

Introducción

Este proyecto final del primer curso de Big Data e Inteligencia Artificial se centra en el proceso completo de análisis de datos: desde su ingesta, almacenamiento y procesamiento, hasta su visualización e interpretación.

Para desarrollar este estudio, hemos elegido la climatología como temática, enfocándonos específicamente en las precipitaciones en Australia entre los años 2008 y 2017. (Los archivos han sido sacado del siguiente enlace: <https://www.kaggle.com/datasets/trisha2094/weatheraus>).

Nuestro objetivo es extraer y analizar métricas clave que permitan comprender mejor las tendencias climáticas en este periodo.

Las métricas principales que evaluaremos son:

- Fecha con la temperatura más alta registrada.
- Fecha con la temperatura más baja registrada.
- Fecha con la mayor cantidad de lluvia registrada.
- Fecha con el viento más extremo registrado.
- Periodo de sequía más largo registrado.
- Cantidad de días sin lluvia por año.
- Media anual de lluvia y temperatura.
- Variabilidad de temperatura máxima y mínima por año.
- Días que superaron los 40°C por año (temperatura extrema).
- Días con temperaturas menores a -5°C por año (*en Australia, el frío extremo es poco frecuente, por lo que establecemos este umbral como referencia*).
- Días con viento extremo (>93 km/h) por año.

Ingesta y almacenamiento de datos

Para iniciar este proyecto lo primero que tenemos que hacer es la ingesta de datos, para ello vamos a usar S3 para la entrada y el almacenamiento de archivos, al igual que usaremos en su transformación y filtración de datos Glue, después lo volveremos a volcar con los datos que requerimos en el S3 y realizaremos las consultas con colab.

También trataremos los datos con un ciclo de vida para que no ocupe más recursos de los que se debe.

Inserción de datos:

Lo primero que se va a realizar en la inserción de datos, para ello vamos al S3 y crearemos un bucket de entrada de datos (nombre dado finalentrada):

Crear bucket [Información](#)

Los buckets son contenedores de datos almacenados en S3.

Configuración general

Región de AWS
EE.UU. Este (Norte de Virginia) us-east-1

Tipo de bucket [Información](#)

☒ **Uso general**
Recomendado para la mayoría de los casos de uso y patrones de acceso. Los buckets de uso general son del tipo de bucket de S3 original, permiten una combinación de clases de almacenamiento que almacenan objetos de forma redundante en múltiples zonas de disponibilidad.

☐ **Directorio**
Recomendado para casos de uso de baja latencia. Estos buckets utilizan únicamente la clase de almacenamiento S3 Express One Zone, que proporciona un procesamiento más rápido de los datos dentro de una única zona de disponibilidad.

Nombre del bucket [Información](#)
finalentrada

Los nombres de los buckets deben tener entre 3 y 63 caracteres y ser únicos dentro del espacio de nombres global. Los nombres de los buckets también deben empezar y terminar con una letra o un número. Los caracteres válidos son a-z, 0-9, guiones (-) y puntos (.). [Más información](#)

Copiar la configuración del bucket existente (opcional)
Solo se copia la configuración del bucket en los siguientes ajustes.

[Elegir el bucket](#)

Formato: s3://bucket/prefijo

Propiedad de objetos [Información](#)

El bucket "finalentrada" se creó correctamente
Para cargar archivos y carpetas, o para configurar ajustes adicionales del bucket, elija [Ver detalles](#).

Instantánea de la cuenta: actualizada cada 24 horas [Todas las regiones de AWS](#) [Ver panel de Storage Lens](#)

Storage Lens permite visualizar el uso del almacenamiento y las tendencias de la actividad. Las métricas no incluyen los buckets de directorio. [Más información](#)

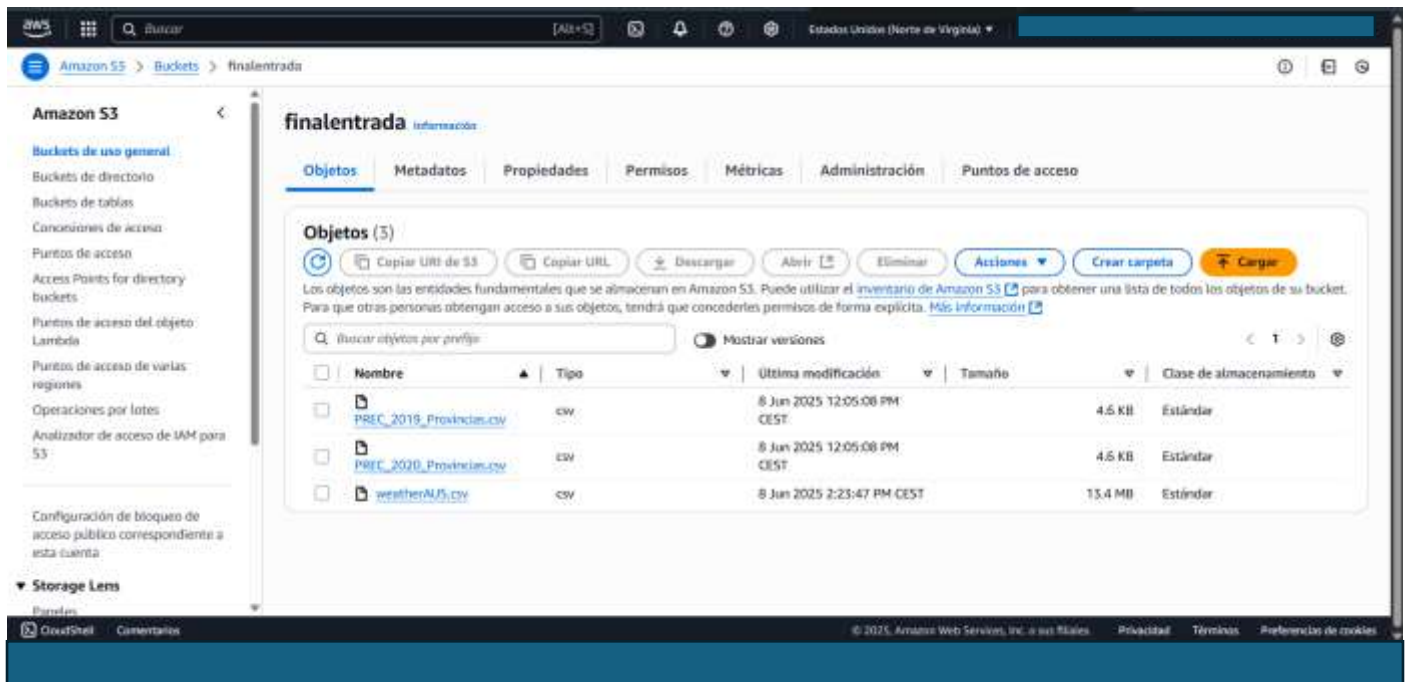
Buckets de uso general | **Buckets de directorio**

Buckets de uso general (1) [Información](#) [Todas las regiones de AWS](#)

Los buckets son contenedores de datos almacenados en S3.

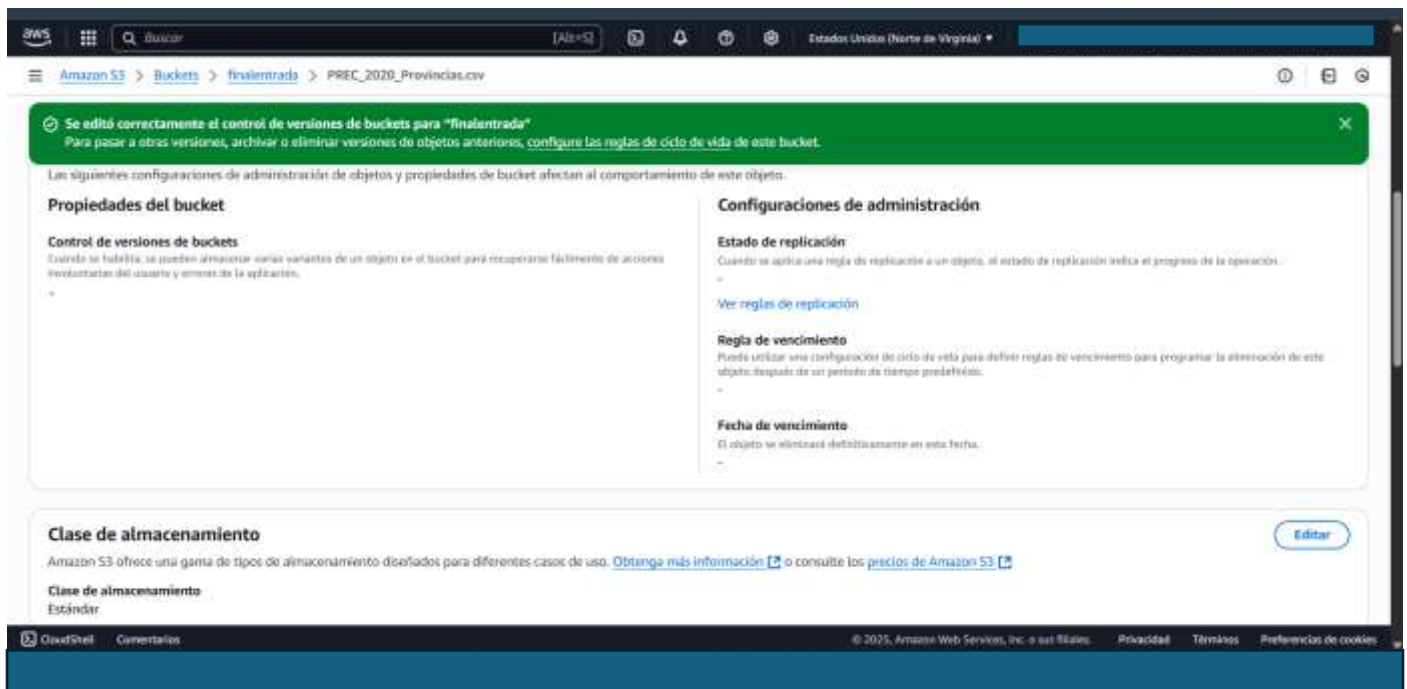
Nombre	Región de AWS	Analizador de acceso de IAM	Fecha de creación
finalentrada	EE.UU. Este (Norte de Virginia) us-east-1	Ver analizador para us-east-1	8 Jun 2025 11:58:17 AM CEST

