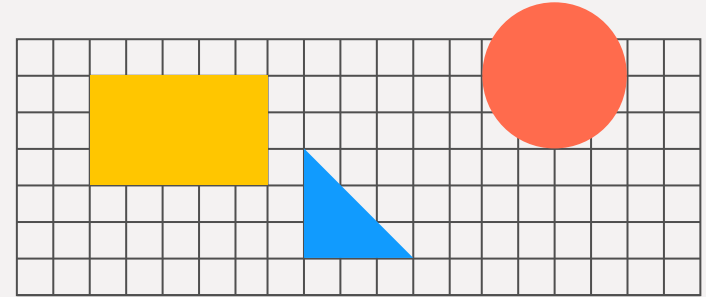


# Mini Project 1: AWS ETL Pipeline using S3, Lambda, RDS, and EC2

Team 1: Chu-Chun Ku, Flavien Foreste, Mmesoma Udensi, Yi-Ting Lee, Yun-Shan Chung



# Introduction

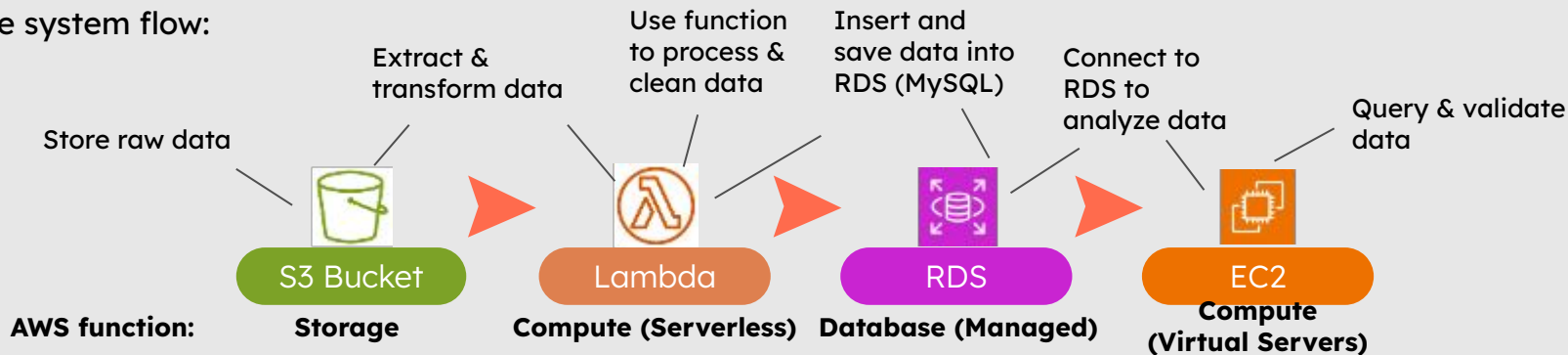
**Purpose:** Build an **ETL (Extract, Transform, Load) pipeline** to automatically process / clean raw data.

**Key Learning:**

- AWS service integration
- Building real-world cloud pipelines
- Solving practical issues (timeouts, memory, etc.)

Raw dataset: Review.csv (568,454 rows)

The system flow:



Setup Process

Challenges Faced

Final Results

# Step 1. Set Up Amazon S3

team1-raw [Info](#)

Objects

Metadata

Properties

Permissions

Metrics

Management

Access Points

## Objects (2)



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"/>	<a href="#">Reviews_small.csv</a>	csv	March 1, 2025, 21:43:07 (UTC-06:00)	30.5 MB	Standard
<input type="checkbox"/>	<a href="#">Reviews.csv</a>	csv	March 1, 2025, 21:52:58 (UTC-06:00)	287.0 MB	Standard

## Step 2. Set Up RDS MySQL instance

team1-rds



Modify

Actions ▾

### Summary

DB identifier  
team1-rds

Status  
Available

Role  
Instance

Engine  
MySQL  
Community

Recommendations

CPU  
0.00%

Class  
db.t4g.micro

Current activity

Region & AZ  
us-east-1c



Connectivity & security

Monitoring

Logs & events

Configuration

Zero-ETL integrations



### Connectivity & security

#### Endpoint & port

Endpoint

team1-rds.cts5bvbwlupa.  
us-east-1.rds.amazonaws.com

#### Networking

Availability Zone  
us-east-1c

VPC

#### Security

VPC security groups  
default (sg-  
01566b17d94ca389a)

