

End to End Data Analytics Pipeline on AWS with S3, Glue, Athena & QuickSight

Introduction

In today's data-driven world, efficient data processing, storage, and analysis are essential for extracting insights and enabling informed decision-making. This project showcases a complete **End to End Data Analytics pipeline** using core AWS services — **Amazon S3**, **AWS Glue**, **Amazon Athena**, and **Amazon QuickSight** — to handle the full data lifecycle, from raw ingestion to rich visual analysis.

The project leverages the Data Science Job Salaries dataset from Kaggle ([link](#)), which offers real-world insights into salary trends and compensation structures within the data science profession.

The primary objective is to build a scalable, serverless data pipeline that:

- Ingests raw data into Amazon S3
- Queries the data interactively using Amazon Athena
- Transforms and processes data through AWS Glue ETL
- Visualizes the results with Amazon QuickSight dashboards

Part 1:

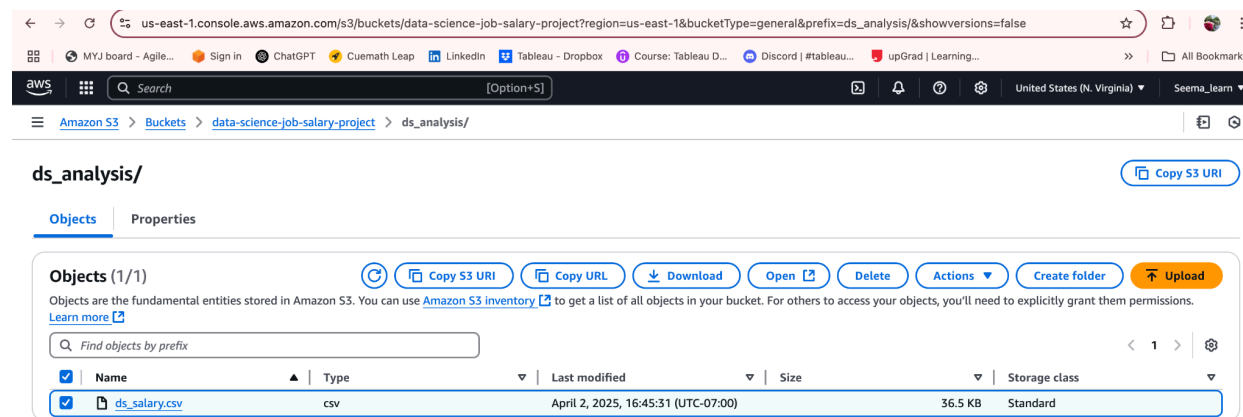
CSV File Uploaded to S3 Bucket:

- **Objective:** The "**ds_salary.csv**" file was uploaded to S3 for data analysis, enabling efficient storage, retrieval, and processing within AWS services.
- **Source and Destination:**
 - **Source:** The dataset was obtained from Kaggle ([Data Science Job Salaries](#)) and uploaded from the local system.
 - **Destination:** The file was uploaded to the S3 bucket at **s3://data-science-job-salary-project/ds_analysis/ds_salary.csv**

- **Steps Taken:**

- The file was manually uploaded to the S3 bucket.
- The column header was removed prior to uploading.

S3 console screenshot:



- **S3 URI:**

s3://data-science-job-salary-project/ds_analysis/ds_salary.csv

Part 2:

SQL Query Documentation for Athena-based Data Analysis

Introduction:

- **Purpose of the analysis:** This document outlines the SQL queries executed in AWS Athena to perform data analysis on the "ds_salary.csv" dataset, which was uploaded to an S3 bucket. The analysis aimed to derive insights on salary trends based on job roles, experience, and location.
- **Tools Used:** AWS Athena, SQL, S3, etc.

Data Source:

- **Dataset:** "ds_salary.csv" from Kaggle, uploaded to an S3 bucket.