Subimos con los archivos que vamos a trabajar:

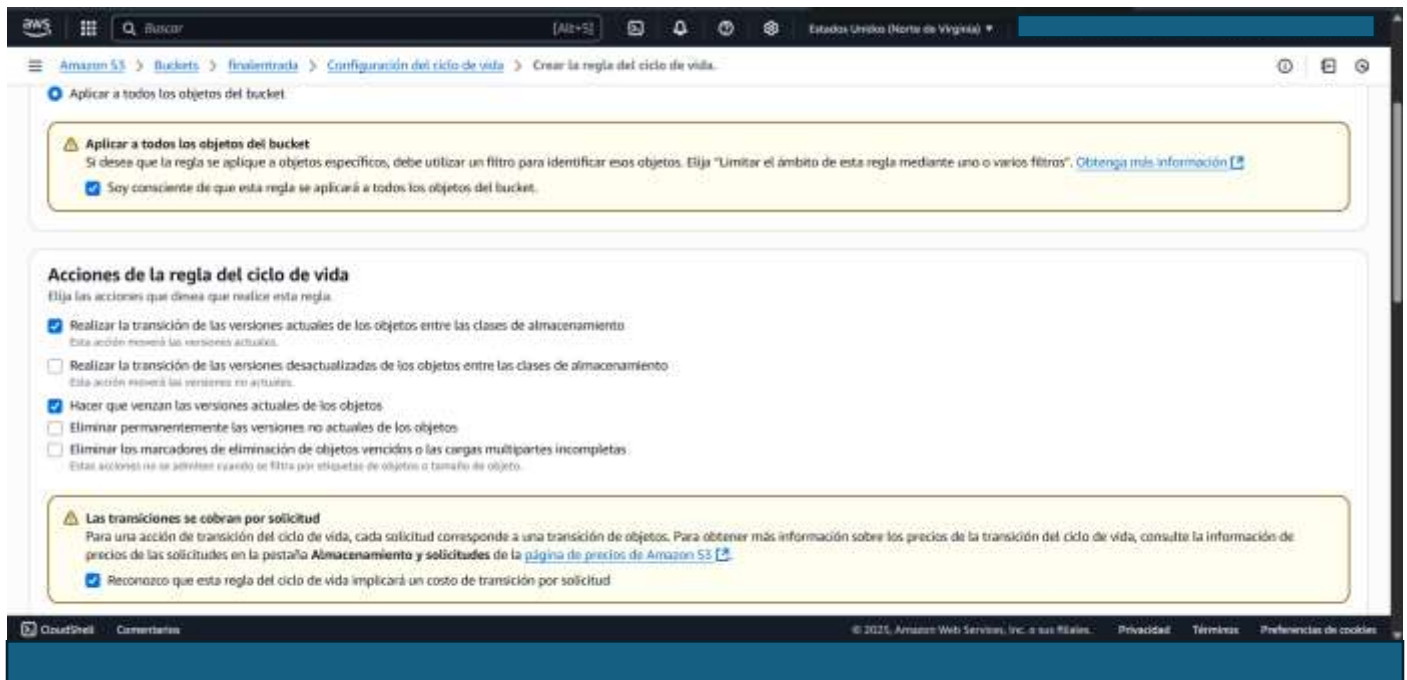


(se suben 3 archivos por los experimentos durante el trabajo)

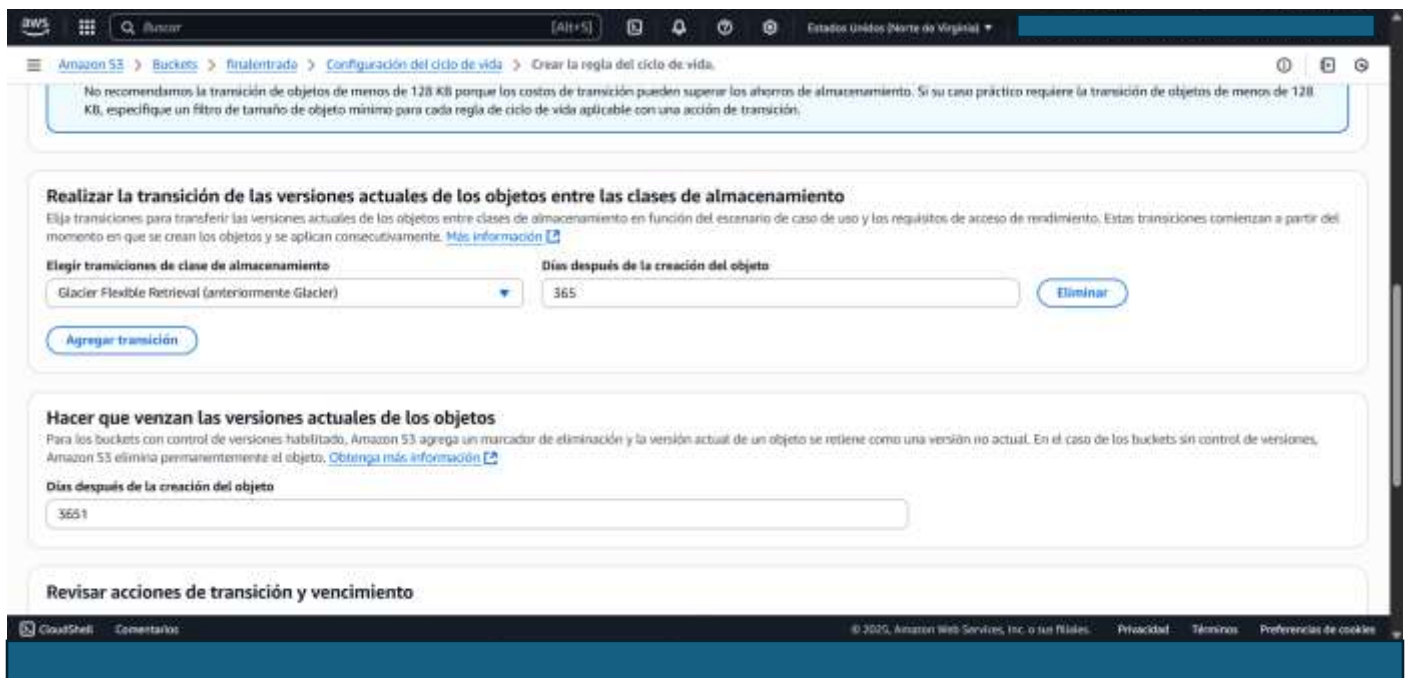
Tras subir los archivos y antes de hacer nada, tocaremos el ciclo de vida. Para ello lo primero que tenemos que hacer es habilitar el control de versiones para evitar problemas.

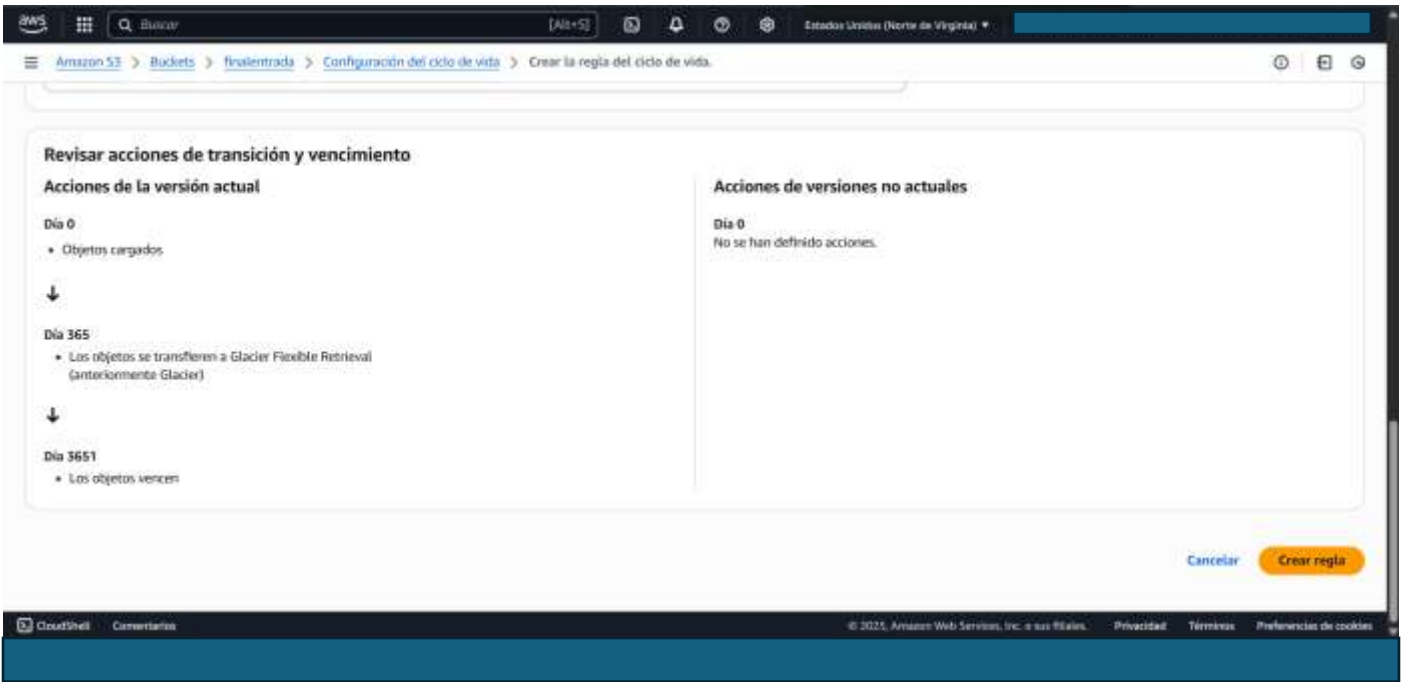


Dado que estos datos no son de alta prioridad, pero queremos conservarlos durante el mayor tiempo posible sin ocupar recursos innecesarios, configuraremos una política de ciclo de vida en que no se borren durante 10 años. Esto nos permitirá optimizar el almacenamiento, asegurando que los datos permanezcan accesibles, pero con un uso eficiente de los recursos disponibles.