### Security Groups (3) [Info](#)



Actions ▾

Export security groups to CSV











Find resources by attribute or tag

<input type="checkbox"/>	Name	Security group ID	Security group name	VPC ID	Description		
<input type="checkbox"/>	team1_db_sg	sg-01566b17d94ca389a	default	vpc-0d3248ca0e0b80bf5	default VPC security group		
<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-0c6100df3f1c3ded5	IPv4	MYSQL/Aurora	TCP	3306	0.0.0.0/0

## Step 3. Set Up EC2: Let MySQL connection in the EC2 terminal

Use an **EC2 instance** as a **client machine** to access and validate RDS.

<input checked="" type="checkbox"/>	Name 	Instance ID	Instance state 	Instance type 	Status check 
<input checked="" type="checkbox"/>	team1-ec2	i-06bbbfef7c1e6b2ea	 Running  	t2.micro	 2/2 checks passed

**Connect to RDS using MySQL CLI:**

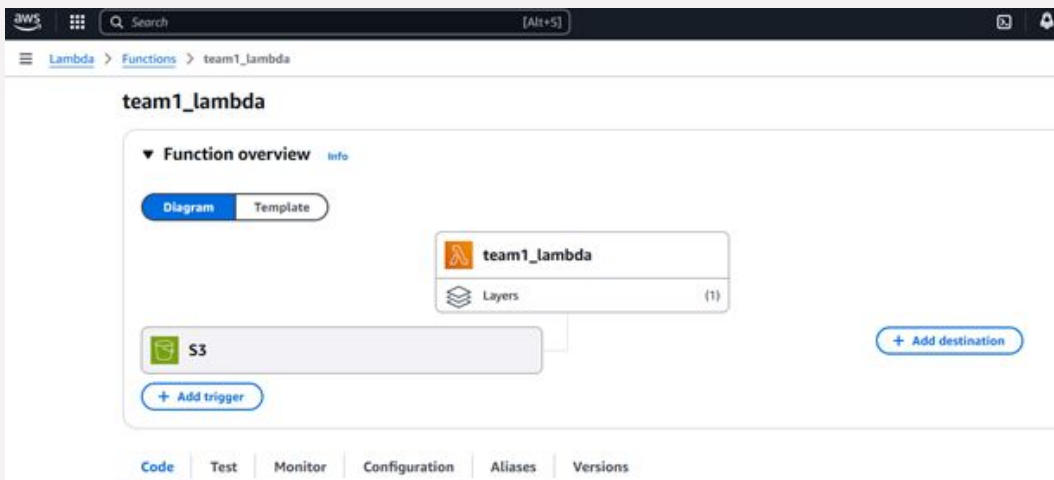
```
mysql -h team1-rds.cts5bvbwlupa.us-east-1.rds.amazonaws.com -u team1 -p
```

```
~ $ mysql -h team1-rds.cts5bvbwlupa.us-east-1.rds.amazonaws.com -u team1 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 32
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

## Step 4. Set Up AWS Lambda: Lambda function settings



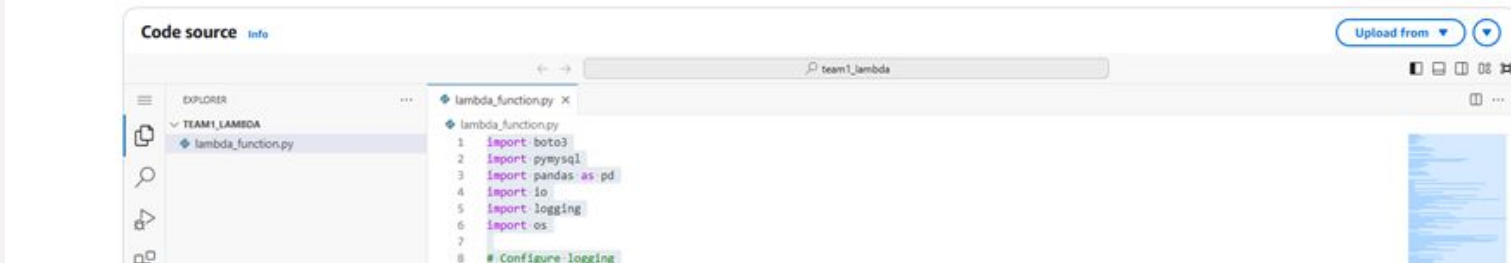
The screenshot shows the AWS Lambda console for the function 'team1\_lambda'. The 'Function overview' section is active, displaying a 'Diagram' tab with a visual representation of the function's dependencies, including an S3 bucket. Below the diagram, there is a '+ Add trigger' button. The 'Layers' section shows one layer is attached. At the bottom, there are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'.

