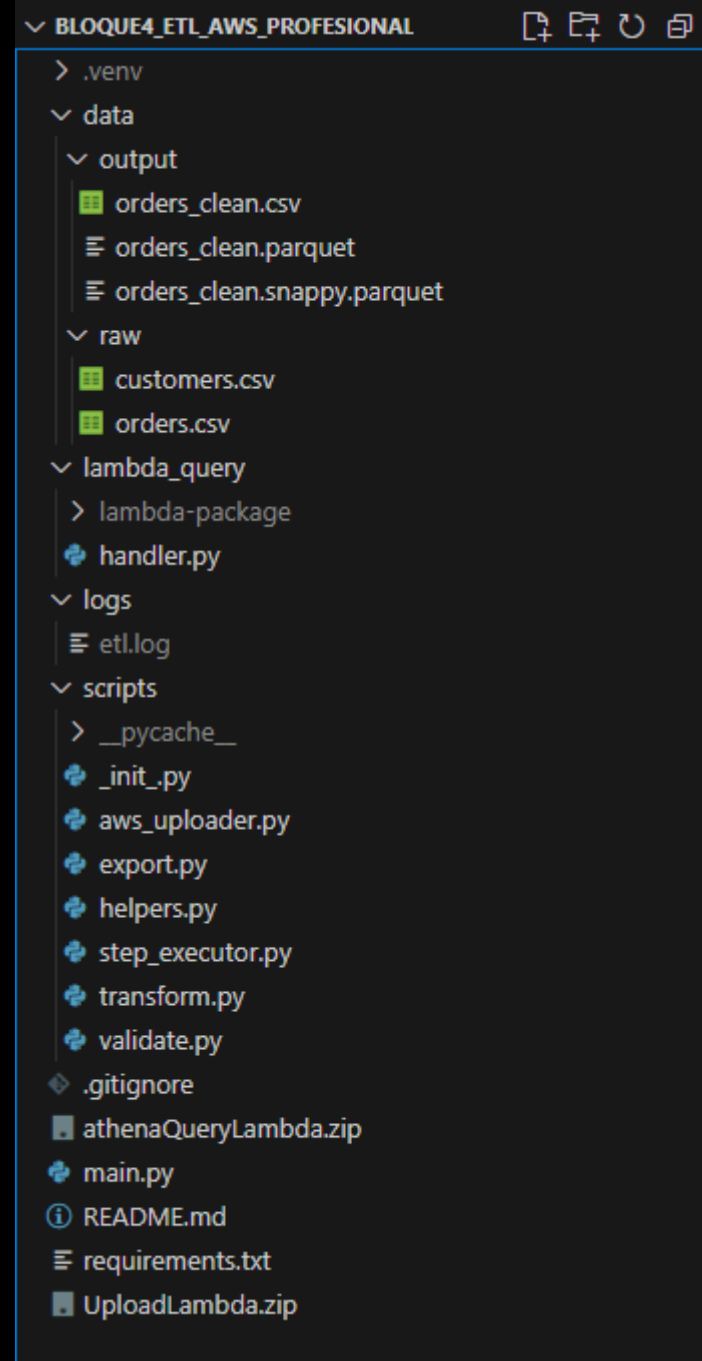


Estructura de carpetas del proyecto (VSC)