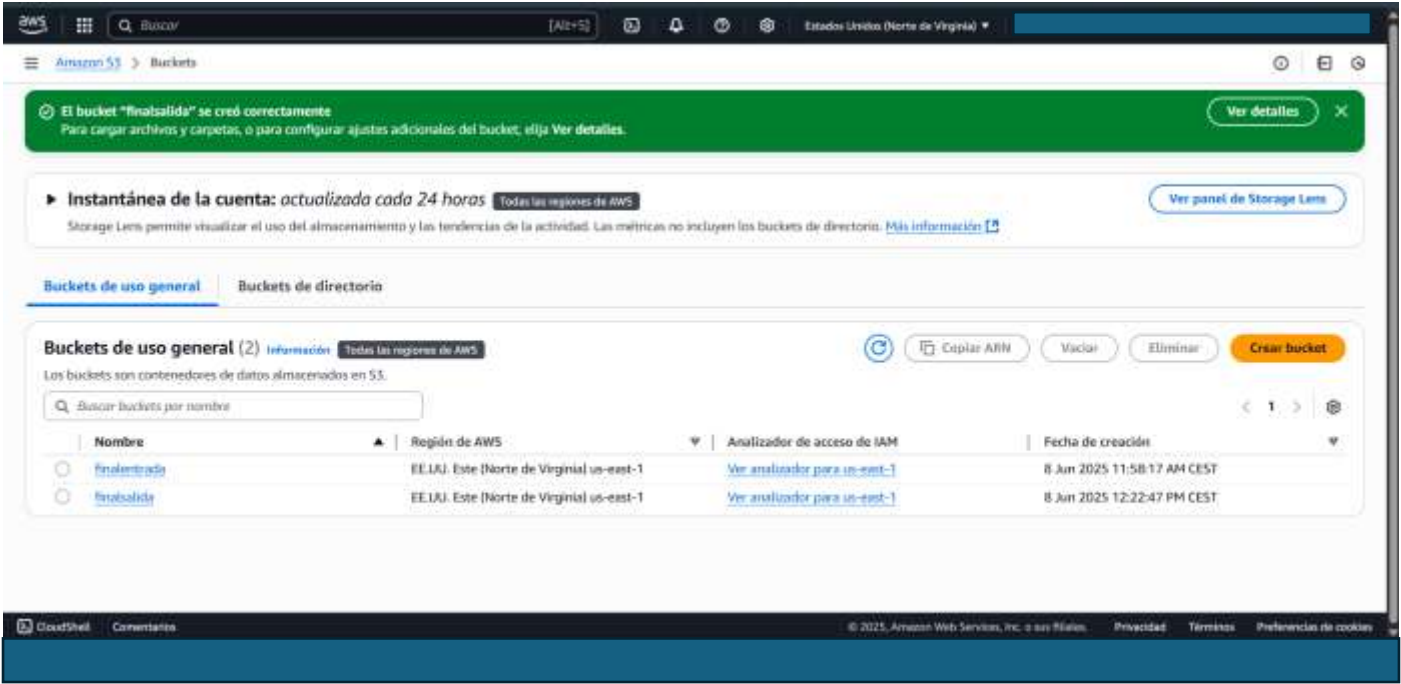


Como es un archivo de gran duración, lo dejaremos durante un año presente, tras pasar dicho año, se desplazara a s3glacier flexible retrieval. Tras pasar 10 años, se eliminará.





Tras crear la reglade vida, ya pasamos a generar otro bucket de salida de datos que lo usaremos más adelante:

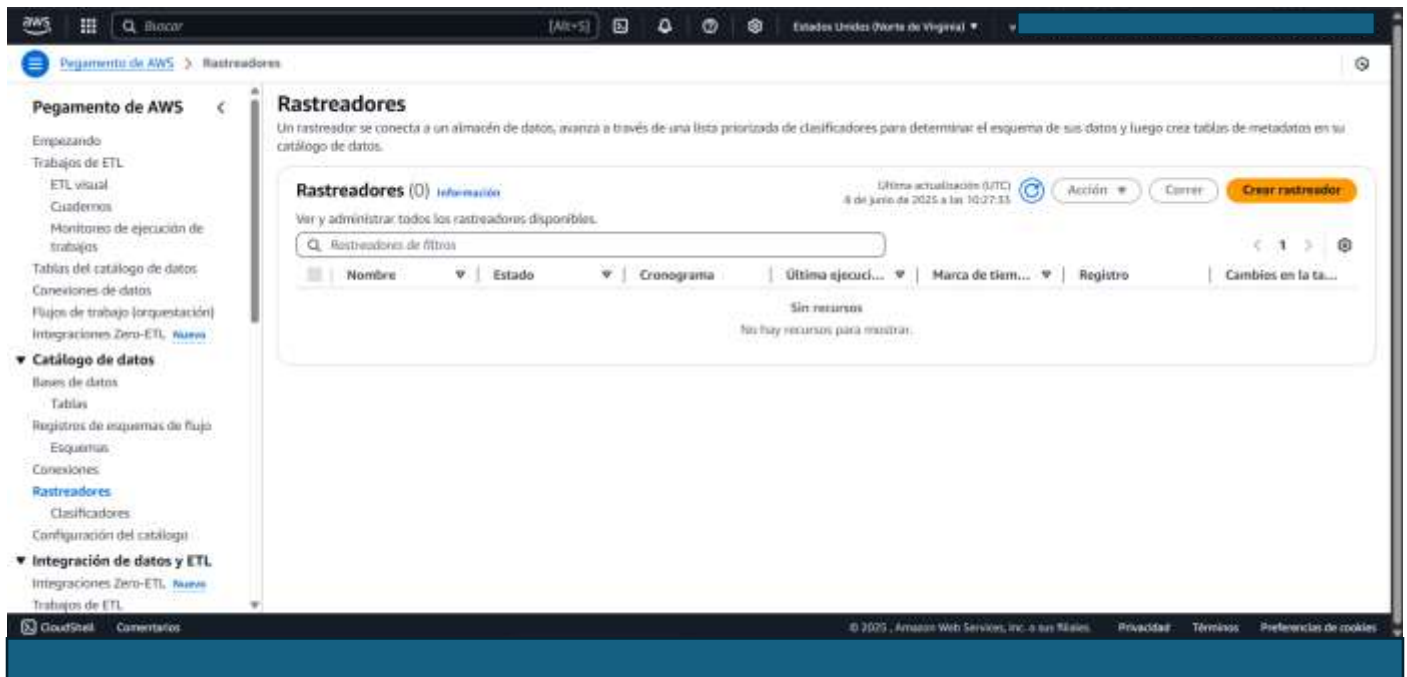


El nombre dado es finalsalida.

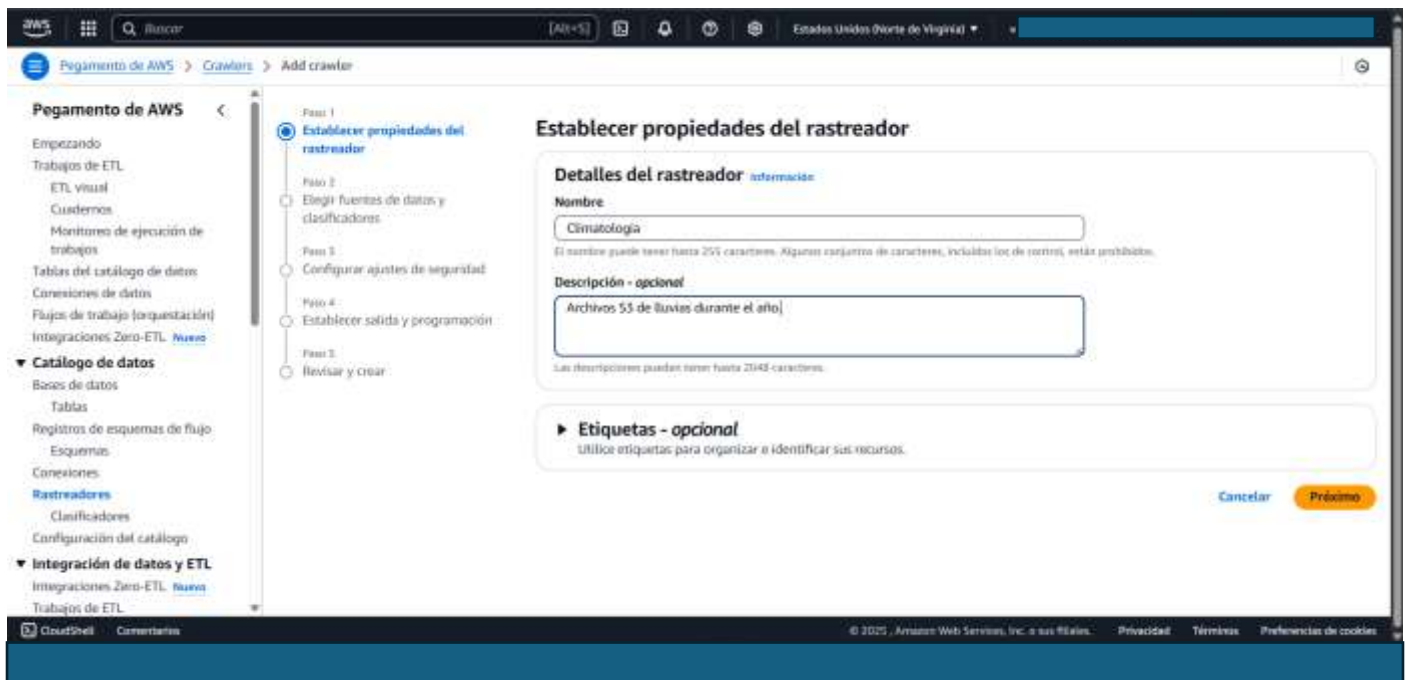
Procesamiento de datos con AWS Glue

Una vez configurado el almacenamiento y el ciclo de vida de los datos, es momento de proceder con su transformación y filtrado mediante AWS Glue. Este servicio nos permite automatizar la extracción y preparación de los datos almacenados en S3, optimizando su estructura para posteriores consultas y análisis.

