

Step 1: Creating bucket on aws

The screenshot shows the AWS S3 console interface. The left sidebar contains the 'Amazon S3' menu with options like 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'IAM Access Analyzer for S3', 'Storage Lens', 'Dashboards', and 'AWS Organizations settings'. The main content area is titled 'Buckets (22)' and includes a search bar 'Find buckets by name' and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. A table lists the buckets:

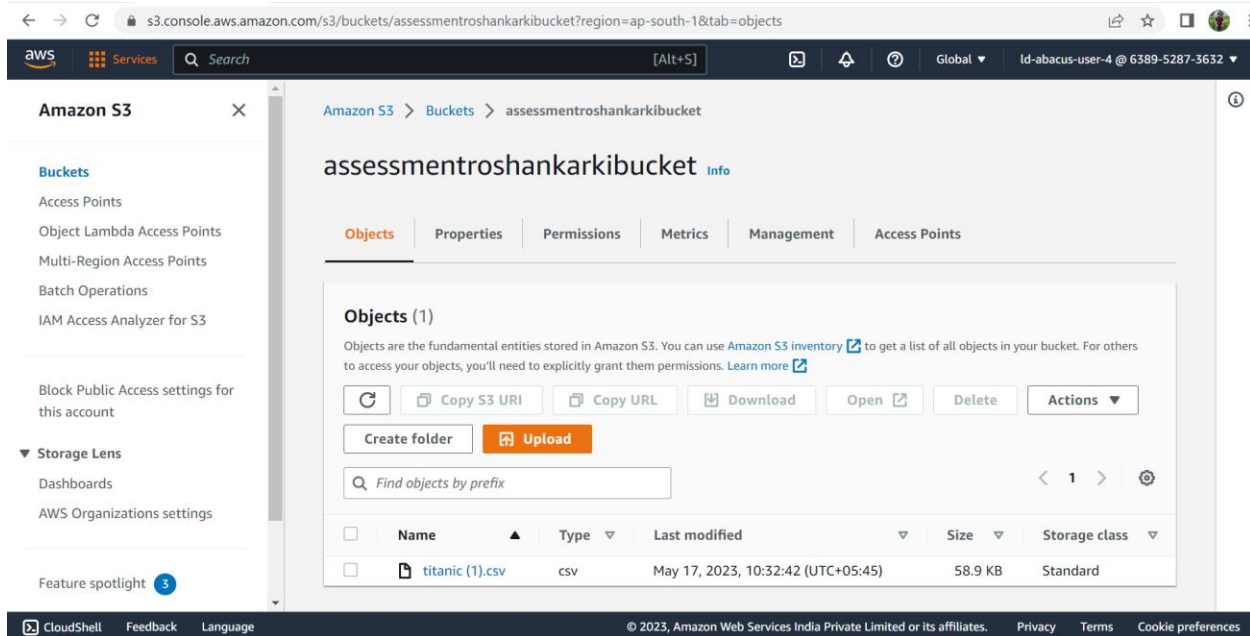
Name	AWS Region	Access	Creation date
assessmenttroshankarkibucket	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	May 17, 2023, 10:31:14 (UTC)
bipinbucket123	US East (N. Virginia) us-east-1	Bucket and objects not public	May 10, 2023, 10:50:50 (UTC)
bipinkhanalbucket1234	US East (N. Virginia) us-east-1	Bucket and objects not public	May 10, 2023, 10:56:40 (UTC)

The screenshot shows the AWS S3 console interface for the 'Upload' page of the 'assessmenttroshankarkibucket'. A green banner at the top indicates 'Upload succeeded'. Below the banner, a table shows the upload details:

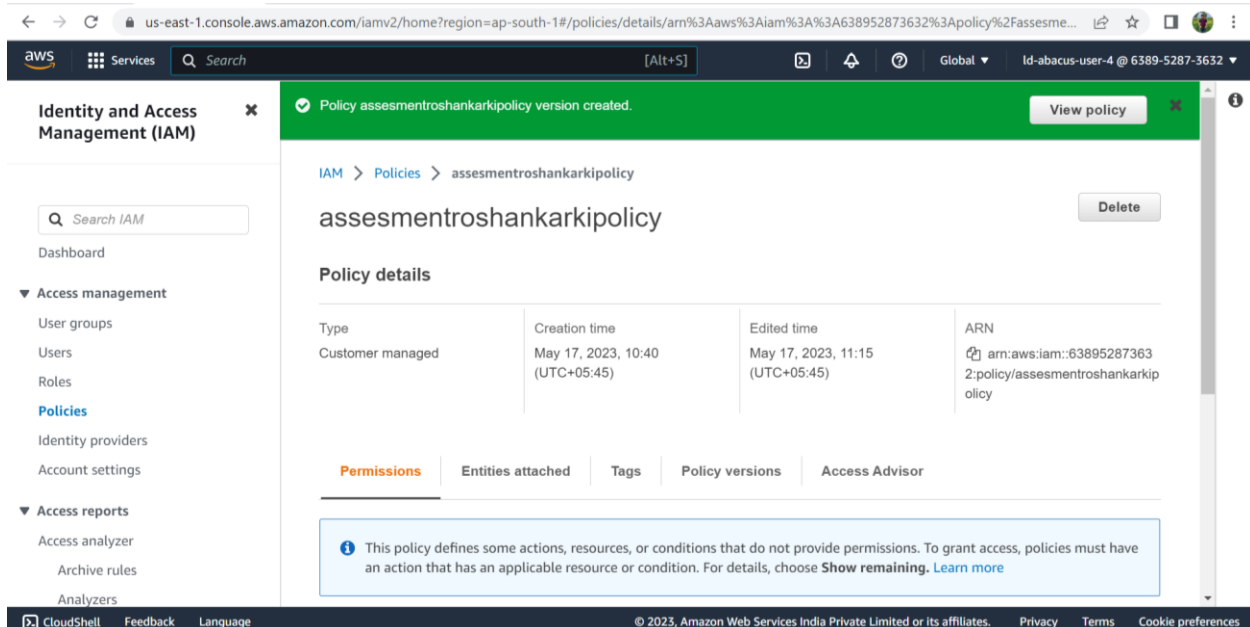
Destination	Succeeded	Failed
s3://assessmenttroshankarkibucket	1 file, 58.9 KB (100.00%)	0 files, 0 B (0%)

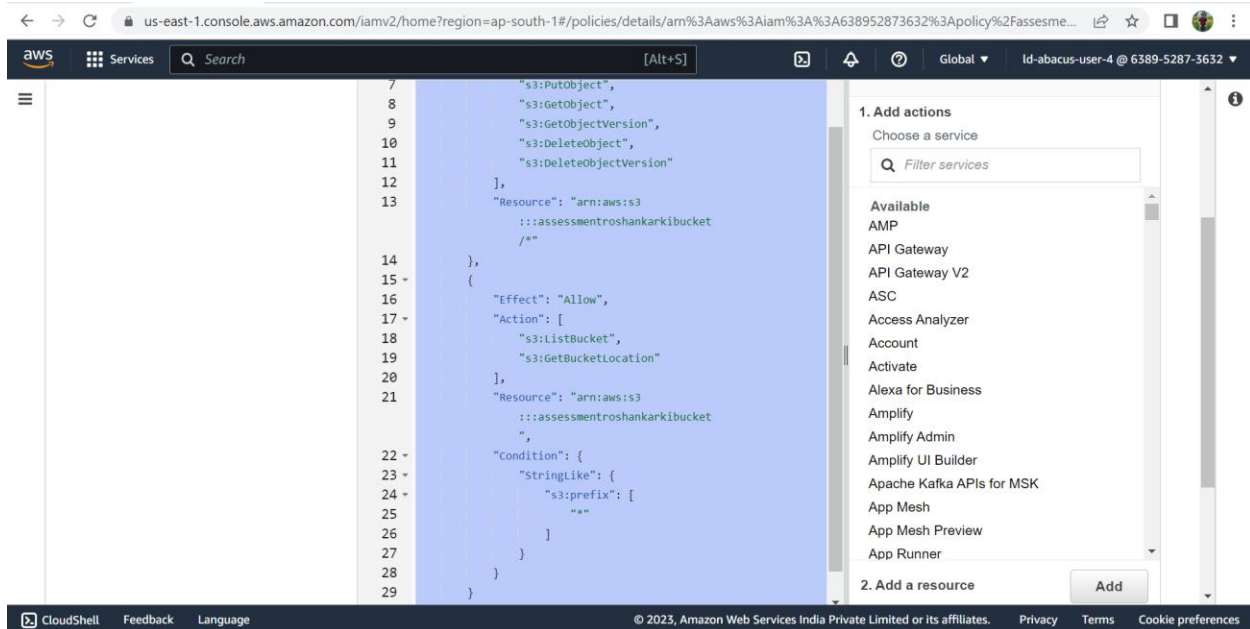
Below the table, the 'Files and folders' section shows a list of files and folders. The table has columns: Name, Folder, Type, Size, Status, and Error.