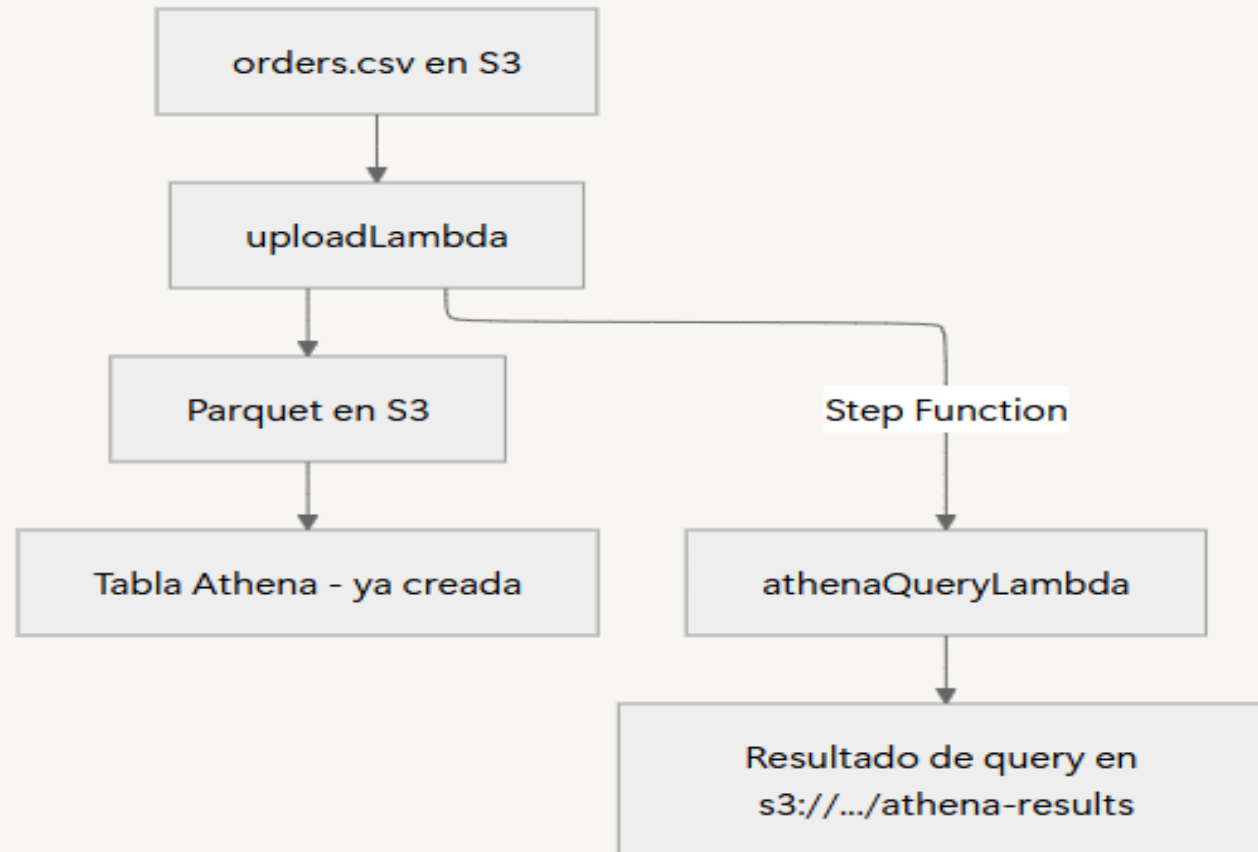
📁 Estructura del proyecto (local)

```
aws-etl-pipeline/
├── data/
│   ├── raw/
│   │   ├── orders.csv          # ✅ Input local
│   │   └── customers.csv
│   └── output/
│       ├── orders_clean.csv
│       ├── orders_clean.parquet
│       └── orders_clean.snappy.parquet
├── lambda_query/
│   ├── handler.py             # ✅ Lambda Athena
│   └── lambda-package/        # Dependencias ZIP
├── scripts/
│   ├── aws_uploader.py
│   ├── export.py
│   ├── helpers.py
│   ├── step_executor.py
│   ├── transform.py
│   └── validate.py
├── logs/
│   └── etl.log
├── UploadLambda.zip           # Lambda CSV → Parquet
├── athenaQueryLambda.zip      # Lambda Athena
├── main.py
├── requirements.txt
└── README.md
```

Flujo del Pipeline

graph TD

```
A[orders.csv en S3] --> B[uploadLambda]
B --> C[Parquet en S3]
C --> D[Tabla Athena - ya creada]
B -->|Step Function| E[athenaQueryLambda]
E --> F[Resultado de query en s3://.../athena-results]
```