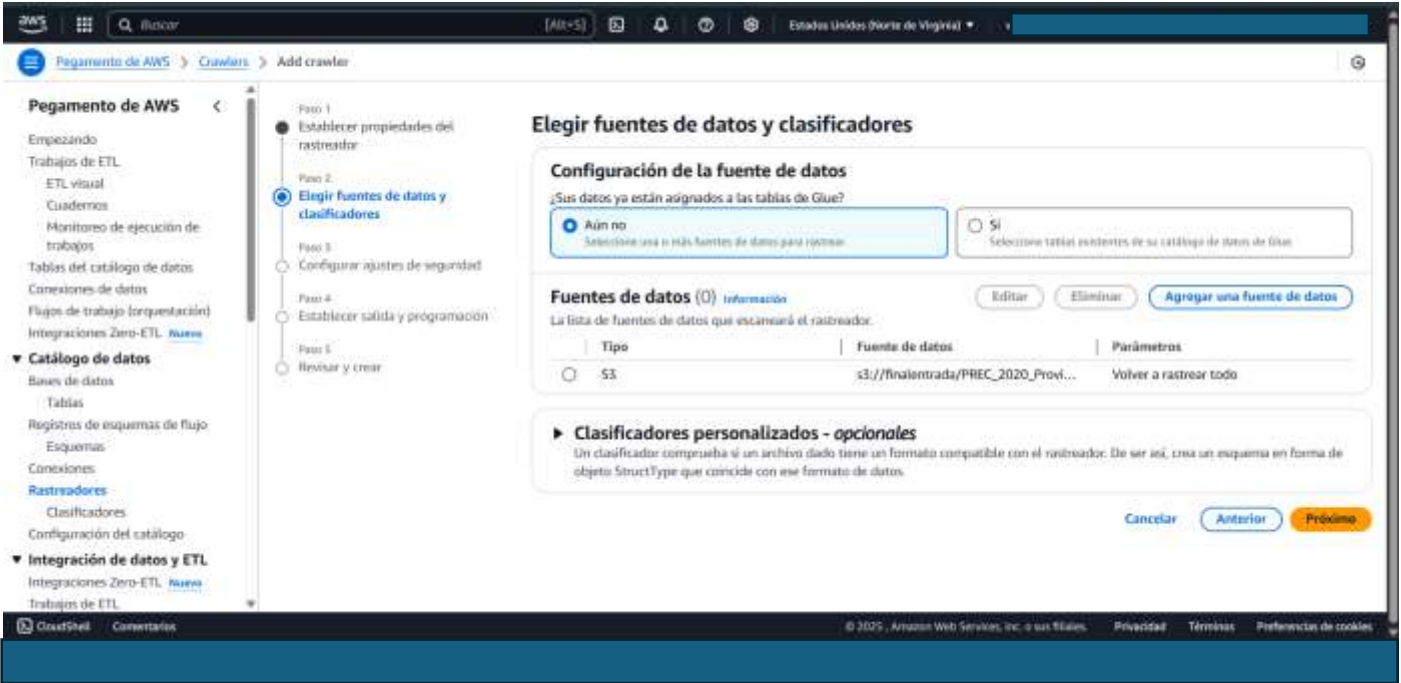
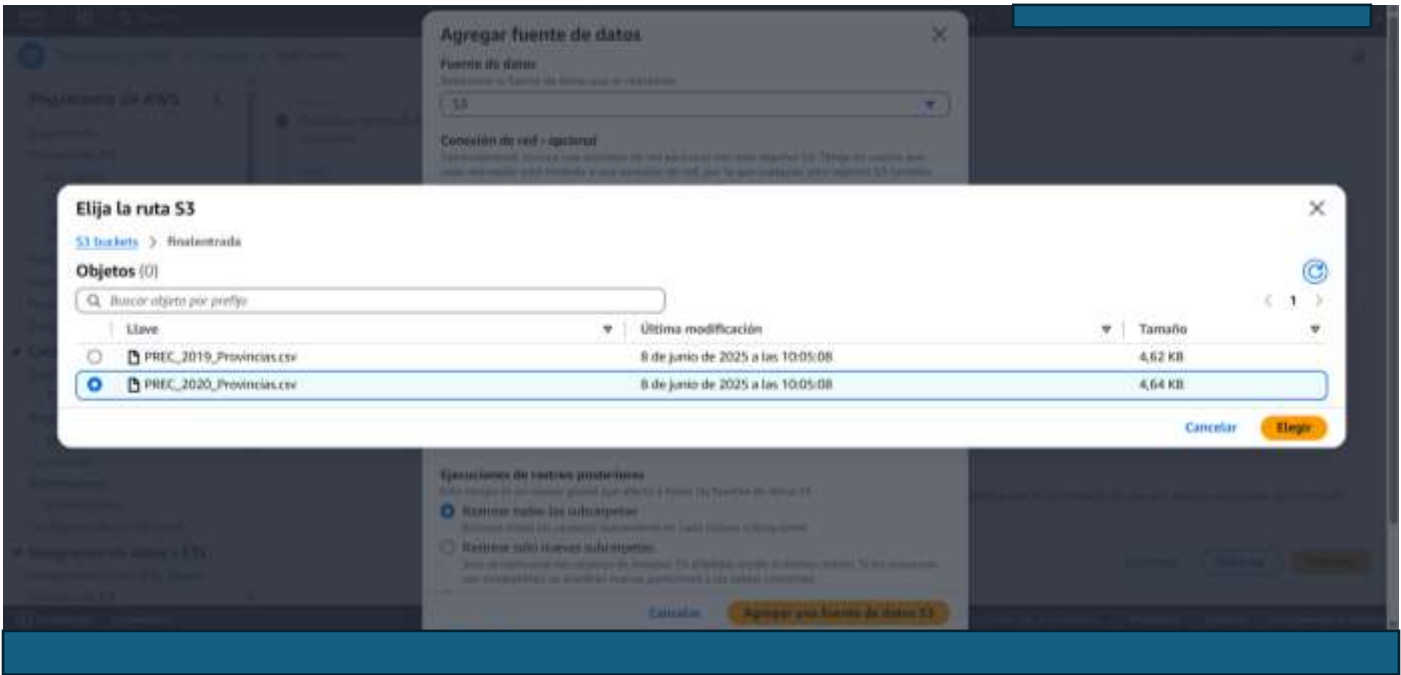
El primer paso en este proceso fue la creación de un rastreador dentro de AWS Glue, el cual se encargó de localizar y catalogar los archivos previamente almacenados. Para ello, establecimos un rastreador con el nombre "climatología", asegurándonos de que estuviera correctamente vinculado al bucket de entrada. Con este rastreador configurado, AWS Glue identificó los archivos disponibles en S3,

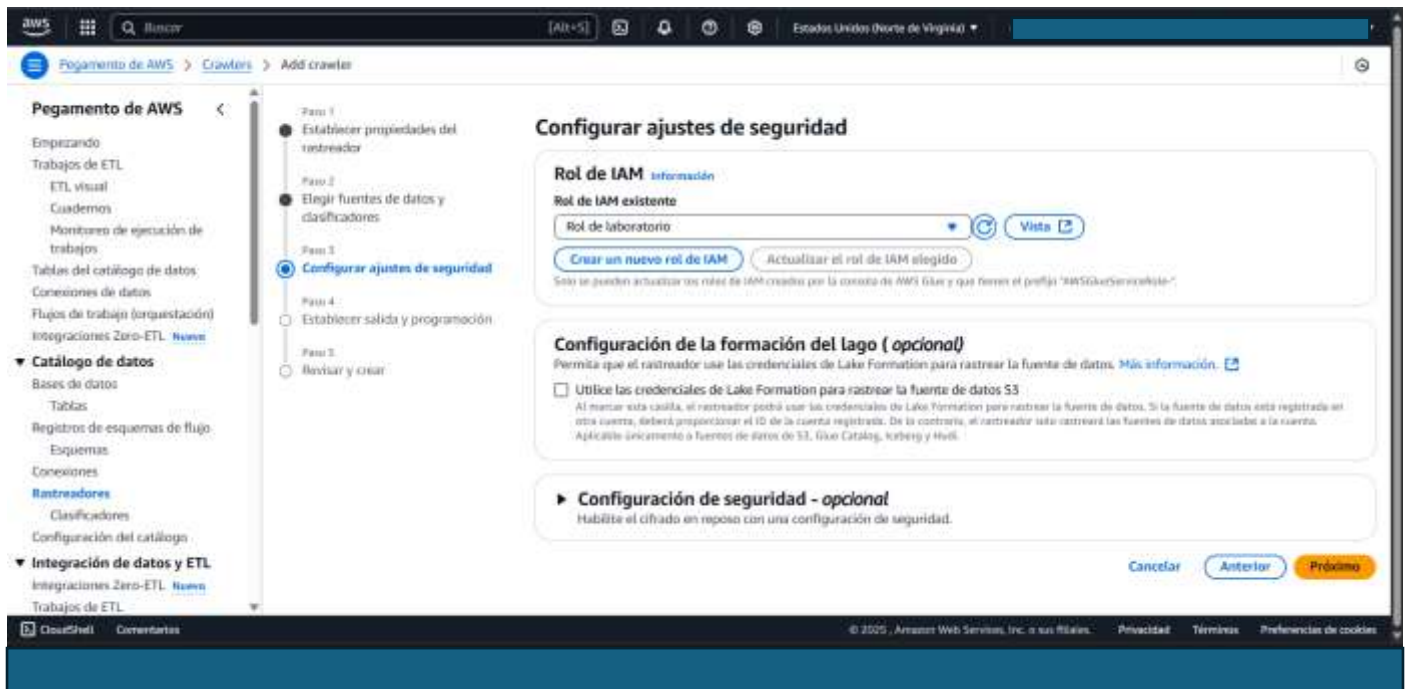


Le daremos el nombre de climatología.