Name	Folder	Type	Size	Status	Error
titanic (1).csv	-	text/csv	58.9 KB	Succeeded	-



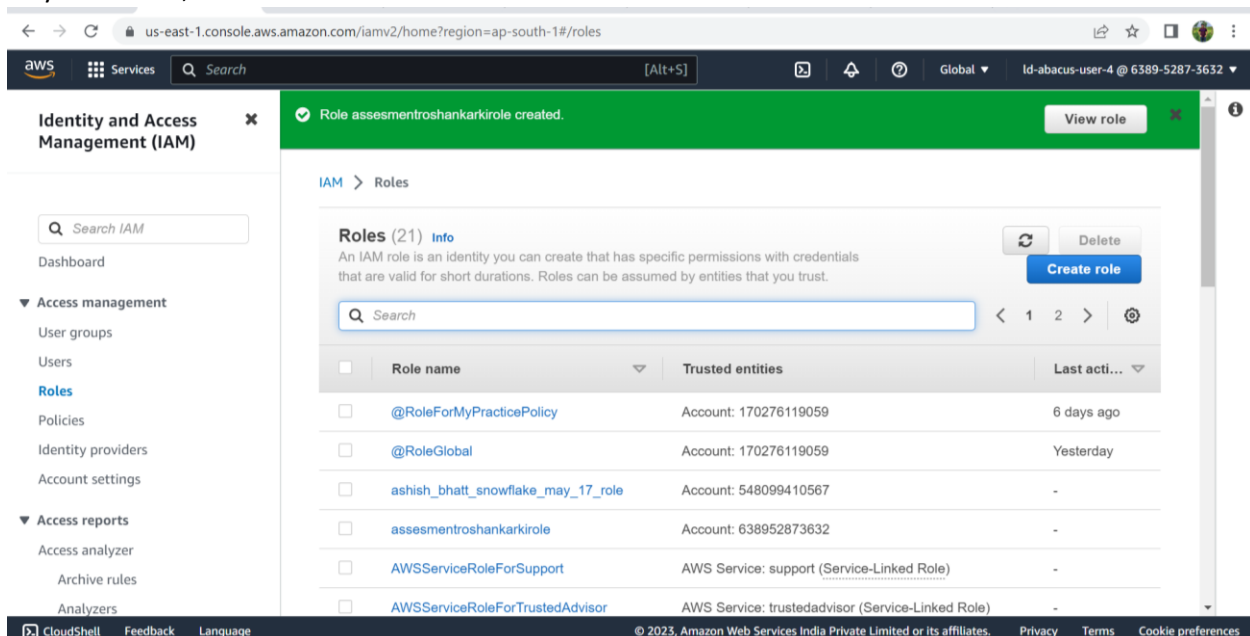
Step 2: Creating policy





Step 3: Creating role

As you can see, role as 'assessmenttroshankarkirole' has been created.



Creating storage integration:

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am + PREVIEW

Databases Worksheets

Search + ...

- 2023-05-16 10:45am
- 2023-05-15 4:44pm
- 2023-05-15 10:42am
- 2023-05-11 2:46pm
- snowpipe
- 2023-05-09 2:45pm
- BucketAndReadingLoading
- 2023-05-08 3:06pm
- 2023-05-08 1:03pm
- 2023-05-08 1:03pm
- 2023-05-08 12:46pm
- 2023-05-08 10:42am
- 2023-05-05 12:26pm
- store_procedure_day5
- 2023-05-03 3:20pm
- 2023-05-03 3:00pm
- creatingFormatAndReading

FIRSTDATABASE.PERSON Settings Latest Version

```

1 create or replace storage integration awsStorageIntegration100
2   type = external_stage
3   storage_provider = s3
4   storage_aws_role_arn = 'arn:aws:iam::638952873632:role/assessmenttroshankarkirole'
5   enabled = true
6   storage_allowed_locations = ( 's3://assessmenttroshankarkibucket' )
7   -- storage_blocked_locations = ( 's3://<location1>', 's3://<location2>' )
8   -- comment = '<comment>';
9

```

Results Chart

status
1 Integration AWSSTORAGEINTEGRATION100 successfully created.

