

UNIT 5.2 GRADED ASSIGNMENT

Group members

Ifra Saleem (2303.khi.deg.003)

Umaina Siddiqui (2023.KHI.DEG.033)

UNIT 5.2 GRADED ASSIGNMENT

Task:

Using the earnings CSV as a base, prepare a new data file with employees' office locations. Make sure there are 5-6 distinct locations that are shared between employees.

Create a Glue job that aggregates the data based on the office location to calculate average salaries and raise percentages for these locations.

Solution:

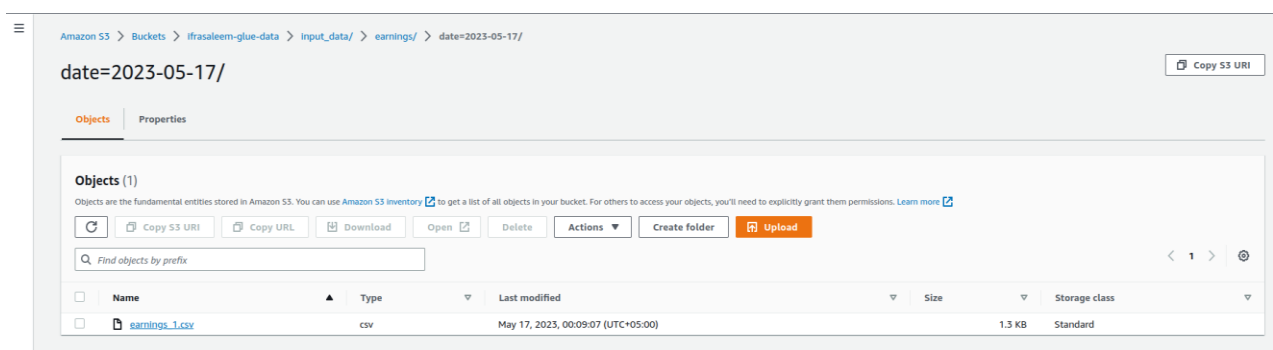
Input data:

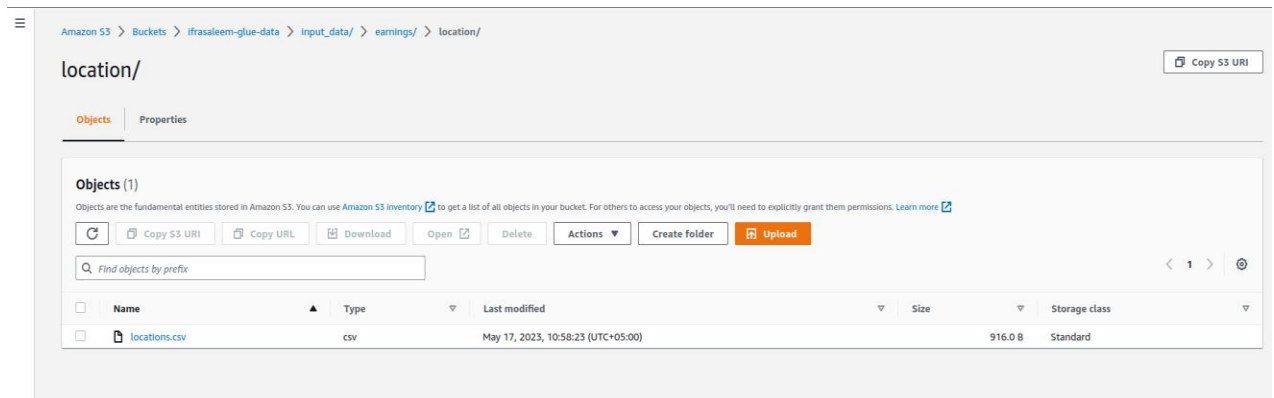
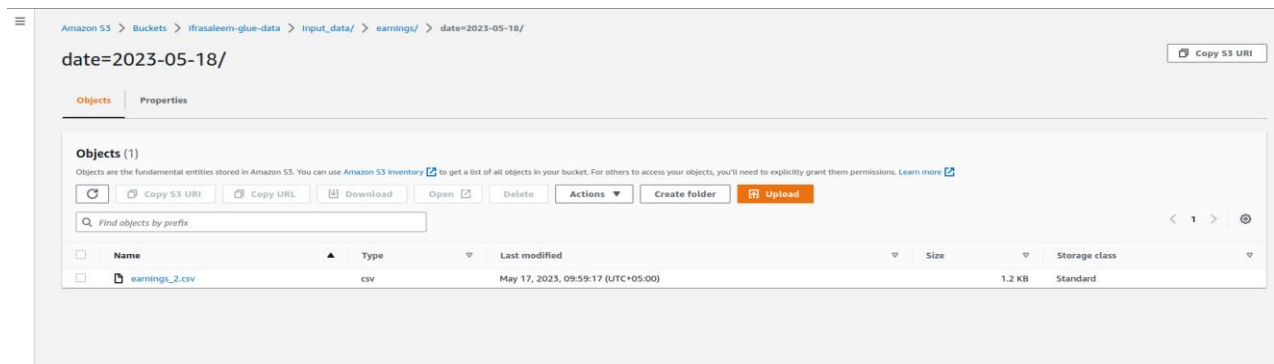
Here I upload three csv files to different folders of the bucket.