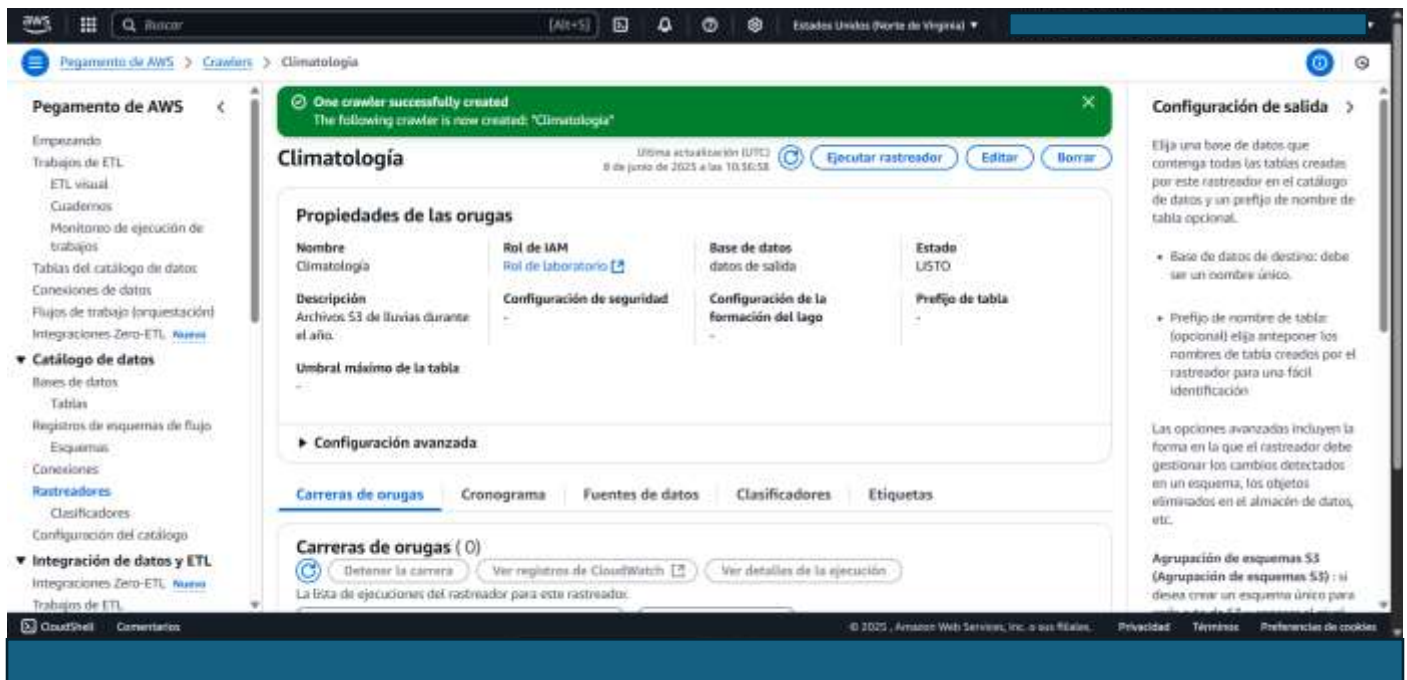


Elegimos el archivo S3 que vamos a trabajar (después se modificaría esto para detectar la carpeta)





Creamos y dejamos listo:



Una vez completada la ejecución del rastreador, accedemos al visualizador de tablas de AWS Glue, donde podemos examinar los datos detectados antes de proceder con su transformación final. En esta etapa, se verificó la correcta delimitación de los datos, asegurándonos de que estuvieran estructurados de manera clara y consistente. Dado que el archivo contenía información que no consideramos relevante para el análisis, realizamos una limpieza y filtrado, eliminando columnas innecesarias y organizando los registros con los parámetros específicos que necesitamos para la extracción de métricas.

El procedimiento es: primero colocar una entrada de S3, eligiendo el archivo que vamos a utilizar.

Trabajofinal1

Visual | Script | Job details | Runs | Data quality | Schedules | Version Control

Data source - S3 bucket Amazon S3

Data preview (200) Info READY End session Previewing 23 of 23 fields

Filter sample dataset

date	location	mintemp	maxtemp	rainfall
2008-12-01	Albury	13.4	22.9	0.6
2008-12-02	Albury	7.4	25.1	0
2008-12-03	Albury	12.9	25.7	0

Additional options

Delimiter: Comma (,)

Escape character - optional: Enter a character to use for escaping

Quote character: Double quote (")

☒ First line of source file contains column headers

☐ Records in source files can span multiple lines

Unsaved job found. We found an unsaved job, do you wish to restore it? Restore

Para poder separar los datos en tablas legibles, hemos tenido que poner la delimitación con la (,)

A continuación, como dicho archivo tiene muchos datos que no los consideramos necesarios, los eliminaremos de la tabla al igual que pondremos los datos correspondientes a dicho dato:

Trabajofinal1

Visual | Guion | Detalles del trabajo | Camaras | Calidad de los datos

Transformar

Nombre: Eliminacioncolumnas

Padres de nodos: Amazon S3

Cambiar esquema (Aplicar mapeo)

Clave de origen	Clave de o...	Tipo de datos	Gota
fecha	date	cadena	<input type="checkbox"/>
ubicación			<input checked="" type="checkbox"/>
mintemp	mintemp	doble	<input type="checkbox"/>

Trabajofinal1

Última modificación: 06/08/2025, 16:49:14

Comportamiento

Ahorrar

Correr

Visual

Guion

Detalles del trabajo

Carreras

Calidad de los datos

Transformar - Cambiar e...
Eliminacioncolumnas

Vista previa de datos

Esquema de salida

Vista previa de datos (200)

información

LISTO

Finalizar sesión

Vista previa de 5 de 5 campos

Filtrar conjunto de datos de muestra

fecha	mintemp	temperatura máxima	lluvia
1 de diciembre de 2008	13.4	22.9	0.6
02-12-2008	7.4	25.1	0

Cambiar esquema (Aplicar mapeo)

Clave de origen	Clave de objetivo	Tipo de datos	Gota
fecha	date	cadena	<input type="checkbox"/>
ubicación			<input checked="" type="checkbox"/>
mintemp	mintemp	doblo	<input type="checkbox"/>
temperatura máxima	maxtemp	doblo	<input type="checkbox"/>
lluvia	rainfall	doblo	<input type="checkbox"/>
evaporación			<input checked="" type="checkbox"/>
luz solar			<input checked="" type="checkbox"/>
ráfagas de viento			<input checked="" type="checkbox"/>
velocidad de las ráfagas de viento	windgustspeed	double	<input type="checkbox"/>
viento9am			<input checked="" type="checkbox"/>
viento3pm			<input checked="" type="checkbox"/>

CloudShell

Comentarios

© 2025, Amazon Web Services, Inc. o sus filiales.

Privacidad

Términos

Preferencias de cookies

Trabajofinal1

Última modificación: 06/08/2025, 16:49:14

Comportamiento

Ahorrar

Correr

Visual

Guion

Detalles del trabajo

Carreras

Calidad de los datos

Transformar - Cambiar e...
Eliminacioncolumnas

Vista previa de datos

Esquema de salida

Vista previa de datos (200)

información

LISTO

Finalizar sesión

Vista previa de 5 de 5 campos

Filtrar conjunto de datos de muestra

fecha	mintemp	temperatura máxima	lluvia
1 de diciembre de 2008	13.4	22.9	0.6
02-12-2008	7.4	25.1	0

Cambiar esquema (Aplicar mapeo)

Clave de origen	Clave de objetivo	Tipo de datos	Gota
viento3pm			<input checked="" type="checkbox"/>
velocidad del viento9am			<input checked="" type="checkbox"/>
velocidad del viento3pm			<input checked="" type="checkbox"/>
humedad9am			<input checked="" type="checkbox"/>
humedad3pm			<input checked="" type="checkbox"/>
presión9am			<input checked="" type="checkbox"/>
presión3pm			<input checked="" type="checkbox"/>
nube9am			<input checked="" type="checkbox"/>
nube3pm			<input checked="" type="checkbox"/>
temperatura9am			<input checked="" type="checkbox"/>
temperatura3pm			<input checked="" type="checkbox"/>
lluvia hoy			<input checked="" type="checkbox"/>
lluvia mañana			<input checked="" type="checkbox"/>

CloudShell

Comentarios

© 2025, Amazon Web Services, Inc. o sus filiales.

Privacidad

Términos

Preferencias de cookies

Con los datos ya preparados y ajustados, procedimos a volcarlos en el bucket de salida previamente configurado, "finalsalida".

Trabajofinal1

Last modified on 8/6/2025, 16:58:43

Actions Save Run

Visual Script Job details Runs Data quality Schedules Version Control

Transform : Change Sch...
Eliminacioncolumnas

Data target : S3 bucket
Salida

Target node not supported
You have selected a data target node which is not supported for data preview. Please select another type of node instead.

Data target properties - 53

Name
Salida

Node parents
Choose which nodes will provide inputs for this one.
Choose one or more parent node
Eliminacioncolumnas
ApplyMapping - Transform

Format
CSV

Compression Type
Snappy

CloudShell Comentarios

© 2025, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

Guardamos y corremos.

Trabajofinal1

Last modified on 8/6/2025, 16:58:30

Actions Save Run

Visual Script Job details Runs Data quality Schedules Version Control

Job runs (1/2) info

Last updated 6/7/2025 at 14:58:34

View details Stop job run Troubleshoot with AI

Table View Card View

Filter job runs by property

Run status	Retries	Start time (Local)	End time (Local)	Duration	Capacity (DPUs)	Worker type	Glue version
Running	0	06/08/2025 16:58:32	-	0 s	10 DPUs	G.1X	5.0

Run details Input arguments (9) Logs Run insights Metrics Troubleshooting analysis - preview Spark UI

Stop job run

Job name
Trabajofinal1

Start time (Local)
06/08/2025 16:58:32

Glue version
5.0

Last modified on (Local)
06/08/2025 16:58:32

End time (Local)

Worker type
G.1X

Log group name
/aws-glue/jobs

CloudShell Comentarios

© 2025, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

Amazon S3 console showing the 'Trabajofinal1' job run details. The job run is 'Succeeded' with a duration of 1 m 54 s. The logs show the execution of the 'aws-glue-d1-package-5.0.532.jar' and the 'org.apache.spark.util.ShutdownHookWrapper'.

Run status	Retries	Start time (Local)	End time (Local)	Duration	Capacity (DPU)	Worker type	Glue version
Succeeded	0	06/08/2025 16:58:32	06/08/2025 17:00:25	1 m 54 s	10 DPU	G.1X	5.0
Failed	0	06/08/2025 14:56:44	06/08/2025 14:59:11	2 m 13 s	10 DPU	G.1X	5.0

Log details:

```
at:518) ~[aws-hadoop-assembly-2.69.0.jar:1]
at: com.amazonaws.services.glue.LogPusher.upload(LogPusher.scala:72) ~[aws-glue-d1-package-5.0.532.jar:1]
at: org.apache.spark.util.ShutdownHookWrapper$.shutdownHook$1(ShutdownHookWrapper.scala:9) ~[aws-glue-d1-package-5.0.532.jar:1,5.4-amzn-0]
```

Revisamos que tenemos el archivo listo en el S3.