- **S3 Bucket:**

s3://data-science-job-salary-project/ds_analysis/ds_salary.csv

SQL Query Documentation:

Database & Table Creation

The screenshot displays a SQL query editor interface. At the top, there are tabs for 'Query 4', 'Query 5', and 'datascience_analysi...'. The active tab 'Query 4' shows a single line of SQL code: `create database datascience_analysis;`. Below this, a larger window shows a more complex SQL query for creating an external table. The query is as follows:

```
1 CREATE EXTERNAL TABLE IF NOT EXISTS `datascience_analysis`.`datascience_table` (  
2   `id` int,  
3   `work_year` int,  
4   `experience_level` string,  
5   `employment_type` string,  
6   `job_title` string,  
7   `salary` float,  
8   `salary_currency` string,  
9   `salary_in_usd` float,  
10  `employee_residence` string,  
11  `remote_ratio` int,  
12  `company_location` string,  
13  `company_size` string  
14 )  
15 ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe'  
16 WITH SERDEPROPERTIES ('field.delim' = ',')  
17 STORED AS INPUTFORMAT 'org.apache.hadoop.mapred.TextInputFormat' OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io  
   .HiveIgnoreKeyTextOutputFormat'  
18 LOCATION 's3://data-science-job-salary-project/ds_analysis/'  
19 TBLPROPERTIES ('classification' = 'csv');
```

Below the query editor, there is a status bar indicating 'SQL Ln 19, Col 42'. To the right of the status bar are icons for running, explaining, and creating the query. Below the status bar, there are buttons for 'Run again', 'Explain', 'Cancel', 'Clear', and 'Create'. To the right of these buttons is a toggle for 'Reuse query results up to 60 minutes ago'. Below the buttons, there are tabs for 'Query results' and 'Query stats'. The 'Query results' tab is active and shows a green bar indicating 'Completed'. To the right of the green bar, the following statistics are displayed: 'Time in queue: 45 ms', 'Run time: 398 ms', and 'Data scanned: -'.

Query to count total records:

Query 4 : X

Query 5 : X

datascience_analysi... : X

```
1 #Query to count total records
2 SELECT count(*) as total_records
3 from datascience_analysis.datascience_table;
4
```

SQL Ln 16, Col 1

Run

Explain

Cancel

Clear

Create

Reuse query results
up to 60 minutes ago

Query results | Query stats

Completed Time in queue: 109 ms Run time: 359 ms Data scanned: 36.54 KB

Results (1)

Copy

Download results CSV

Search rows

< 1 >

#	total_records
1	607

Query to Get Average Salary by Job Title:

23 #Query to Get Average Salary by Job Title:

24 SELECT job_title,

25 avg(salary) as averge_salary

26 from datascience_analysis.datascience_table

27 group by job_title

28 order by avg(salary) desc;

SQL Ln 24, Col 1

Run again

Explain

Cancel

Clear

Create

Reuse query results
up to 60 minutes ago

Query results | Query stats

Completed Time in queue: 68 ms Run time: 776 ms Data scanned: 36.54 KB

Results (50)

Copy

Download results CSV

Search rows

< 1 >

#	job_title	averge_salary
1	Head of Machine Learning	6000000.0
2	ML Engineer	2676666.8
3	BI Data Analyst	1902045.4
4	Lead Data Scientist	1101666.6
5	Data Science Manager	1062598.6

Query to Find the Highest Salary by Experience Level:

12 #Query to Find the Highest Salary by Experience Level

13 select experience_level,

14 max(salary) as Highest_Salary

15 from datascience_analysis.datascience_table

16 group by experience_level order by Highest_Salary desc;

17

18

19 #Top-Paying Locations

SQL Ln 13, Col 1

Run again

Explain

Cancel

Clear

Create

Reuse query results up to 60 minutes ago

Query results

Query stats

Completed

Time in queue: 117 ms

Run time: 540 ms

Data scanned: 513.76 KB

Results (492)

Copy

Download results CSV

Search rows

< 1 >

#	experience_level	Highest_Salary
1	MI	3.04E7
2	SE	7000000.0
3	EX	6000000.0
4	EN	4450000.0

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Company Locations with the highest average salaries

17 #Top-Paying Locations

18 select company_location,

19 AVG(salary) as average_salary

20 from datascience_analysis.datascience_table

21 group by company_location

22 order by average_salary desc;

23

SQL Ln 18, Col 1

Run again

Explain

Cancel

Clear

Create

Reuse query results up to 60 minutes ago

Query results

Query stats

Completed

Time in queue: 61 ms

Run time: 469 ms

Data scanned: 36.54 KB

Results (50)

Copy

Download results CSV

Search rows

< 1 >

#	company_location	average_salary
1	CL	3.04E7
2	HU	1.1E7
3	JP	3408666.8
4	IN	2065208.2
5	AS	1335000.0
6	MX	279333.34

Salary Distribution by Company Size

25
26 #Salary Distribution by Company Size
27 select company_size,
28 AVG(salary) as average_salary
29 from datascience_analysis.datascience_table
30 group by company_size
31 order by average_salary desc;
32