S3 bucket:

S3 bucket is used to retrieve or store large amounts of data in the form of objects within containers which is known as the bucket in AWS. It is used as backup or restore.

- So, in this task we created different folders inside that bucket and in the input data folder we created another folder earnings upload the csv files inside the folders inside the earning folder. So basically, we created the bucket to store our data and then we can use that data to create crawlers and jobs.





RDS:

RDS is a managed database service provided by AWS. Through RDS we can easily setup, operate and manage databases. It supports various database engines, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and Microsoft SQL Server. It supports various database engines, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and Microsoft SQL Server.

- So here we created a database, and we can use the credentials of this database to populate the database. We used the credentials in populate_db.py file and gave the values of username, password and endpoint URL.

Amazon RDS

Dashboard

Databases

Query Editor

Performance insights

Snapshots

Exports in Amazon S3

Automated backups

Reserved instances

Proxies

Subnet groups

Parameter groups

Option groups

Custom engine versions

Events

Event subscriptions

Recommendations

Certificate update

RDS > Databases > ifrasaleem-employees-db

ifrasaleem-employees-db

Modify Actions

Summary

DB identifier ifrasaleem-employees-db	CPU 3.50%	Status Available	Class db.t3.micro
Role Instance	Current activity 0 Connections	Engine PostgreSQL	Region & AZ us-east-1b

Connectivity & security

Monitoring

Logs & events

Configuration

Maintenance & backups

Tags

Connectivity & security

<div>Endpoint & port</div> <div>Endpoint ifrasaleem-employees-db.cjmrmsz5azdy.us-east-1.rds.amazonaws.com</div> <div>Port 5432</div>	<div>Networking</div> <div>Availability Zone us-east-1b</div> <div>VPC vpc-011f1c14c6aa80284</div> <div>Subnet group default-vpc-011f1c14c6aa80284</div>	<div>Security</div> <div>VPC security groups default (sg-01857ae9f2b7affcd)</div> <div>Active</div> <div>Publicly accessible Yes</div> <div>Certificate authority rds-ca-2019</div>
--	--	---

IAM Roles:

IAM roles are used to give permission to users, services or applications.

- We created an IAM role to use it while creating the glue job as it gives permission to services. And we used it while creating the crawlers and glue job.

Identity and Access Management (IAM)

Q Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access analyzer

Archive rules

Analyzers

Settings

Credential report

Organization activity

Service control policies (SCPs)

IAM > Roles > ifrasaleem-glue-role

ifrasaleem-glue-role

Delete

Allows Glue to call AWS services on your behalf.

Summary

Edit

Creation date
May 17, 2023, 00:25 (UTC+05:00)

Last activity
5 hours ago

ARN
arn:aws:iam::543470447653:role/ifrasaleem-glue-role

Maximum session duration
1 hour

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (2)

You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter.

1

	Policy name	Type	Attached entities	Description	Creation time	Edited time
<input type="checkbox"/>	AmazonS3FullAccess	AWS managed	1	Provides full access to all buckets via t...	8 years ago	1 year ago
<input type="checkbox"/>	AWSGlueServiceRole	AWS managed	1	Policy for AWS Glue service role which ...	5 years ago	4 years ago

VPC Endpoint:

VPC endpoints are used to privately connect virtual VPC to support AWS services and VPC endpoint services without the need for internet gateways. We can access AWS services by using private IP addresses.