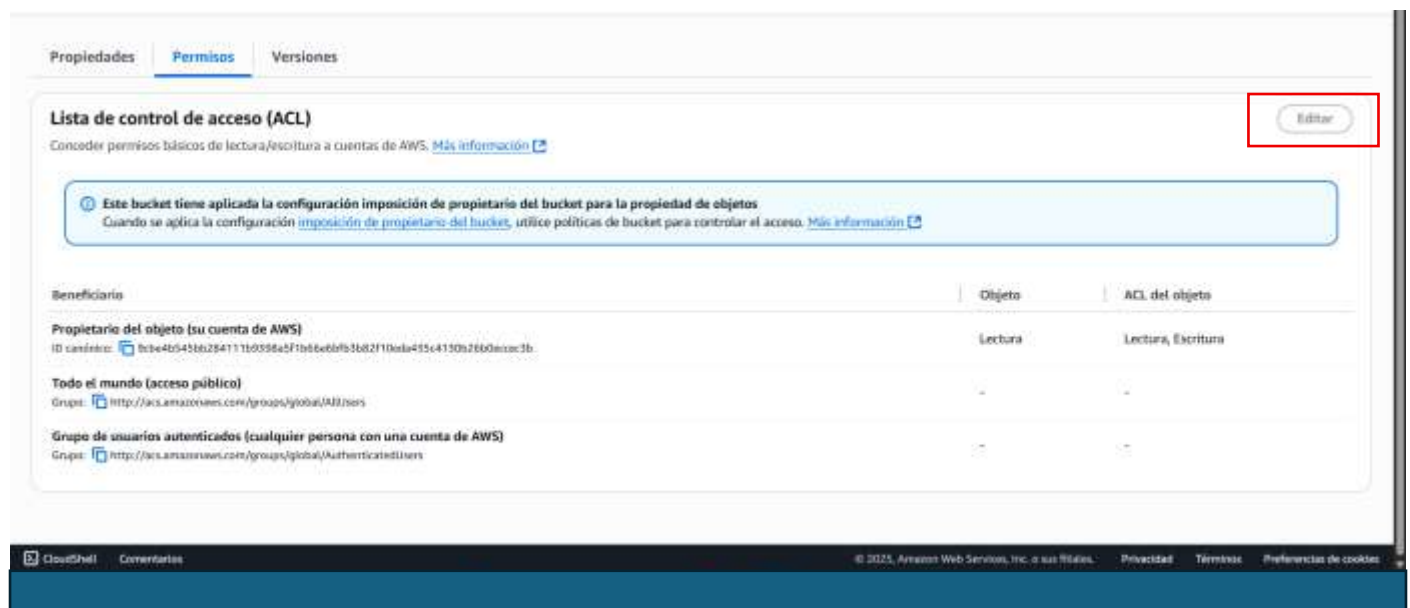
Amazon S3 console showing the 'finalsalida' bucket. The bucket contains two objects: 'PREC_2020_Provincias.csv' (4.5 KB) and 'run-1749394804014-part-1-00000' (4.1 MB).

Nombre	Tipo	Última modificación	Tamaño	Clase de almacenamiento
PREC_2020_Provincias.csv	CSV	8 Jun 2025 1:52:22 PM CEST	4.5 KB	Estándar
run-1749394804014-part-1-00000	-	8 Jun 2025 5:00:13 PM CEST	4.1 MB	Estándar

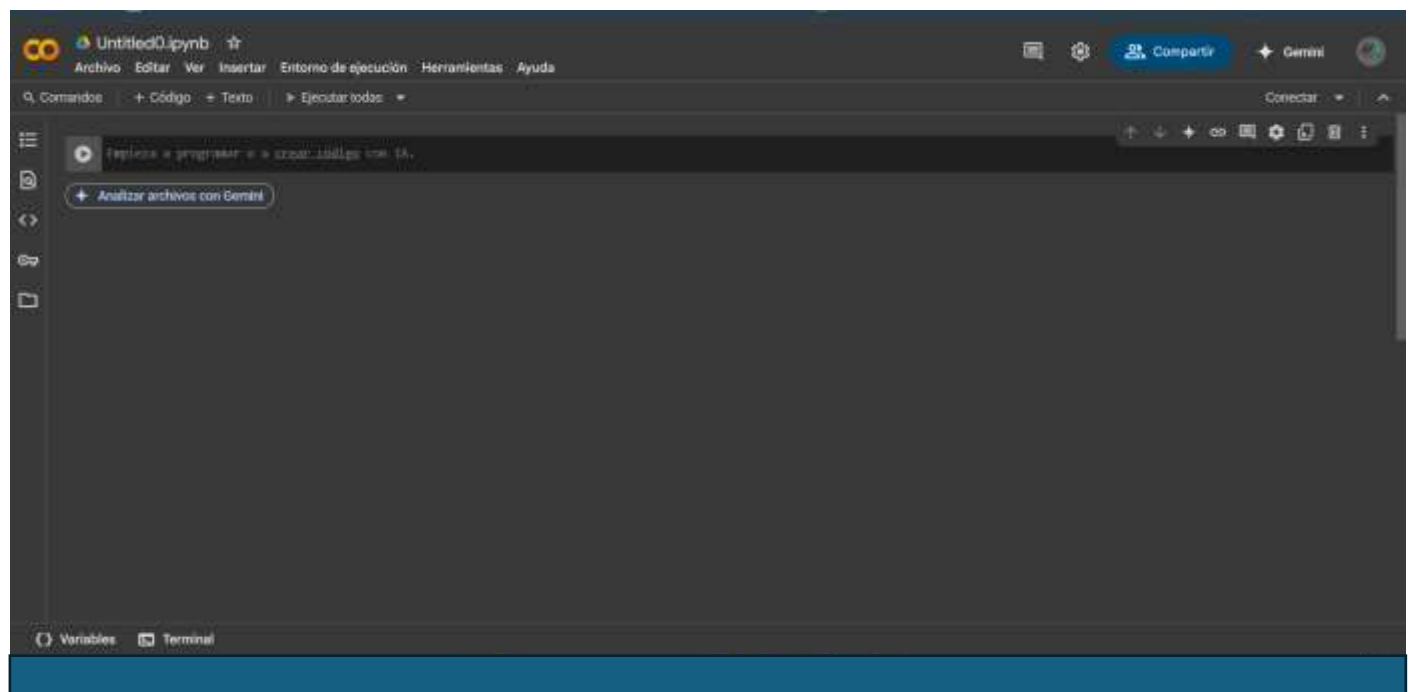
Este paso nos permitió contar con un conjunto de datos refinado, listo para ser consultado y analizado en profundidad. Gracias a AWS Glue, pudimos automatizar el proceso de identificación, limpieza y almacenamiento, optimizando el flujo de trabajo y asegurando que los datos sean accesibles en su formato más útil para la fase final de visualización en Google Colab.

Visualización de datos

Para importar los datos desde S3, nos encontramos con la limitación del rol de laboratorio, que impide la importación directa desde Colab. Como alternativa, optamos por descargar manualmente los archivos y trabajar con ellos localmente en Colab. Este método nos permitió acceder a la información sin restricciones y proceder con el análisis de datos.



Abrimos el colab.



Importamos en el S3 el archivo que hemos descargado.

Importamos y previsualizamos los archivos para ver que se han cargado correctamente (se tubo varios problemas de visualización y se tubo que indagar como poder verlos sin cortar ni alteraciones)

Te damos la bienvenida a Colaboratory. No se pueden guardar cambios.

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda

Comandos + Código + Texto Ejecutar todas Copiar en Drive

Archivos

- Analiza tus archivos con código escrito por Gemini Subir
- sample_data
- nu-1749394804014-part-000...

Interrogación de video: predice lo que ocurre entre el primer y el último fotograma de un video.

```
import pandas as pd

# Cargar el archivo CSV desde Colab
df = pd.read_csv("/content/nu-1749394804014-part-000000.csv") # Cambia "archivo.csv" por el nombre real de tu archivo.

# Configurar Pandas para mostrar todo el contenido sin truncar
pd.set_option("display.max_rows", None) # Muestra todas las filas
pd.set_option("display.max_columns", None) # Muestra todas las columnas
pd.set_option("display.max_colwidth", None) # Evita que se corten valores largos

# Mostrar el dataframe completo
print(df)
```

164269	2014-03-23	22.0	34.7	NaN	43.0
164270	2014-03-24	NaN	31.0	NaN	NaN
164271	2014-03-25	32.8	28.2	NaN	43.0
164272	2014-03-26	32.2	29.7	0.0	37.0
164273	2014-03-27	31.5	32.5	0.0	30.0
164274	2014-03-28	33.8	32.8	0.0	51.0
164275	2014-03-29	32.5	35.1	0.0	28.0
164276	2014-03-30	38.1	37.4	0.0	39.0
164277	2014-03-31	38.7	37.6	0.0	48.0
164278	2014-04-01	39.2	37.2	3.0	41.0
164279	2014-04-02	30.5	36.7	0.0	50.0
164280	2014-04-03	39.1	28.8	3.0	41.0
164281	2014-04-04	36.0	28.7	0.0	35.0
164282	2014-04-05	36.7	32.2	0.0	30.0
164283	2014-04-06	39.3	22.0	3.2	35.0

70.34 GB de espacio disponible

Variables Terminal

✓ 17:32 Python 3

Una vez visto que los archivos funcionan, comenzaremos con nuestra exposición de datos:

Lo primero que hacemos es dar formato y medidas para evitar fallos a la hora de coger los datos. (también se tubo que investigar pues las formulas son bastante más complejas y son muchos datos a analizar)

Te damos la bienvenida a Colaboratory. No se pueden guardar cambios.