SQL Ln 27, Col 1

Run again

Explain

Cancel

Clear

Create

Reuse query results
up to 60 minutes ago

Query results

Query stats

Completed

Time in queue: 100 ms Run time: 488 ms Data scanned: 36.54 KB

Results (3)

Copy

Download results CSV

Search rows

< 1 >

#	company_size	average_salary
1	L	593695.8
2	S	377710.0
3	M	146522.5

Top-Paying Job Titles in Each Location

30 #Top-Paying Job Titles in Each Location
31 select job_title,
32 company_location,
33 max(salary) as max_Salary
34 from datascience_analysis.datascience_table
35 group by
36 company_location, job_title
37 order by max_Salary desc;
38

SQL Ln 31, Col 1

Run again

Explain

Cancel

Clear

Create

Reuse query results
up to 60 minutes ago

Query results

Query stats

Completed

Time in queue: 93 ms Run time: 472 ms Data scanned: 36.54 KB

Results (179)

Copy

Download results CSV

Search rows

< 1 ... >

#	job_title	company_location	max_Salary
1	Data Scientist	CL	3.04E7
2	Data Scientist	HU	1.1E7
3	BI Data Analyst	US	1.1E7
4	ML Engineer	JP	8500000.0
5	Data Science Manager	IN	7000000.0
6	Head of Machine Learning	IN	6000000.0
7	Machine Learning Engineer	IN	4900000.0
8	Data Engineer	JP	4450000.0

Highest salary in each year

```
37
38 #Highest Salary in Each Year
39 select work_year,
40        max(salary) as max_Salary
41 from datascience_analysis.datascience_table
42 group by work_year
43 order by work_year;
44
```

SQL Ln 39, Col 1

[Run again](#) [Explain](#) [Cancel](#) [Clear](#) [Create](#)

☒ Reuse query results up to 60 minutes ago

[Query results](#) [Query stats](#)

Completed Time in queue: 110 ms Run time: 1.991 sec Data scanned: 513.76 KB

Results (4) [Copy](#) [Download results CSV](#)

Search rows

#	work_year	max_Salary
1	2020	1.1E7
2	2021	3.04E7
3	2022	6000000.0
4		

Insights and Results:

- **Head of Machine Learning** has the **highest average salary**.
- **Senior-level Software Engineer (SE)** earns the **highest salary by experience level**.
- **Chile (CL)** is the **top-paying location**.
- **Large companies (Company size L)** offer the **highest average salary**.
- **Data Scientist** is the **highest-ranking job title based on salary**.
- The **year 2020** recorded the **highest salary**.

The results are provided in the attached snapshots for each query.

Query Execution and Result Validation:

- Execution Method: Queries were executed directly in the Athena console.
- Result Location:

s3://data-science-job-salary-project/ds_analysis

Part 3:

Step-by-Step ETL Implementation using AWS Glue for Data Science Job Salary Analysis

Overview

This project demonstrates an ETL pipeline built using AWS Glue to process Data Science Job Salary data. The pipeline extracts data from an S3, transforms it using AWS Glue Jobs, and loads the transformed data into S3 for reporting and analytics.

Data Flow Pipeline

Source: S3 (Raw data: CSV) → AWS Glue Job (Transform: Renaming columns & Normalize) → S3 (Processed Data: CSV Format) → QuickSight (for Analysis & Visualization)

Data Extraction (Extract)

- Source: Amazon S3 (Raw Data) - The dataset includes salary details for Data Science job positions.

ETL Data Transformation (Transform) Steps in AWS Glue Studio

Steps:

1. Rename column names.

The screenshot shows the 'Change Schema (Apply mapping)' interface in AWS Glue Studio. At the top, it says 'Choose which nodes will provide inputs for this one.' and 'Choose one or more parent node'. Below this, a box indicates 'Amazon S3' and 'S3 - DataSource'. The main part of the interface is a table with four columns: 'Source key', 'Target key', 'Data type', and 'Drop'. The table lists 12 source keys (col0 to col11) and maps them to target keys (id, work_year, experience, employme, job_title, salary, salary_curr, salary_in_1, employee_, remote_rai, company_l, company_s) with their respective data types (int, string, float) and a 'Drop' checkbox.