- We created VPC endpoint so that we can use it in our script file and populate the database. It is basically connecting the data present locally with the RDS database.

VPC dashboard

EC2 Global View

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

Peering connections

Security

Network ACLs

Security groups

DNS firewall

VPC > Endpoints > vpce-02dd14eb90539884c

vpce-02dd14eb90539884c

Actions

Details

Endpoint ID: vpce-02dd14eb90539884c

Status: Available

Creation time: Wednesday, May 17, 2023 at 24:31:05 GMT+5

Endpoint type: Gateway

VPC ID: vpc-011f1c14c6aa80284

Status message: -

Service name: com.amazonaws.us-east-1.s3

Private DNS names enabled: No

Route tables

Policy

Tags

Route tables (1)

Find resources by attribute or tag

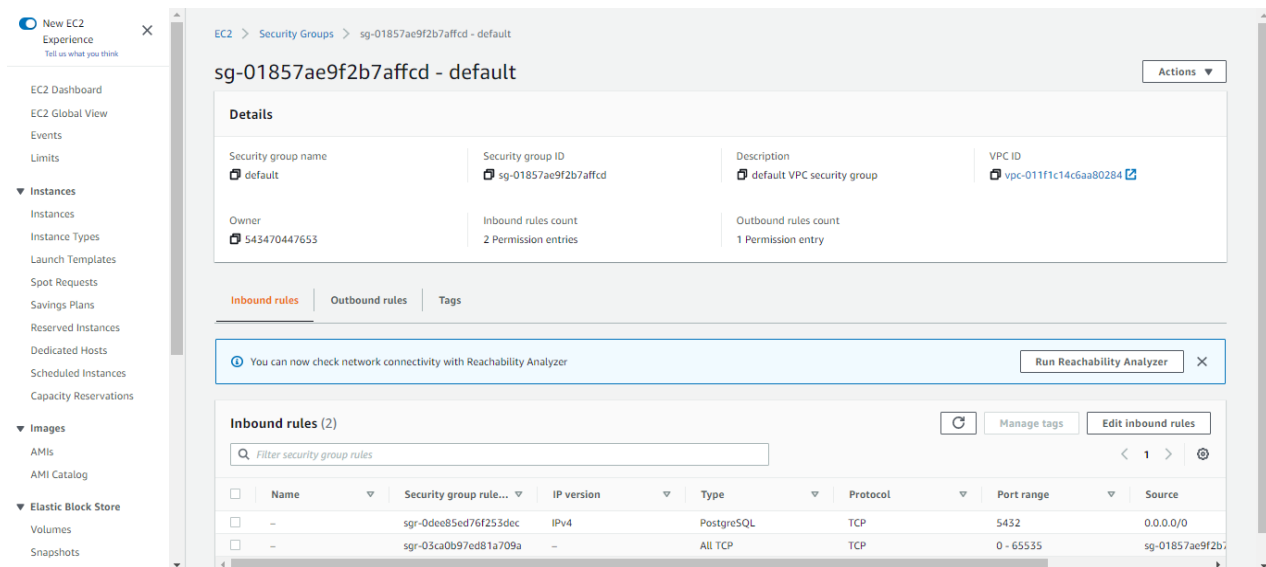
Name	Route Table ID	Main	Associated Id
-	rtb-0af873a7cb5216709	Yes	6 subnets

Security group rules:

It allows rules to define the types of traffic that are permitted to reach your instances. So, using these security group rules, we can allow or block any kind of traffic from any source.

- After creating the RDS database we define two inbound rules to give Glue Crawler access to RDS DB and to give access to RDS DB from our python script.