```
14
15 # Database connection details (Replace these with actual RDS details)
16 DB_HOST = "team1-rds.cts5bvbwlypa.us-east-1.rds.amazonaws.com"
17 DB_USER = "team1"
18 DB_PASSWORD = "10430380"
19 DB_NAME = "reviews"
20 DB_PORT = 3306
21
22 # S3 bucket details
23 S3_BUCKET = "team1-raw"
24 INPUT_FILE = "Reviews.csv"

Last modified
10 minutes ago

Function ARN
arn:aws:lambda:us-east-1:056498985289:function:team1_lambda

Function URL
Info
```



The screenshot shows the 'Code source' tab for the 'team1\_lambda' function. The 'EXPLORER' pane on the left shows the file structure for 'TEAM1\_LAMBDA' with 'lambda\_function.py' selected. The main editor displays the code for 'lambda\_function.py', which includes imports for boto3, pymysql, pandas, io, logging, and os, followed by a comment '# Configure logging'.

```

aws  [Option+S] Q Search [Option+S]
Last login: Tue Mar 4 02:20:55 2025 from ec2-18-206-107-28.compute-1.amazonaws.com
#
# Amazon Linux 2
#
# AL2 End of Life is 2026-06-30.
#
# A newer version of Amazon Linux is available!
#
# Amazon Linux 2023, GA and supported until 2028-03-15.
# https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-86-40 ~]$ mysql -h team1-rdss.cawkb0h9h3cp.us-east-1.rds.amazonaws.com -u team1 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 611
Server version: 8.0.40 Source Distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> USE reviews;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews;
+-----+
| COUNT(*) |
+-----+
| 568427 |
+-----+
1 row in set (3.73 sec)

MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews WHERE score = 5;
+-----+
| COUNT(*) |
+-----+
| 363122 |
+-----+
1 row in set (2.97 sec)

MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews WHERE score BETWEEN 2 AND 4;
+-----+
| COUNT(*) |
+-----+
| 153037 |
+-----+
1 row in set (3.71 sec)

MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews WHERE score = 1;
+-----+
| COUNT(*) |
+-----+
| 52268 |
+-----+
1 row in set (2.98 sec)

```

# Challenge Faced

## Step 6 - Set Up AWS Lambda: Lambda troubleshooting

Before: setting memory allocation at 1024mb; timeout at 8mins

▼ 2025-03-04T05:09:49.519Z REPORT RequestId: 37325474-add1-4a13-bdce-5138bd0bbd24 Duration: 480000.00 ms Billed Duratio...

REPORT RequestId: 37325474-add1-4a13-bdce-5138bd0bbd24 Duration: 480000.00 ms Billed Duration: 480000 ms  
Memory Size: 1024 MB Max Memory Used: 1022 MB Init Duration: 2933.00 ms Status: timeout

After: setting memory allocation at 2048mb; timeout at 10mins

▼ 2025-03-04T06:45:27.253Z REPORT RequestId: 6a05a17f-0759-40ef-b7b2-54f15cc57961 Duration: 89914.73 ms Billed Duration...

REPORT RequestId: 6a05a17f-0759-40ef-b7b2-54f15cc57961 Duration: 89914.73 ms Billed Duration: 89915 ms  
Memory Size: 2048 MB Max Memory Used: 1171 MB Init Duration: 2532.63 ms



# Final Results



```
MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews;
+-----+
| COUNT(*) |
+-----+
| 568427 |
+-----+
1 row in set (3.73 sec)

MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews WHERE score = 5;
+-----+
| COUNT(*) |
+-----+
| 363122 |
+-----+
1 row in set (2.97 sec)

MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews WHERE score BETWEEN 2 AND 4;
+-----+
| COUNT(*) |
+-----+
| 153037 |
+-----+
1 row in set (3.71 sec)

MySQL [reviews]> SELECT COUNT(*) FROM customer_reviews WHERE score = 1;
^ [+-----+
| COUNT(*) |
+-----+
| 52268 |
+-----+
1 row in set (2.98 sec)
```