Source key	Target key	Data type	Drop
col0	id	int	<input type="checkbox"/>
col1	work_year	int	<input type="checkbox"/>
col2	experience	string	<input type="checkbox"/>
col3	employme	string	<input type="checkbox"/>
col4	job_title	string	<input type="checkbox"/>
col5	salary	float	<input type="checkbox"/>
col6	salary_curr	string	<input type="checkbox"/>
col7	salary_in_1	float	<input type="checkbox"/>
col8	employee_	string	<input type="checkbox"/>
col9	remote_rai	int	<input type="checkbox"/>
col10	company_l	string	<input type="checkbox"/>
col11	company_s	string	<input type="checkbox"/>

2. Standardizing Column Values

As part of the transformation process, the **employment_type**, **company_size**, and **experience_level** columns are standardized to improve data consistency. This includes:

- Converting **employment_type** (e.g., 'PT' → 'Part-time').
- Transforming **company_size** (e.g., 'S' → 'Small').
- Modifying **experience_level** (e.g., 'EN' → 'Entry-level').

The following SQL query is applied during the transformation to ensure uniformity:

The screenshot shows the 'ds-glue-transformation' interface. At the top right, it says 'Last modified on 4/3/2025, 7:38:01 PM' and has buttons for 'Actions', 'Save', and 'Run'. Below this, there are two sections: 'Input sources' with a 'Change Schema' button, and 'SQL aliases' with a text box containing 'myDataSource'. The main section is 'SQL query', which includes the instruction 'Enter a SQL statement to add to your job.' and a large text area containing the following SQL query:

```
1 SELECT *,
2 CASE
3   WHEN employment_type = 'PT' THEN 'Part-time'
4   WHEN employment_type = 'FT' THEN 'Full-time'
5   WHEN employment_type = 'CT' THEN 'Contract'
6   WHEN employment_type = 'FL' THEN 'Freelance'
7 END AS employment_type,
8 CASE
9   WHEN company_size = 'S' THEN 'small'
10  WHEN company_size = 'M' THEN 'medium'
11  WHEN company_size = 'L' THEN 'large'
12 END AS company_size,
13 CASE
14   WHEN experience_level = 'EN' THEN 'Entry-level'
15   WHEN experience_level = 'MI' THEN 'Junior mid level'
16   WHEN experience_level = 'SE' THEN 'Intermediate Senior-level'
17   WHEN experience_level = 'EX' THEN 'Expert Executive-level/ Director'
18 END AS experience_level
19 FROM myDataSource
20
```

ds-glue-transformation

Last modified on 4/3/2025, 7:38:01 PM

Actions

Save

Run

Visual

Script

Job details

Runs

Data quality

Schedules

Version Control

+

Data source - S3 bucket

Amazon S3

Transform - Change Schema

Change Schema

Transform - SQL Query

SQL Query

Data target - S3 bucket

Target Amazon S3

Data preview

Output schema

Data source properties - S3

Name

Amazon S3

S3 source type

Info

S3 location

Choose a file or folder in an S3 bucket.

Data Catalog table

S3 URL

s3://data-science-job-salary-projec

View

Browse S3

Recursive

Read files in all subdirectories.

Data format

CSV

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

Data Loaded (Load) to S3

- Processed Data Format: CSV
- Storage: **s3://data-science-job-salary-project/ds_glue_tartget/**

ds_glue_tartget/

Copy S3 URI

Objects

Properties

Objects (1)

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

	Name	Type	Last modified	Size	Storage class
	run-1743728966949-part-r-00000	-	April 3, 2025, 18:09:33 (UTC-07:00)	43.8 KB	Standard

Part 4:

Visualization using QUICKSIGHT

Overview

In this section, we will analyze the **Data Science Job Salaries** dataset to reveal key insights, including salary trends, job role distributions, and other significant patterns. Leveraging **Amazon QuickSight**, to build a visually engaging dashboard that delivers a clear and comprehensive overview of the data.

Configuring Permissions for QuickSight Access

To allow Amazon QuickSight to access the data stored in Amazon S3, we configured the following two settings:

1. IAM Policy Configuration

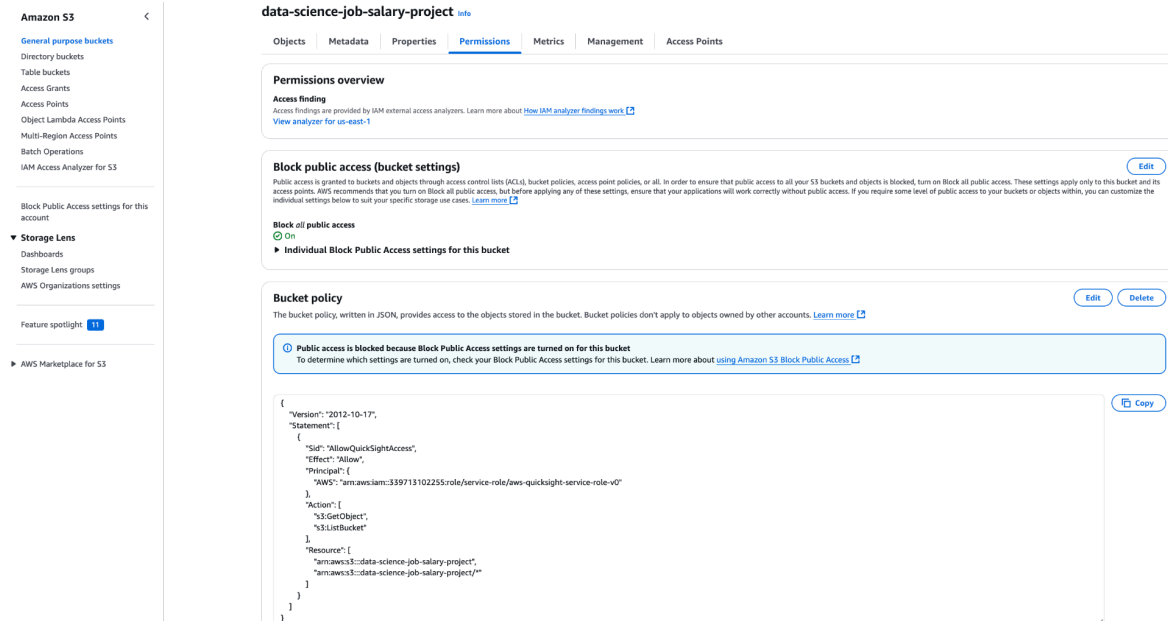
Create and attach an IAM policy that grants the necessary permissions—specifically `s3:GetObject` and `s3:ListBucket`—to the IAM role (`aws-quicksight-service-role-v0`) or the user associated with QuickSight.

The screenshot displays the AWS IAM console interface for the `aws-quicksight-service-role-v0` role. The left sidebar shows the navigation menu with 'Identity and Access Management (IAM)' selected. The main content area shows the role's details, including its ARN and creation date. The 'Permissions' tab is active, showing a list of three attached policies: `AWSQuickSightIAMPolicy`, `AWSQuickSightS3Policy`, and `QuickSightS3AccessPolicy`. The `QuickSightS3AccessPolicy` is expanded, revealing its JSON content, which grants `s3:GetObject` and `s3:ListBucket` permissions on the `arn:aws:s3::data-science-job-salary-project` resource. The JSON content is as follows:

```
1- {
2-   "Version": "2012-10-17",
3-   "Statement": [
4-     {
5-       "Effect": "Allow",
6-       "Action": [
7-         "s3:GetObject",
8-         "s3:ListBucket"
9-       ],
10-      "Resource": [
11-        "arn:aws:s3::data-science-job-salary-project",
12-        "arn:aws:s3::data-science-job-salary-project/*"
13-      ]
14-    }
15-  ]
16- }
```

2. S3 Bucket Policy Update

Modify the S3 bucket policy to explicitly grant access to QuickSight. This ensures that QuickSight can properly retrieve data from the specified S3 bucket.



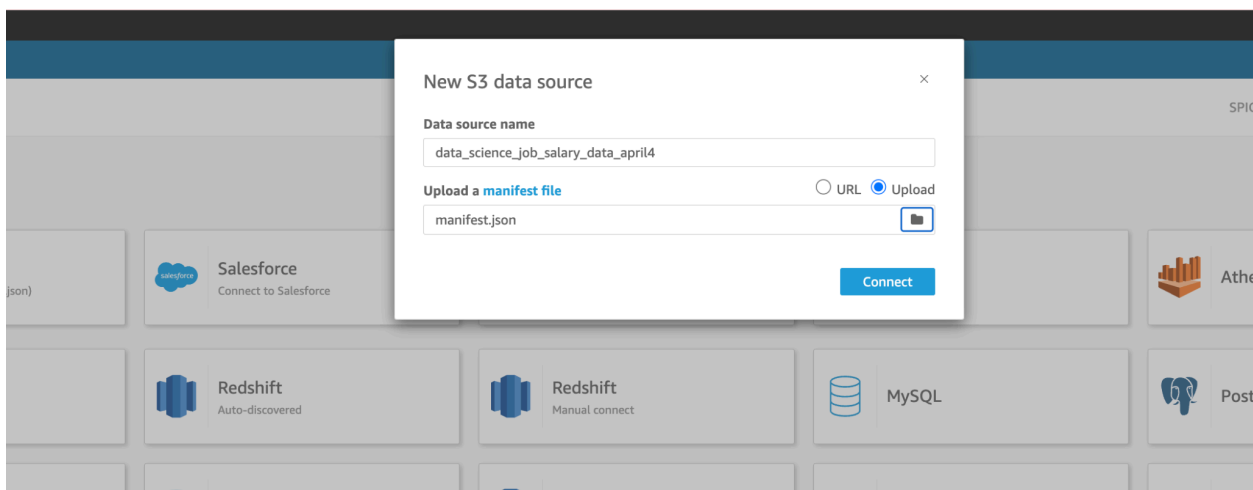
Data Source:

- Transformed data from the AWS Glue ETL job, stored in an S3 bucket.
- Data: S3 Bucket:

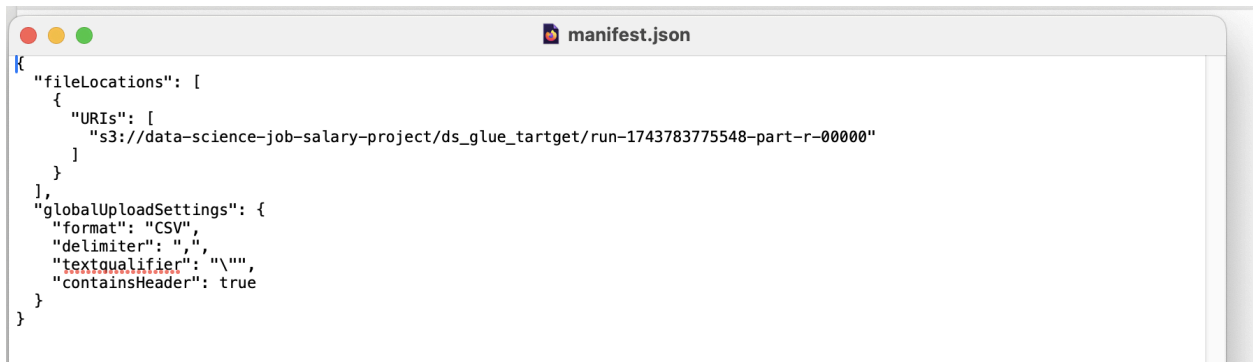
`s3://data-science-job-salary-project/ds_glue_target/run-1743783775548-part-r-00000`

Connecting QuickSight to the Dataset S3:

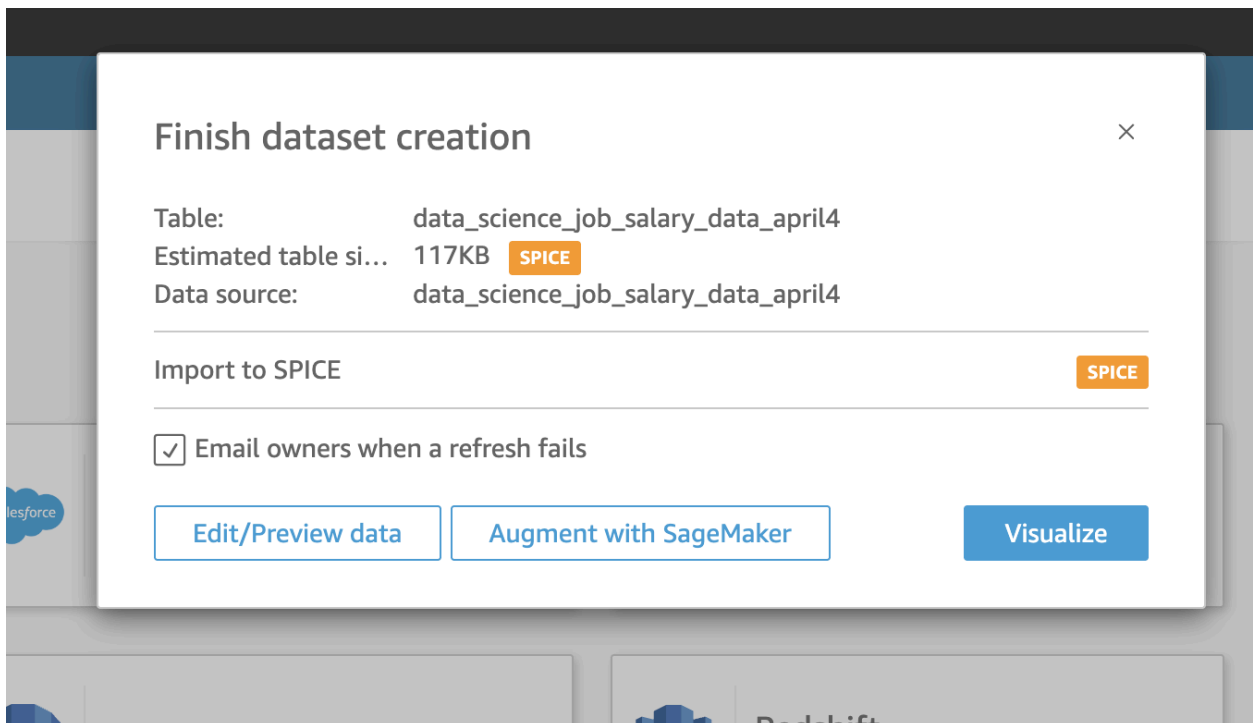
A new dataset named `data_science_job_salary_data_april4` was created in QuickSight by connecting to the S3 bucket (`s3://data-science-job-salary-project/ds_glue_target/run-1743783775548-part-r-00000`), where the transformed data is stored in CSV format.