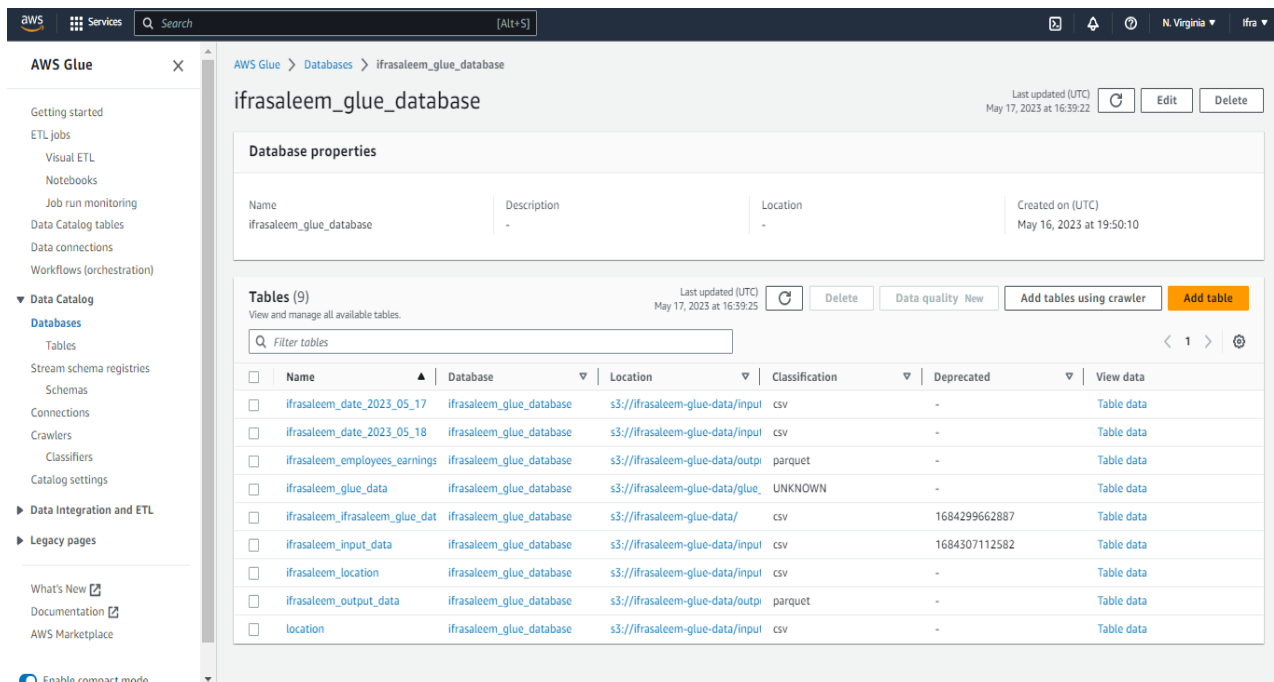
Security group rules (3)			
Filter by Security group rules			
Security group	Type	Rule	
default (sg-01857ae9f2b7affcd)	EC2 Security Group - Inbound	sg-01857ae9f2b7affcd	
default (sg-01857ae9f2b7affcd)	CIDR/IP - Inbound	0.0.0.0/0	
default (sg-01857ae9f2b7affcd)	CIDR/IP - Outbound	0.0.0.0/0	



Data Catalog Database:

Data Catalog database stores metadata information of various data assets. It includes data source location, data formats, schema definitions etc.

- We created Data Catalog Database so that we can use it while creating S3 crawler as it will provide service to store metadata information. And we selected it as target database in S3 crawler as it will save all the necessary metadata.



Glue Crawlers:

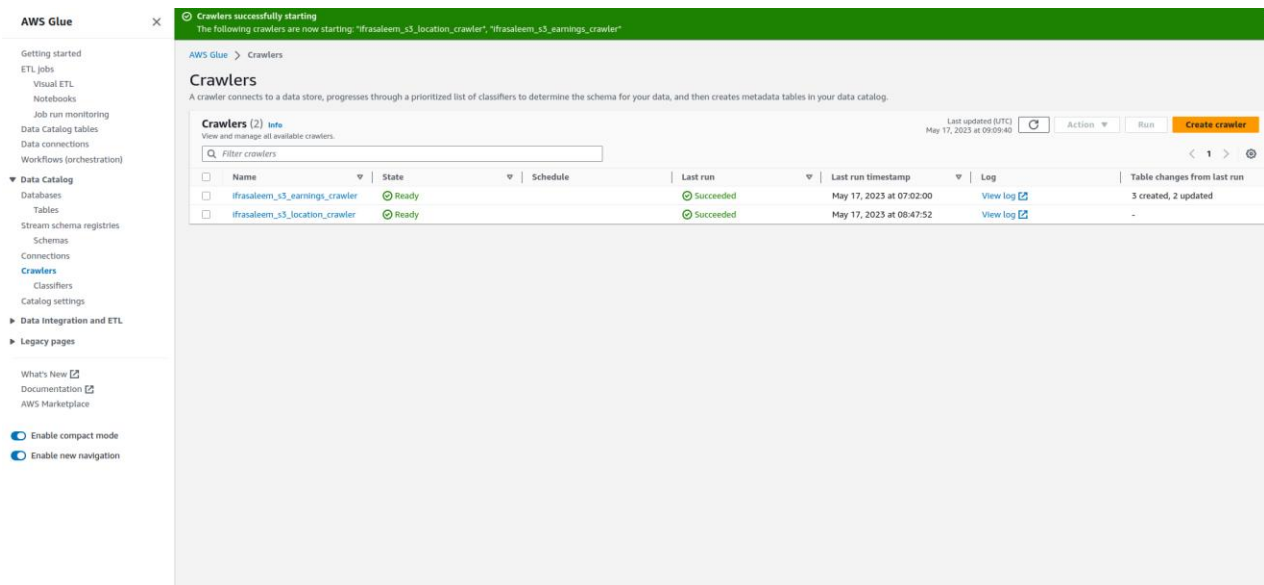
It is a fully managed ETL service. It automatically discovers and catalog metadata about data assets in various data stores. They analyze the data to infer its schema, structure, and format, and then create corresponding metadata tables in the AWS Glue Data Catalog.