Query Details

- Query duration 87ms
- Rows 1
- Query ID 01ac572b-0000-5a4a-...
- status 100% filled

Looking for external id and user arn

← → ↺ app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am + PREVIEW

Databases Worksheets

Search + ...

- 2023-05-16 10:45am
- 2023-05-15 4:44pm
- 2023-05-15 10:42am
- 2023-05-11 2:46pm
- snowpipe
- 2023-05-09 2:45pm
- BucketAndReadingLoading
- 2023-05-08 3:06pm
- 2023-05-08 1:03pm
- 2023-05-08 1:03pm
- 2023-05-08 12:46pm
- 2023-05-08 10:42am
- 2023-05-05 12:26pm
- store_procedure_day5
- 2023-05-03 3:20pm
- 2023-05-03 3:00pm
- creatingFormatAndReading

FIRSTDATABASE.PERSON Settings Latest Version

```

7   -- storage_blocked_locations = ( 's3://<location1>', 's3://<location2>' )
8   -- comment = '<comment>';
9
10  //( it will give External id and User ARN)
11  desc integration awsStorageIntegration100
12
13  //creating table
14  CREATE or REPLACE TABLE titanic (
15    PassengerId INT NOT NULL,

```

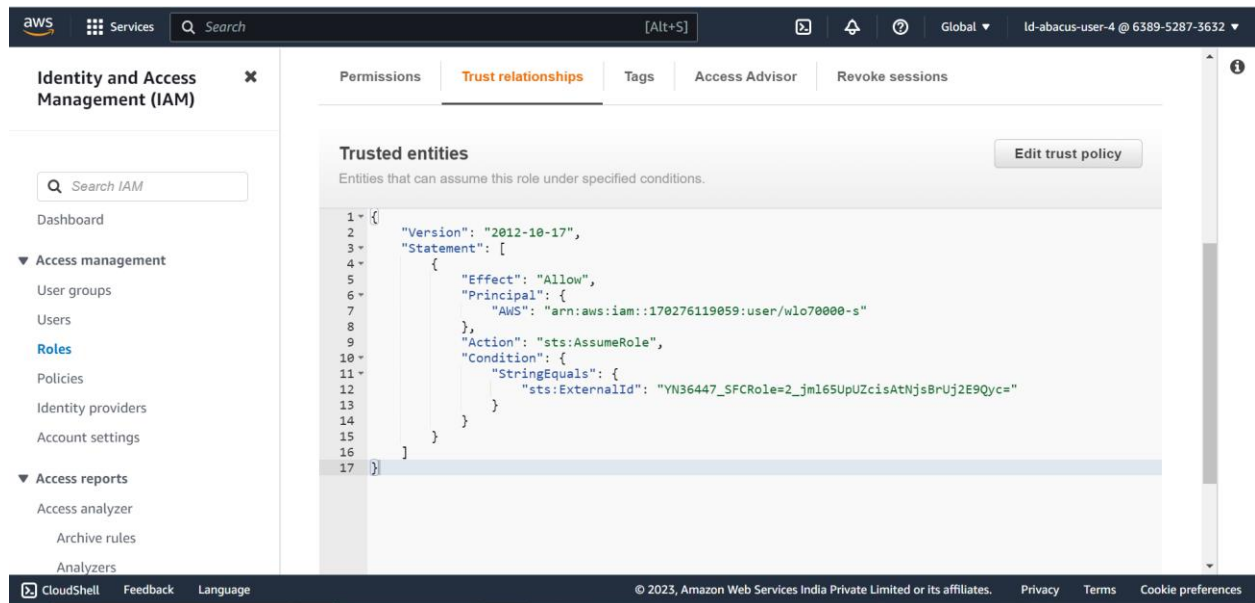
Results Chart

	property	property_type	property_value
1	ENABLED	Boolean	true
2	STORAGE_PROVIDER	String	S3
3	STORAGE_ALLOWED_LOCATIONS	List	s3://assessmenttroshankarkibucket
4	STORAGE_BLOCKED_LOCATIONS	List	
5	STORAGE_AWS_IAM_USER_ARN	String	arn:aws:iam::170276119059:user/w1070000-s
6	STORAGE_AWS_ROLE_ARN	String	arn:aws:iam::638952873632:role/assessmenttroshankarkirole
7	STORAGE_AWS_EXTERNAL_ID	String	YN36447_SFCRole=2_S4GNo78m1kpKfkaFDNLiz8tfzSo=
8	COMMENT	String	