Manifest.json file: a manifest file is used to connect to data stored in Amazon S3. It helps QuickSight understand how to locate and interpret your files in S3.



```
manifest.json
{
  "fileLocations": [
    {
      "URIs": [
        "s3://data-science-job-salary-project/ds_glue_tartget/run-1743783775548-part-r-00000"
      ]
    }
  ],
  "globalUploadSettings": {
    "format": "CSV",
    "delimiter": ",",
    "textQualifier": "\"",
    "containsHeader": true
  }
}
```



Finish dataset creation [Close]

Table: data_science_job_salary_data_april4

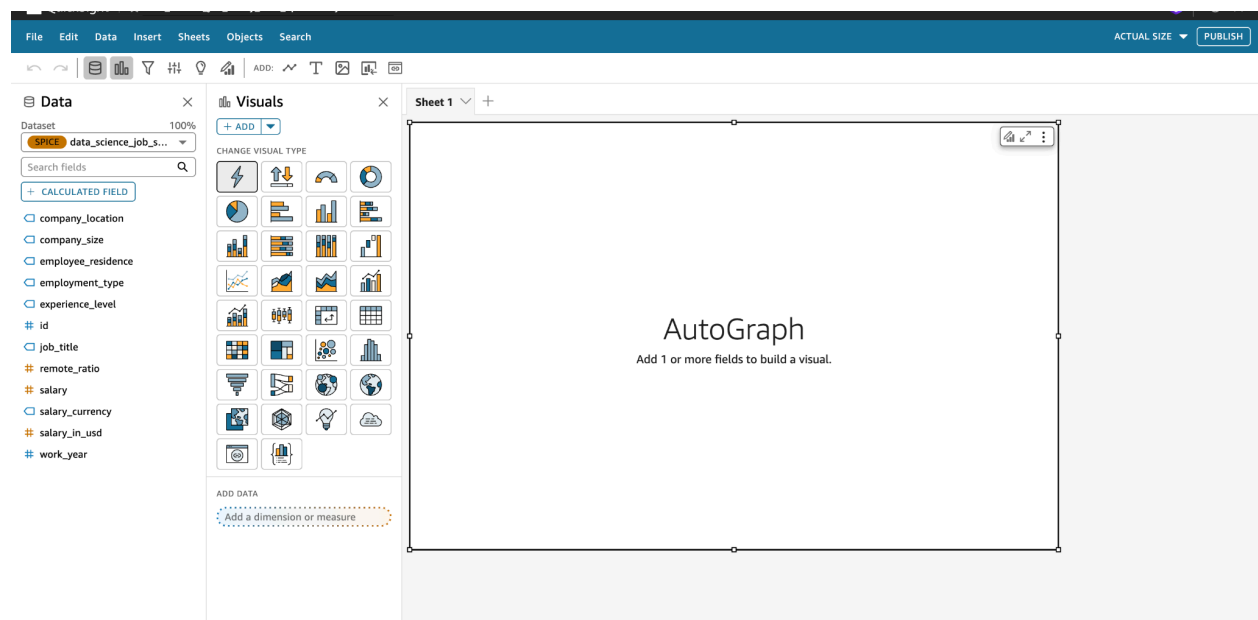
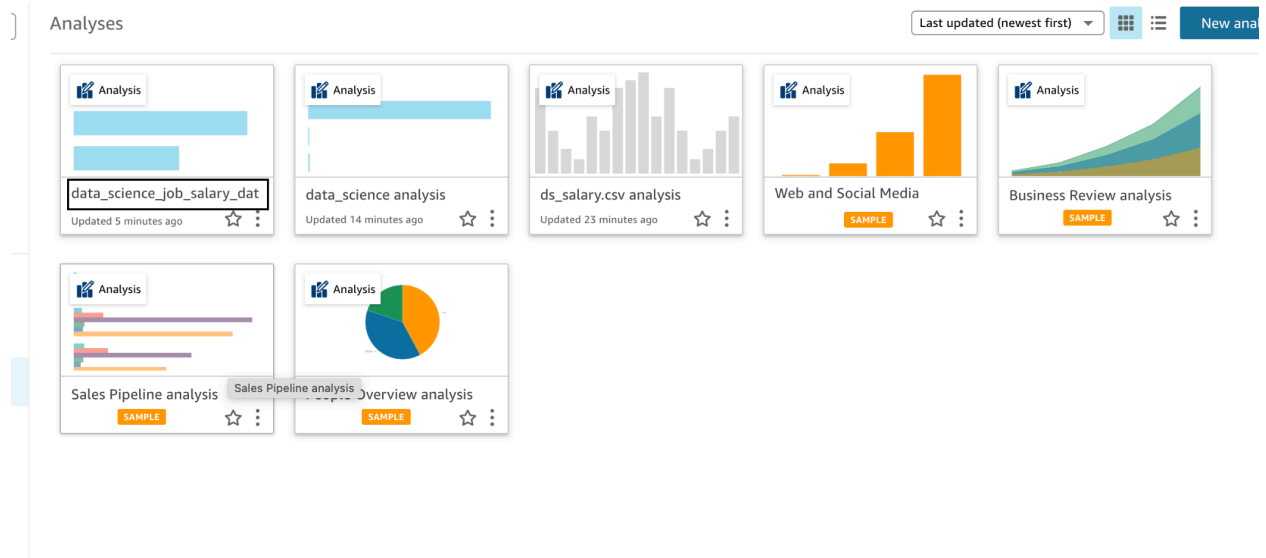
Estimated table size: 117KB [SPICE]