Dataset de Entrada

orders.csv X

data > raw > orders.csv > data

	order_id	customer_id	order_date	quantity	unit_price	status
1	101	1	2023-07-10	3	150	completed
2	102	2	2023-07-11	1	1200	pending
3	103	3	2023-07-12	5	200	cancelled
4	104	4	2023-07-13	2	750	completed
5	105	5	2023-07-14	10	80	pending
6	106	6	2023-07-15	4	300	completed
7	107	7	2023-07-16	6	180	pending
8	108	8	2023-07-17	3	950	cancelled
9	109	9	2023-07-18	8	110	completed
10	110	10	2023-07-19	2	1000	pending
11	111	11	2023-07-20	1	2200	completed
12	112	12	2023-07-21	7	140	completed
13	113	13	2023-07-22	5	210	pending
14	114	14	2023-07-23	2	1300	cancelled
15	115	15	2023-07-24	9	95	completed
16	116	16	2023-07-25	3	400	completed
17	117	17	2023-07-26	4	275	pending
18	118	18	2023-07-27	1	850	cancelled
19	119	19	2023-07-28	6	160	completed
20	120	20	2023-07-29	2	1250	pending
21	121	21	2023-07-30	10	70	cancelled
22	122	22	2023-07-31	3	300	completed
23	123	23	2023-08-01	2	1450	pending
24	124	24	2023-08-02	4	220	cancelled
25	125	25	2023-08-03	5	195	completed
26	126	26	2023-08-04	3	180	pending

Lambda #1 - UploadLambda

UploadLambda

▼ Function overview [Info](#)

[Diagram](#)[Template](#)**UploadLambda**

Layers

(0)

[+ Add trigger](#)[+ Add destination](#)[Code](#)[Test](#)[Monitor](#)[Configuration](#)[Aliases](#)[Versions](#)

Code source [Info](#)



EXPLORER



lambda_function.py

handler.py X



UPLOADLAMBDA

> lambda-package



handler.py



DEPLOY

Deploy (Ctrl+Shift+U)

Test (Ctrl+Shift+I)

handler.py

```
1 import boto3
2
3 def lambda_handler(event, context):
4     athena = boto3.client("athena")
5     bucket = event["bucket"]
6
7     query = """
8         SELECT status, SUM(total_amount) AS total_sales
9         FROM default.orders_clean
10        WHERE status IN ('pending', 'returned', 'cancelled', 'shipped')
11        GROUP BY status
12    """
13
14    response = athena.start_query_execution(
15        QueryString=query,
16        QueryExecutionContext={"Database": "default"},
17        ResultConfiguration={"OutputLocation": f"s3://{bucket}/athena-results/"})
18
19
20    return {"QueryExecutionId": response["QueryExecutionId"]}
21
```

Exportación optimizada en formato Parquet (Snappy)

bloque4/

Objects Properties

Objects (1)



Copy S3 URI

Copy URL

Download

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to

Find objects by prefix

<input type="checkbox"/>	Name	Type	Last modified
<input type="checkbox"/>	 orders_clean.snappy.parquet	parquet	August 1, 2025, 12:52:45 (UTC-03:00)

3 Crear tabla externa en Athena

```
CREATE EXTERNAL TABLE IF NOT EXISTS default.orders_clean (  
    order_id BIGINT,  
    order_date DATE,  
    customer_id BIGINT,  
    product_id BIGINT,  
    product_name STRING,  
    quantity BIGINT,  
    unit_price DOUBLE,  
    currency STRING,  
    status STRING,  
    total_amount DOUBLE  
)  
STORED AS PARQUET  
LOCATION 's3://marcelo-orders-bucket/data/'  
TBLPROPERTIES ('parquet.compress'='SNAPPY');
```

Consulta SQL en Athena



[Amazon Athena](#) > Query editor

Editor

Recent queries

Saved queries

Settings



Athena now supports typeahead code suggestions to speed up SQL query development

Typeahead suggestions are turned on by default. You can change this setting in query editor preferences.