Query Details

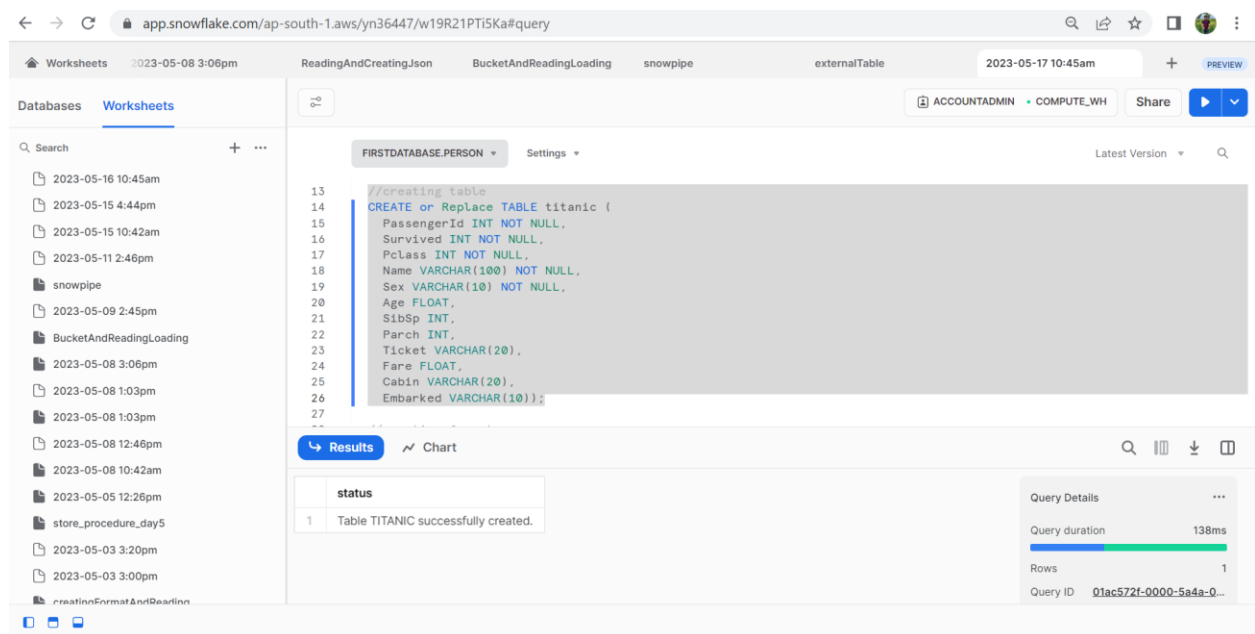
- Query duration 821ms
- Rows 8
- Query ID 01ac572d-0000-5a34-...
- property 100% filled

Now, adding the user-arn and external_id into role.



Step 4: doing it in SQL sheet

Creating table



Creating file format:

```
27
28 //creating format
29 create or replace FILE FORMAT FILE_FORMAT_Roshan
30 TYPE = 'CSV'
31 SKIP_HEADER = 1
32 FIELD_DELIMITER = ','
33 FIELD_OPTIONALLY_ENCLOSED_BY = ''
34
```

	status
1	File format FILE_FORMAT_ROSHAN successfully created.

Query Details ...
Query duration 62ms
Rows 1
Query ID 01ac5730-0000-5a4b-...
status A
100% filled

Creating external stage

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

ACCOUNTADMIN COMPUTE_WH Share

Search 2023-05-16 10:45am 2023-05-15 4:44pm 2023-05-15 10:42am 2023-05-11 2:46pm snowpipe 2023-05-09 2:45pm BucketAndReadingLoading 2023-05-08 3:06pm 2023-05-08 1:03pm 2023-05-08 12:46pm 2023-05-08 10:42am 2023-05-05 12:26pm store_procedure_day5 2023-05-03 3:20pm 2023-05-03 3:00pm creationFormatAndReading

```
31 SKIP_HEADER = 1
32 FIELD_DELIMITER = ','
33 FIELD_OPTIONALLY_ENCLOSED_BY = ''
34
35 CREATE or Replace STAGE my_external_stage
36 STORAGE_INTEGRATION = awsStorageIntegration100
37 URL = 's3://assessmentroshankarkibucket'
38 FILE FORMAT = FILE FORMAT
```

	status
1	Stage area MY_EXTERNAL_STAGE successfully created.