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda

Comandos + Código + Texto Ejecutar todas Copiar en Drive

Archivos

- Analiza tus archivos con código escrito por Gemini Subir
- sample_data
- media_por_fecha.csv
- nu-1749394804014-part-000...

```
[25]: import pandas as pd

# Ruta del archivo
ruta = "/content/nu-1749394804014-part-000000.csv"

# Cargar el archivo
df = pd.read_csv(ruta)

# Asegurar que 'date' sea de tipo fecha
df["date"] = pd.to_datetime(df["date"], errors="coerce")

# Extraer el año para análisis
df["year"] = df["date"].dt.year
# Si no se realiza dicho cambio se podría leer las tablas incorrectamente.

# Verificar que las columnas necesarias existen y convertirlas a numéricas si es necesario
columnas_numericas = ["rainfall", "mintemp", "maxtemp", "windgustspeed"]
for col in columnas_numericas:
    df[col] = pd.to_numeric(df[col], errors="coerce")

# Fechas con la temperatura más alta registrada
max_temp_valor = df["maxtemp"].max()
fechas_max_temp = df[df["maxtemp"] == max_temp_valor]["date"]
print(f"Fechas con la temperatura más alta registrada ({max_temp_valor}°C):\n{fechas_max_temp.tolist()}")

# Fechas con la temperatura más baja registrada
min_temp_valor = df["mintemp"].min()
```

70.34 GB de espacio disponible

Variables Terminal

✓ 18:36 Python 3

Tras tener los esquemas preparados, vamos a sacar la información y las medias que nos interesa:

En el primer bloque sacaremos los días más extremos que hemos tenido tanto de calor, frio, lluvia y viento.

```
Te damos la bienvenida a Colaboratory. No se pueden guardar cambios.
Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda
Comandos + Código + Texto Ejecutar todas Copiar en Drive

Archivos
+ Analiza tus archivos con código escrito por Gemini Subir
sample_data
media_por_fecha.csv
run:1749394804014-part-r-000...

# fecha con la temperatura más alta registrada
max_temp_valor = df["temp"].max()
fechas_max_temp = df[df["temp"] == max_temp_valor]["date"]
print(f"Fecha con la temperatura más alta registrada ({max_temp_valor})°C:\n{fechas_max_temp.tolist()}")

# fecha con la temperatura más baja registrada
min_temp_valor = df["temp"].min()
fechas_min_temp = df[df["temp"] == min_temp_valor]["date"]
print(f"Fecha con la temperatura más baja registrada ({min_temp_valor})°C:\n{fechas_min_temp.tolist()}")

# fecha con la mayor cantidad de lluvia registrada
max_lluvia_valor = df["rainfall"].max()
fechas_max_lluvia = df[df["rainfall"] == max_lluvia_valor]["date"]
print(f"Fecha con la mayor cantidad de lluvia registrada ({max_lluvia_valor}) mm:\n{fechas_max_lluvia.tolist()}")

# fecha con el viento más extremo registrado
max_viento_valor = df["windgustspeed"].max()
fechas_max_viento = df[df["windgustspeed"] == max_viento_valor]["date"]
print(f"Fecha con el viento más extremo registrado ({max_viento_valor}) km/h:\n{fechas_max_viento.tolist()}")

# fecha con la temperatura más alta registrada (48.1°C):
[timestamp("2011-01-25 00:00:00")]
# fecha con la temperatura más baja registrada (-8.5°C):
[timestamp("2009-06-11 00:00:00")]
# fecha con la mayor cantidad de lluvia registrada (371.6 mm):
[timestamp("2009-11-07 00:00:00")]
# fecha con el viento más extremo registrado (135.0 km/h):
[timestamp("2015-04-21 00:00:00"), timestamp("2011-02-03 00:00:00"), timestamp("2010-12-07 00:00:00")]

Variables Terminal 18:39 Python 3
```

La fecha con la sequía más larga de la historia:

```
Te damos la bienvenida a Colaboratory. No se pueden guardar cambios.
Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda
Comandos + Código + Texto Ejecutar todas Copiar en Drive

Archivos
+ Analiza tus archivos con código escrito por Gemini Subir
sample_data
media_por_fecha.csv
run:1749394804014-part-r-000...

# fecha con el viento más extremo registrado ({max_viento_valor}) km/h:\n{fechas_max_viento.tolist()}

# fecha con la temperatura más alta registrada (48.1°C):
[timestamp("2011-01-25 00:00:00")]
# fecha con la temperatura más baja registrada (-8.5°C):
[timestamp("2009-06-11 00:00:00")]
# fecha con la mayor cantidad de lluvia registrada (371.6 mm):
[timestamp("2009-11-07 00:00:00")]
# fecha con el viento más extremo registrado (135.0 km/h):
[timestamp("2015-04-21 00:00:00"), timestamp("2011-02-03 00:00:00"), timestamp("2010-12-07 00:00:00")]

# fecha de inicio y fin de la sequía más larga
df["lluvia_dia"] = df["rainfall"] > 0 & ~ df["temp"].isnull()
df["sequia_grupo"] = df["lluvia_dia"].shift().cumsum()
duracion_sequias = df[df["sequia_grupo"] == False].groupby("sequia_grupo").size()
max_sequia = duracion_sequias.max()
sequia_max_grupo = duracion_sequias.idxmax()

fecha_inicio_sequia = df[df["sequia_grupo"] == sequia_max_grupo]["date"].min()
fecha_fin_sequia = df[df["sequia_grupo"] == sequia_max_grupo]["date"].max()

print(f"Sequia más larga registrada: {max_sequia} días, desde {fecha_inicio_sequia} hasta {fecha_fin_sequia}")

Sequia más larga registrada: 481 días, desde 2015-01-06 00:00:00 hasta 2016-04-30 00:00:00

Variables Terminal 18:44 Python 3
```

Cantidad de días sin lluvia por año.

```
Te damos la bienvenida a Colaboratory. No se pueden guardar cambios.
Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda
Comandos + Código + Texto Ejecutar todas Copiar en Drive

Archivos
+ Analiza tus archivos con código escrito por Gemini Subir
sample_data
media_por_fecha.csv
run:1749394804014-part-r-000...

# fecha con el viento más extremo registrado ({max_viento_valor}) km/h:\n{fechas_max_viento.tolist()}

# fecha con la temperatura más alta registrada (48.1°C):
[timestamp("2011-01-25 00:00:00")]
# fecha con la temperatura más baja registrada (-8.5°C):
[timestamp("2009-06-11 00:00:00")]
# fecha con la mayor cantidad de lluvia registrada (371.6 mm):
[timestamp("2009-11-07 00:00:00")]
# fecha con el viento más extremo registrado (135.0 km/h):
[timestamp("2015-04-21 00:00:00"), timestamp("2011-02-03 00:00:00"), timestamp("2010-12-07 00:00:00")]

# fecha de inicio y fin de la sequía más larga
df["lluvia_dia"] = df["rainfall"] > 0 & ~ df["temp"].isnull()
df["sequia_grupo"] = df["lluvia_dia"].shift().cumsum()
duracion_sequias = df[df["sequia_grupo"] == False].groupby("sequia_grupo").size()
max_sequia = duracion_sequias.max()
sequia_max_grupo = duracion_sequias.idxmax()

fecha_inicio_sequia = df[df["sequia_grupo"] == sequia_max_grupo]["date"].min()
fecha_fin_sequia = df[df["sequia_grupo"] == sequia_max_grupo]["date"].max()

print(f"Sequia más larga registrada: {max_sequia} días, desde {fecha_inicio_sequia} hasta {fecha_fin_sequia}")

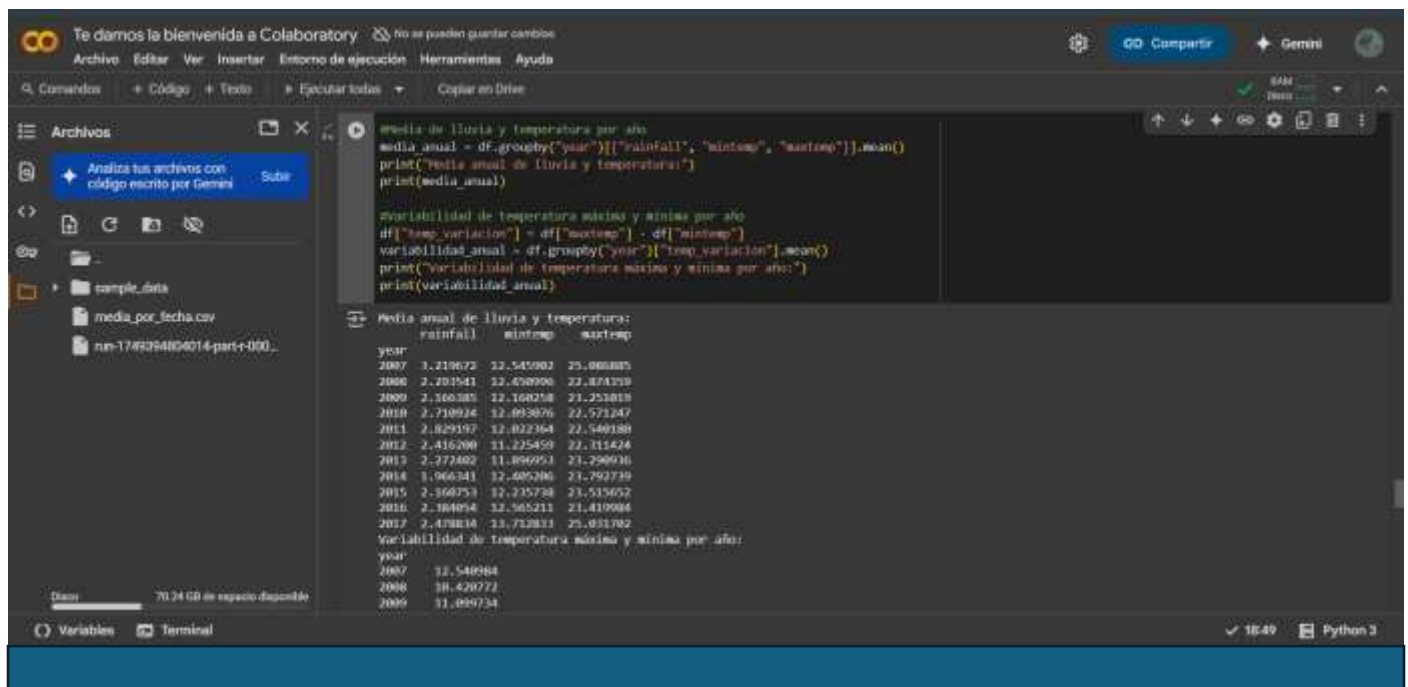
Sequia más larga registrada: 481 días, desde 2015-01-06 00:00:00 hasta 2016-04-30 00:00:00

# cantidad de días sin lluvia por año
dias_secos = df[df["rainfall"] == 0].groupby("year")["date"].nunique()
print("Cantidad de días sin lluvia por año:")
print(dias_secos)

Cantidad de días sin lluvia por año:
year
2007    51
2008   195
2009   365
2010   365
2011   159
2012   195
2013   127
2014   303
2015   363
2016   366
2017   176
Name: date, dtype: int64

Variables Terminal 18:47 Python 3
```

Media de lluvia y temperatura por año, y variabilidad de temperatura máxima y mínima por año.