Data



Data source

AwsDataCatalog



Catalog

None



Database

default



Query 1 : ✕

✓ Query 2 : ✕

```
1 SELECT status, SUM(total_amount) AS total_sales
2 FROM default.orders_clean
3 WHERE status IN ('pending', 'returned', 'cancelled', 'shipped')
4 GROUP BY status
5
```


Resultados de la consulta en la nube

SQL Ln 1, Col 1

Run again

Explain [↗](#)

Cancel

Clear

Create ▼

Query results

Query stats

✓ Completed

Results (4)

🔍 Search rows


#	▼	status	▼	total_sales
1		returned		4419.64
2		cancelled		3409.68
3		shipped		4259.4199999999999
4		pending		4519.3600000000001


Lambda #2 - athenaQueryLambda

athenaQueryLambda

▼ Function overview [Info](#)

Diagram | Template

 athenaQueryLambda

 Layers (0)

+ Add trigger

+ Add destination

Code | Test | Monitor | Configuration | Aliases | Versions

Code source [Info](#)

← →

athenaQueryLambda

EXPLORER

▼ ATHENAQUERYLAMBDA

handler.py

DEPLOY

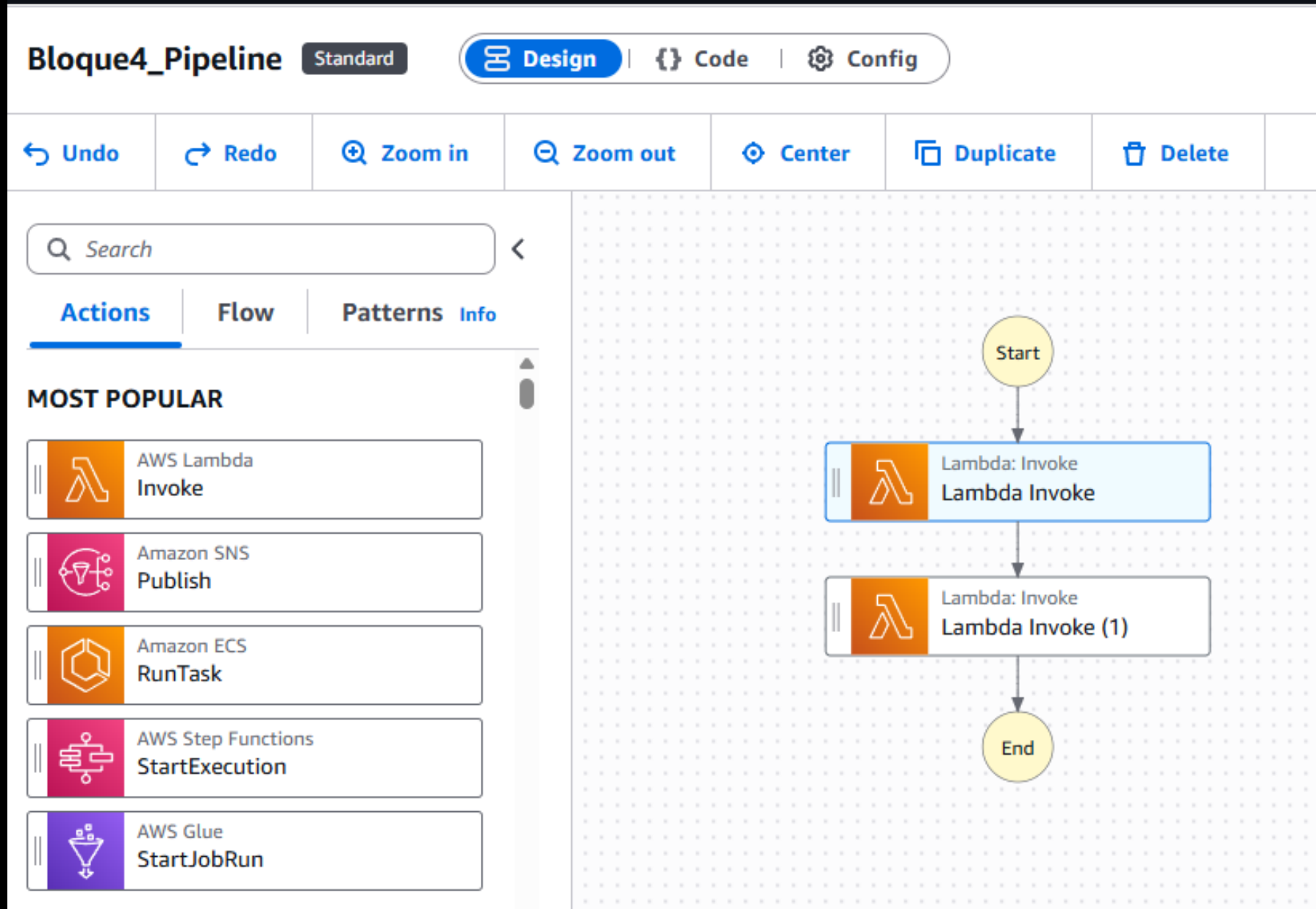
Deploy (Ctrl+Shift+U)