Query Details ...
Query duration 1.9s
Rows 1
Query ID 01ac5731-0000-5a0d-...
status A
100% filled

```

39
40 COPY INTO titanic
41 FROM @my_external_stage
42 FILE_FORMAT = (FORMAT_NAME='FILE_FORMAT_Roshan');
43
44 SELECT * FROM titanic;

```

		status	rows_parsed	rows_loaded	error_limit
1	://assessmentroshankarkibucket/titanic (1).csv	LOADED	891	891	1

Selecting queries:

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreatingJson BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

2023-05-16 10:45am
2023-05-15 4:44pm
2023-05-15 10:42am
2023-05-11 2:46pm
snowpipe
2023-05-09 2:45pm
BucketAndReadingLoading
2023-05-08 3:06pm
2023-05-08 1:03pm
2023-05-08 12:46pm
2023-05-08 10:42am
2023-05-05 12:26pm
store_procedure_day5
2023-05-03 3:20pm
2023-05-03 3:00pm
creationFormatAndReading

```

44 SELECT * FROM titanic;
45
46 //Those people, who are survived
47 select count(*) as Survived from titanic where SURVIVED = 1;
48
49 //Those people, who died
50 select count(*) as Survived from titanic where SURVIVED = 0;

```

	PASSENGERID	SURVIVED	PCLASS	NAME	SE:
1	1	0	3	Braund, Mr. Owen Harris	ma
2	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Thayer)	fen
3	3	1	3	Heikkinen, Miss. Laina	fen
4	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	fen
5	5	0	3	Allen, Mr. William Henry	ma
6	6	0	3	Moran, Mr. James	ma
7	7	0	1	McCarthy, Mr. Timothy J	ma
8	8	0	3	Palsson, Master. Gosta Leonard	ma
9	9	1	2	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Barn)	fen

Query Details

PASSENGERID #

1 891

SURVIVED #

A. How many people survive and how many people dead

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

Search 2023-05-16 10:45am 2023-05-15 4:44pm 2023-05-15 10:42am 2023-05-11 2:46pm snowpipe 2023-05-09 2:45pm BucketAndReadingLoading 2023-05-08 3:06pm 2023-05-08 1:03pm 2023-05-08 1:03pm 2023-05-08 12:46pm 2023-05-08 10:42am 2023-05-05 12:26pm store_procedure_day5 2023-05-03 3:20pm 2023-05-03 3:00pm creatingFormatAndReading

FIRSTDATABASE.PERSON Settings Latest Version

```
46 //Those people, who are survived
47 select count(*) as Survived from titanic where SURVIVED = 1;
48
49 //Those people, who died
50 select count(*) as Survived from titanic where SURVIVED = 0;
51
52 //those people, who survive and does not have sibling
```

Results Chart

SURVIVED	
1	342

Query Details

Query duration 149ms

Rows 1

Query ID 01ac5733-0000-5a0d...

SURVIVED #

100% filled

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

Search 2023-05-16 10:45am 2023-05-15 4:44pm 2023-05-15 10:42am 2023-05-11 2:46pm snowpipe 2023-05-09 2:45pm BucketAndReadingLoading 2023-05-08 3:06pm 2023-05-08 1:03pm 2023-05-08 1:03pm 2023-05-08 12:46pm 2023-05-08 10:42am 2023-05-05 12:26pm store_procedure_day5 2023-05-03 3:20pm 2023-05-03 3:00pm creatingFormatAndReading

FIRSTDATABASE.PERSON Settings Latest Version

```
46 //Those people, who are survived
47 select count(*) as Survived from titanic where SURVIVED = 1;
48
49 //Those people, who died
50 select count(*) as Survived from titanic where SURVIVED = 0;
51
52 //those people, who survive and does not have sibling
```

Results Chart

SURVIVED	
1	549

Query Details

Query duration 151ms

Rows 1

Query ID 01ac5734-0000-5a4a...