The image shows a Colaboratory notebook with the following code and output:

```
#Media de lluvia y temperatura por año
media_anual = df.groupby("year")["rainfall", "mintemp", "maxtemp"].mean()
print("Media anual de lluvia y temperatura:")
print(media_anual)

#variabilidad de temperatura máxima y mínima por año
df["temp_variacion"] = df["maxtemp"] - df["mintemp"]
variabilidad_anual = df.groupby("year")["temp_variacion"].mean()
print("Variabilidad de temperatura máxima y mínima por año:")
print(variabilidad_anual)
```

Output:

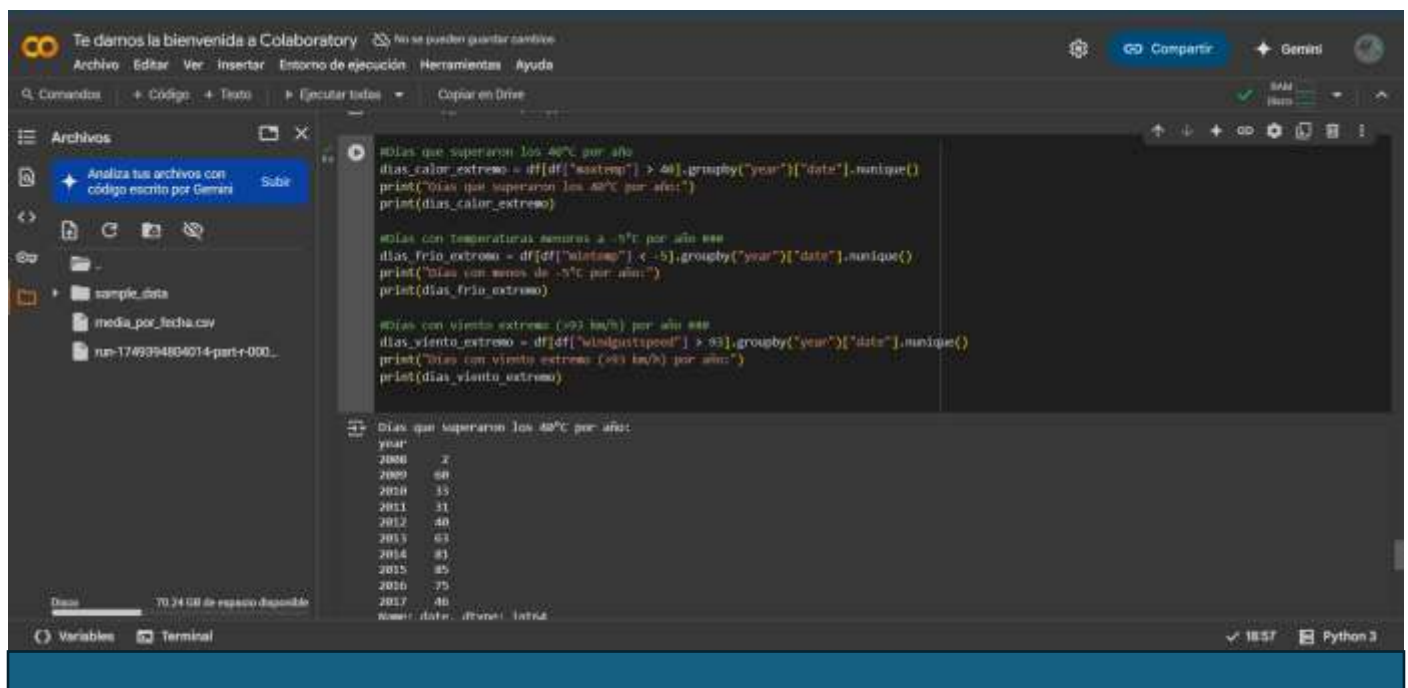
Media anual de lluvia y temperatura:

year	rainfall	mintemp	maxtemp
2007	3.210672	12.545902	25.908885
2008	2.201941	12.420906	23.373239
2009	2.360385	12.168258	23.253819
2010	2.710914	12.093876	23.571247
2011	2.829197	12.022364	23.540189
2012	2.436200	11.225459	22.211424
2013	2.272402	11.896953	23.290036
2014	1.966341	12.405206	23.792739
2015	2.360753	12.235728	23.515652
2016	2.384014	12.505211	23.439084
2017	2.478814	13.732833	25.931782

Variabilidad de temperatura máxima y mínima por año:

year	temp_variacion
2007	12.548984
2008	10.420772
2009	11.090734

Días que superaron los 40°C por año (temperatura extrema), temperaturas menores a -5°C por año, y viento extremo (>93 km/h) por año.



The image shows a Colaboratory notebook with the following code and output:

```
#Días que superaron los 40°C por año
dias_calor_extremo = df[df["maxtemp"] > 40].groupby("year")["date"].nunique()
print("Días que superaron los 40°C por año:")
print(dias_calor_extremo)

#Días con temperaturas menores a -5°C por año ***
dias_frio_extremo = df[df["mintemp"] < -5].groupby("year")["date"].nunique()
print("Días con menos de -5°C por año:")
print(dias_frio_extremo)

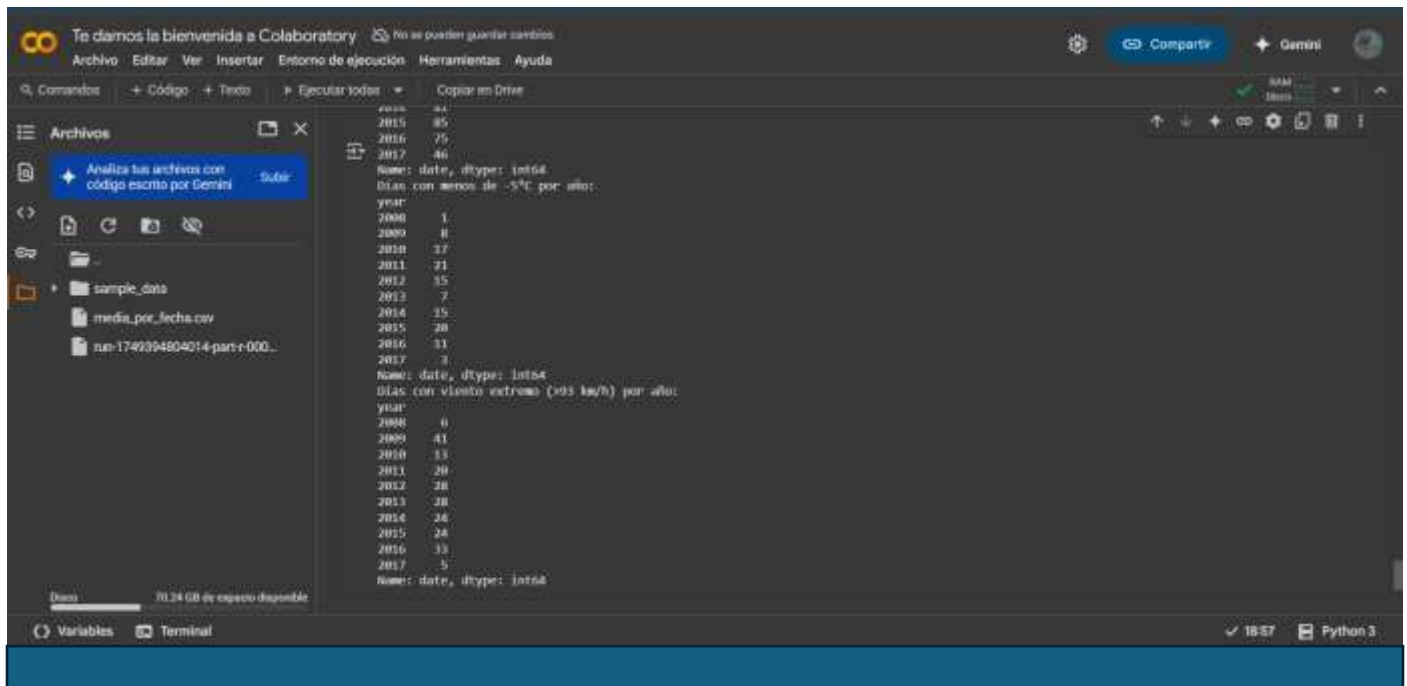
#Días con viento extremo (>93 km/h) por año ***
dias_viento_extremo = df[df["windgustspeed"] > 93].groupby("year")["date"].nunique()
print("Días con viento extremo (>93 km/h) por año:")
print(dias_viento_extremo)
```

Output:

Días que superaron los 40°C por año:

year	date
2008	2
2009	60
2010	15
2011	31
2012	40
2013	63
2014	81
2015	85
2016	75
2017	86

Names: date, df.index: index



Conclusiones

El análisis de datos climáticos en Australia entre 2008 y 2017 nos ha permitido observar patrones significativos en la variabilidad de temperaturas, precipitaciones y eventos extremos. Hemos identificado los días más cálidos, fríos y lluviosos del período, así como el día con vientos más intensos. También hemos determinado la duración de la sequía más larga registrada, evidenciando la recurrencia de períodos de escasez de lluvia.

Las medias anuales de lluvia y temperatura nos proporcionaron una perspectiva general del comportamiento climático en distintos años, ayudándonos a detectar posibles cambios en las tendencias de precipitaciones. La variabilidad de temperatura también mostró fluctuaciones considerables en las diferencias entre temperaturas máximas y mínimas, lo que podría tener implicaciones sobre la percepción térmica y el impacto en los ecosistemas.

Asimismo, los eventos climáticos extremos resaltan la importancia de monitorear el número de días con temperaturas superiores a 40°C y vientos extremos, factores que pueden influir en la vida cotidiana, la agricultura y la infraestructura del país.

Este análisis ha demostrado la utilidad del procesamiento de datos para comprender mejor la climatología y su evolución, estableciendo un marco para futuras investigaciones y posibles aplicaciones en la predicción de eventos climáticos.