- We created two glue crawlers. One crawler is used to analyze and manage earnings data and the other is used for locations data.

S3 Crawler:

AWS Glue Crawlers can be used to crawl data stored in Amazon S3 buckets. It analyzes and processes the data stored in s3 bucket using AWS glue and other AWS services.

- We used S3 data source as it used to discover and catalog metadata about the data stored in your Amazon S3 buckets.



Glue Job:

It is also used to manage ETL service. It is used to create and run ETL workflows to process and transform data stored in various data sources, including amazon S3, relational database and data warehouses.

- We created a glue job to build a schema and to process the data source. We joined the earnings data and locations data and then wrote a SQL query to calculate the average earnings and raise percentage according to the locations.

AWS Glue

Getting started

ETL jobs

Visual ETL

Notebooks

Job run monitoring

Data Catalog tables

Data connections

Workflows (orchestration)

Data Catalog

Databases

Tables

Stream schema registries

Schemas

Connections

Crawlers

Classifiers

Catalog settings

Data Integration and ETL

Legacy pages

What's New

Documentation

AWS Marketplace

Enable compact mode

Enable new navigation

ifrasaleem_employee_earnings_job

Last modified on 5/17/2023, 2:07:17 PM

Try new UI

End session

Actions

Save

Run

Visual

Script

Job details

Runs

Data quality

Schedules

Version Control

Source

Action

Target

Undo

Redo

Remove

Transform

Output schema

Data preview

Data source - S3 bucket

Amazon S3

Data source - S3 bucket

Amazon S3

Transform - Join

Join

Transform - SQL Query

SQL Query

Data target - S3 bucket

Amazon S3

Name

SQL Query

Node parents

Choose which nodes will provide inputs for this one.

Choose one or more parent nodes

Join

Join - Transform

Associate an alias with each input source

Input sources

Join

SQL aliases

myDataSource

SQL query

1 SELECT

2 location,

3 avg(earnings) AS average_earnings,

4 (avg(earnings) - min(earnings)) / min(earnings) * 100 AS raise_percentage

5 FROM

6 myDataSource

7 GROUP BY

8 location;

Job runs:

AWS Glue

Getting started

ETL jobs

Visual ETL

Notebooks

Job run monitoring

Data Catalog tables

Data connections

Workflows (orchestration)

Data Catalog

Databases

Tables

Stream schema registries

Schemas

Connections

Crawlers

Classifiers

Catalog settings

Data Integration and ETL

Legacy pages

What's New

Documentation

AWS Marketplace

Enable compact mode

Enable new navigation

ifrasaleem_employee_earnings_job

Last modified on 5/17/2023, 2:07:17 PM

Try new UI

End session

Actions

Save

Run

Visual

Script

Job details

Runs

Data quality

Schedules

Version Control

Job runs (1/14)

Filter job runs by property

Run status

Retry

Start time

End time

Duration

Capacity

Worker type

Glue version

05/17/2023 14:04:52

Job name

ifrasaleem_employee_earnings_job

Id

j_r_49w7354428c2409cd7961cc227d3870e8192c8f459a2f1fee

Run status

Succeeded

Glue version

3.0

Retry attempt number

Initial run

Start time

May 17, 2023 2:04:52 PM

End time

May 17, 2023 2:07:23 PM

Start-up time

53 seconds

Execution time

Last modified on

Trigger name

Security configuration

Query output:

AWS Glue

Getting started

ETL jobs

Visual ETL

Notebooks

Job run monitoring

Data Catalog tables

Data connections

Workflows (orchestration)

Data Catalog

Databases

Tables

Stream schema registries

Schemas

Connections

Crawlers

Classifiers

Catalog settings

Data Integration and ETL

Legacy pages

What's New

Documentation

AWS Marketplace

Enable compact mode

Enable new navigation

ifrasaleem_employee_earnings_job

Last modified on 5/17/2023, 5:39:30 PM

Try new UI

End session

Actions

Save

Run

Visual

Script

Job details

Runs

Data quality

New

Schedules

Version Control

Source

Action

Target

Undo

Redo

Remove

Search

Zoom

Fullscreen

Data source - S3 bucket

Amazon S3

✓

Data source - S3 bucket

Amazon S3

✓

Transform - Join

Join

✓

Transform - SQL Query

SQL Query

✓

Data target - S3 bucket

Amazon S3

✓

Transform

Output schema

Data preview

Data preview (5)

info

Previewing 3 of 3 fields

Filter sample dataset

location	average_earnings	raise_percentage
B	6473.346153046154	162.71697052947053
C	5391.681818181818	122.1541746263625
A	5993.6	194.81554353172652
D	5678.72	174.86544046466602
E	5699.818181818182	163.59270710804908

Output folder in the bucket:

Amazon S3

Buckets

ifrasaleem-glue-data

output_data/

employees_earnings/

location/

Copy S3 URI

location/

Objects

Properties

Objects (20)

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](#)

Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

Find objects by prefix

Name	Type	Last modified	Size	Storage class
run-1684309233893-part-block-0-r-00001-snappy.parquet	parquet	May 17, 2023, 12:40:35 (UTC+05:00)	608.0 B	Standard
run-1684309233893-part-block-0-r-00002-snappy.parquet	parquet	May 17, 2023, 12:40:35 (UTC+05:00)	599.0 B	Standard
run-1684309233893-part-block-0-r-00003-snappy.parquet	parquet	May 17, 2023, 12:40:35 (UTC+05:00)	599.0 B	Standard
run-1684309233893-part-block-0-r-00006-snappy.parquet	parquet	May 17, 2023, 12:40:35 (UTC+05:00)	599.0 B	Standard
run-1684310250811-part-block-0-r-00001-snappy.parquet	parquet	May 17, 2023, 12:57:32 (UTC+05:00)	596.0 B	Standard
run-1684310250811-part-block-0-r-00002-snappy.parquet	parquet	May 17, 2023, 12:57:32 (UTC+05:00)	587.0 B	Standard
run-1684310250811-part-block-0-r-00003-snappy.parquet	parquet	May 17, 2023, 12:57:32 (UTC+05:00)	587.0 B	Standard
run-1684310250811-part-block-0-r-00006-snappy.parquet	parquet	May 17, 2023, 12:57:32 (UTC+05:00)	587.0 B	Standard
run-1684312622008-part-block-0-r-00001-snappy.parquet	parquet	May 17, 2023, 13:37:05 (UTC+05:00)	608.0 B	Standard
run-1684312622008-part-block-0-r-00002-snappy.parquet	parquet	May 17, 2023, 13:37:05 (UTC+05:00)	599.0 B	Standard
run-1684312622008-part-block-0-r-00003-snappy.parquet	parquet	May 17, 2023, 13:37:05 (UTC+05:00)	599.0 B	Standard
run-1684312622008-part-block-0-r-00006-snappy.parquet	parquet	May 17, 2023, 13:37:05 (UTC+05:00)	599.0 B	Standard
run-1684312704206-part-block-0-r-00001-snappy.parquet	parquet	May 17, 2023, 13:38:26 (UTC+05:00)	608.0 B	Standard
run-1684312704206-part-block-0-r-00002-snappy.parquet	parquet	May 17, 2023, 13:38:26 (UTC+05:00)	599.0 B	Standard
run-1684312704206-part-block-0-r-00003-snappy.parquet	parquet	May 17, 2023, 13:38:26 (UTC+05:00)	599.0 B	Standard
run-1684312704206-part-block-0-r-00006-snappy.parquet	parquet	May 17, 2023, 13:38:26 (UTC+05:00)	599.0 B	Standard