Test (Ctrl+Shift+I)

handler.py

```
1 import boto3
2
3 def lambda_handler(event, context):
4     athena = boto3.client("athena")
5     bucket = event["bucket"]
6
7     query = """
8         SELECT status, SUM(total_amount) AS total_sales
9         FROM default.orders_clean
10        WHERE status IN ('pending', 'returned', 'cancelled', 's
11        GROUP BY status
12    """
13
14    response = athena.start_query_execution(
15        QueryString=query,
16        QueryExecutionContext={"Database": "default"},
17        ResultConfiguration={"OutputLocation": f"s3://{bucket}/
18    )
19
20    return {"QueryExecutionId": response["QueryExecutionId"]}
21
```

Orquestación – Step Functions (diseño)



Ejecución del workflow (éxito)

DetailsExecution input and outputDefinition

Execution status

✓ Succeeded

Execution type

Standard

Execution ARN

[arn:aws:states:us-east-1:957178787439:execution:Bloque4_Pipeline:5ae656c6-9693-4789-8071-e545935de09e](#)

IAM role ARN

[arn:aws:iam::957178787439:role/service-role/StepFunctions-Bloque4_Pipeline-role-29ltojxu](#)

State transitions

[Learn more](#)

4

Start time

[Aug 9, 2025, 22:09:16.614 \(UTC-03:00\)](#)

End time

[Aug 9, 2025, 22:09:23.311 \(UTC-03:00\)](#)

Duration

00:00:06.697

Alias

-

Version

-

Graph viewTable view

Graph view

Actions

+

+

+

+

+

Start

AWS Lambda: Invoke

Lambda Invoke

✓

AWS Lambda: Invoke

Lambda Invoke (1)

✓

End

Step details

Choose a step to view its details.

Resultados en S3 – carpeta athena-results/

athena-results/

Copy S3 URI

Objects

Properties

Objects (10)

Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

< 1 >

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	29631abc-d681-473e-aa16-d27e79400d61.csv	csv	August 8, 2025, 12:22:46 (UTC-03:00)	126.0 B	Standard
<input type="checkbox"/>	29631abc-d681-473e-aa16-d27e79400d61.csv.metadata	metadata	August 8, 2025, 12:22:46 (UTC-03:00)	124.0 B	Standard
<input type="checkbox"/>	6ac31900-9a5d-4bb4-a5f5-2f90933a6a89.csv	csv	August 9, 2025, 22:09:24 (UTC-03:00)	126.0 B	Standard
<input type="checkbox"/>	6ac31900-9a5d-4bb4-a5f5-2f90933a6a89.csv.metadata	metadata	August 9, 2025, 22:09:24 (UTC-03:00)	124.0 B	Standard
<input type="checkbox"/>	c7e5b099-7167-4ec7-9635-f05984843862.csv	csv	August 7, 2025, 00:28:59 (UTC-03:00)	126.0 B	Standard
<input type="checkbox"/>	c7e5b099-7167-4ec7-9635-f05984843862.csv.metadata	metadata	August 7, 2025, 00:28:59 (UTC-03:00)	124.0 B	Standard
<input type="checkbox"/>	de4ea919-6869-41b2-bfa3-08fb909028f5.csv	csv	August 8, 2025, 12:37:36 (UTC-03:00)	126.0 B	Standard
<input type="checkbox"/>	de4ea919-6869-41b2-bfa3-08fb909028f5.csv.metadata	metadata	August 8, 2025, 12:37:36 (UTC-03:00)	124.0 B	Standard

Salida final – CSV de la consulta

	A	B	C
1	status	total_sales	
2	returned	4419.64	
3	cancelled	3409.68	
4	shipped	4259.42	
5	pending	4519.36	
6			