Data source: data_science_job_salary_data_april4

Import to SPICE [SPICE]

☒ Email owners when a refresh fails

[Edit/Preview data] [Augment with SageMaker] [Visualize]



Creating Visualizations in QuickSight:

The following visualizations were developed using Amazon QuickSight:

- Created below visualizations:
- Average Salary by Job Title
- Average Salary by Employee Country
- Average Salary by Experience Level
- Average Salary by Company Size
- Average Salary by Remote Ratio
- Average Salary Trend Over Years
- Distribution of Employees by Experience Level

- Job Title Distribution
- Funnel View of Job Titles
- Employee Distribution by Job Title and Company Size
- Experience Level Distribution by Employment Type
- Funnel View of Employees by Residence

Sharing the Dashboard:

- **Embedding:** The Amazon QuickSight dashboard embedded into a web page. Access to the embedded dashboard requires an AWS account.

https://us-east-1.quicksight.aws.amazon.com/sn/accounts/339713102255/dashboards/f3b37240-37ea-4dc9-bf06-bd86298a22b1?directory_alias=seema

- **Exporting:** The dashboard exported as a PDF.

[Click here to VIEW PDF](#)

Dashboard Insights:

Key Findings from the Data Science Job Salary Analysis Dashboard

- The majority of employees fall under the **Intermediate/Senior-level** experience category.
- **Data Scientist** is the most common job title among employees.
- **Head of Machine Learning** roles command the highest average salary among all job titles, while the **Junior/Mid-level** experience group earns the highest average salary compared to other experience levels.
- **Large companies** offer the highest average salaries compared to medium and small companies.
- Employees with a **fully remote** work setup (remote_ratio = 100%) earn the highest average salary.
- The year **2021** recorded the highest average salary across all years.
- **Full-time** employment is the most common, with the **Intermediate/Senior-level** making up the largest portion of this group.

- Among all job titles, **Data Scientist** roles are most prevalent, with the majority working in **medium-sized companies**.
- Employees residing in **CL** (Chile) have the highest average salary by country of residence.
- The **United States (US)** has the highest number of employees represented in the dataset.