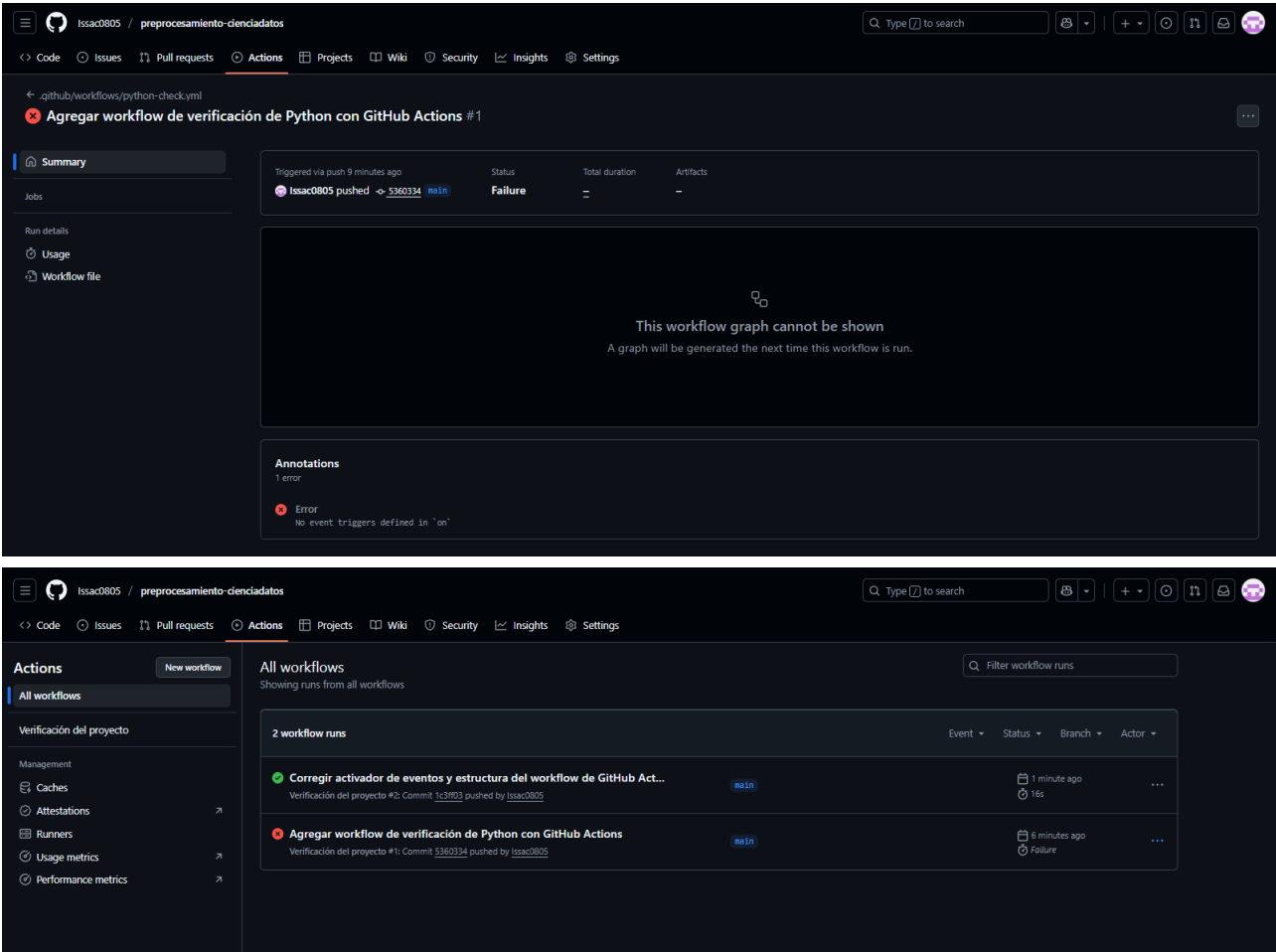
```

1 # archivo principal de preprocesamiento
2 # preprocesamiento.py
3 # nombres: Yoryhi Isaac Cadena Acosta
4 # descripción: Funciones básicas de preprocesamiento de datos usando pandas
5
6 import pandas as pd
7
8
9 def cargar_datos(ruta_archivo):
10     """Carga un dataset en formato csv"""
11     """Parámetros: ruta_archivo (str): ruta del archivo CSV"""
12     """Retorna: DataFrame de pandas con los datos cargados"""
13
14     try:
15         datos = pd.read_csv(ruta_archivo)
16         print("Datos cargados correctamente.")
17         return datos
18     except FileNotFoundError:
19         print("Error: el archivo no existe.")
20     except Exception as e:
21         print(f"Error al cargar los datos: {e}")
22
23 def limpiar_datos(df):
24     """Limpia los datos eliminando valores nulos y duplicados"""
25     """Parámetros: df (DataFrame): dataset cargado"""
26     """Retorna: DataFrame limpio"""
27     df_limpio = df.dropna().drop_duplicates()
28     print("Datos limpiados (sin nulos ni duplicados).")
  
```

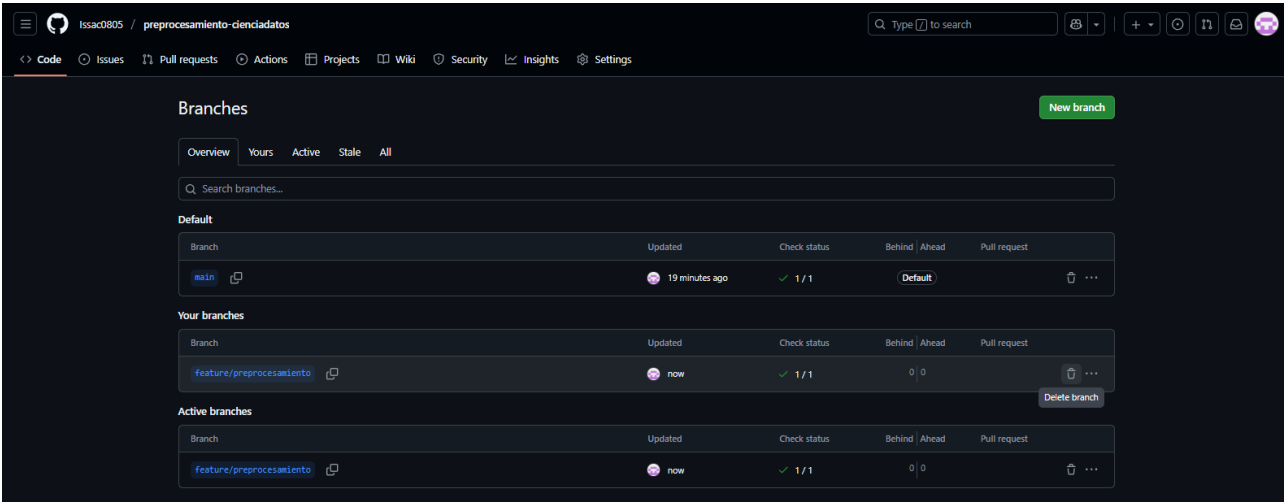
13. Pull Request y fusión en GitHub Se realizó la simulación de una Pull Request desde la rama `feature/preprocesamiento` hacia la rama `main`, gitHub indicó que ambas ramas estaban sincronizadas a la fusión previa realizada desde la terminal con Git, esto afirma que los cambios fueron correctamente integrados en la rama principal.



14. GitHub Actions, y ejecución del Workflow Se configuró un flujo de trabajo automático en `.github/workflows/python-check.yml`, el flujo se ejecuta cada vez que se realiza un push o un pull request en la rama `main`, durante la primera ejecución hubo un error de configuración que se corrigió, validando el script `preprocesamiento.py` y así mostrando la automatización CI/CD.



15. Eliminación de la rama feature/preprocesamiento tras la fusión |



6. Enlace al repositorio en GitHub

<https://github.com/Issac0805/preprocesamiento-cienciadatos>

7. Conclusión

Este proyecto ayudó a practicar el uso de Git y GitHub y guardar los cambios del código de forma ordenada, también sirvió para aplicar pasos básicos para preparar los datos antes de analizarlos, también se siguió el

proceso donde: se crearon líneas de trabajo (ramas), se guardaron cambios con notas (commits), se unieron los avances (merge), y se documentó todo lo hecho con documentación.