SURVIVED #

100% filled

B. How many people survive how does not have sibling

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreatingJson BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

Search

2023-05-16 10:45am
2023-05-15 4:44pm
2023-05-15 10:42am
2023-05-11 2:46pm
snowpipe
2023-05-09 2:45pm
BucketAndReadingLoading
2023-05-08 3:06pm
2023-05-08 1:03pm
2023-05-08 1:03pm
2023-05-08 12:46pm
2023-05-08 10:42am
2023-05-05 12:26pm
store_procedure_day5
2023-05-03 3:20pm
2023-05-03 3:00pm
creationFormatAndReading

FIRSTDATABASE.PERSON Settings

```
52 //those people, who survive and does not have sibling
53 SELECT COUNT (*) AS SURVIVED_NO_SIBLING from titanic where survived=1 AND sibsp=0;
54
55 //those people, who died and does not have sibling
56 SELECT COUNT (*) AS Died_NO_SIBLING from titanic where survived=0 AND sibsp=0;
57
58 //removing the nan data from cabin and displaying updated list
```

Results Chart

	SURVIVED_NO_SIBLING
1	210

Query Details

Query duration 246ms

Rows 1

Query ID 01ac5734-0000-5a4b-...

SURVIVED_NO_SIBLING #

100% filled

https://app.snowflake.com/ap-south-1.aws/yn36447/worksheets

C. How many people died how does not have sibling

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreatingJson BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

Search

2023-05-16 10:45am
2023-05-15 4:44pm
2023-05-15 10:42am
2023-05-11 2:46pm
snowpipe
2023-05-09 2:45pm
BucketAndReadingLoading
2023-05-08 3:06pm
2023-05-08 1:03pm
2023-05-08 1:03pm
2023-05-08 12:46pm
2023-05-08 10:42am
2023-05-05 12:26pm
store_procedure_day5
2023-05-03 3:20pm
2023-05-03 3:00pm
creationFormatAndReading

FIRSTDATABASE.PERSON Settings

```
52 //those people, who survive and does not have sibling
53 SELECT COUNT (*) AS SURVIVED_NO_SIBLING from titanic where survived=1 AND sibsp=0;
54
55 //those people, who died and does not have sibling
56 SELECT COUNT (*) AS Died_NO_SIBLING from titanic where survived=0 AND sibsp=0;
57
58 //removing the nan data from cabin and displaying updated list
```

Results Chart

	DIED_NO_SIBLING
1	398

Query Details

Query duration 70ms

Rows 1

Query ID 01ac5734-0000-5a0d-...

DIED_NO_SIBLING #

100% filled

D. Remove the NAN data from cabin and display updated list

After doing everything, it did it at last to remove nan.

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

Search

2023-05-16 10:45am
2023-05-15 4:44pm
2023-05-15 10:42am
2023-05-11 2:46pm
snowpipe
2023-05-09 2:45pm
BucketAndReadingLoading
2023-05-08 3:06pm
2023-05-08 1:03pm
2023-05-08 1:03pm
2023-05-08 12:46pm
2023-05-08 10:42am
2023-05-05 12:26pm
store_procedure_day5
2023-05-03 3:20pm
2023-05-03 3:00pm
creationFormatAndReading

FIRSTDATABASE.PERSON Settings

```
55 //those people, who died and does not have sibling
56 SELECT COUNT (*) AS Died_NO_SIBLING from titanic where survived=0 AND sibsp=0;
57
58 //removing the nan data from cabin and displaying updated list
59 SELECT * FROM titanic WHERE CABIN IS NOT NULL;
60 Delete from titanic WHERE CABIN IS NULL;
61
```

Results Chart

	number of rows deleted
1	687

Query Details

Query duration 278ms

Rows 1

Query ID 01ac5738-0000-5a48...

number of rows deleted #

100% filled

E. How many male who died and how many female died?

app.snowflake.com/ap-south-1.aws/yn36447/w19R21PTi5Ka#query

Worksheets 2023-05-08 3:06pm ReadingAndCreating.Json BucketAndReadingLoading snowpipe externalTable 2023-05-17 10:45am

Databases Worksheets

Search

2023-05-16 10:45am
2023-05-15 4:44pm
2023-05-15 10:42am
2023-05-11 2:46pm
snowpipe
2023-05-09 2:45pm
BucketAndReadingLoading
2023-05-08 3:06pm
2023-05-08 1:03pm
2023-05-08 1:03pm
2023-05-08 12:46pm
2023-05-08 10:42am
2023-05-05 12:26pm
store_procedure_day5
2023-05-03 3:20pm
2023-05-03 3:00pm
creationFormatAndReading

FIRSTDATABASE.PERSON Settings

```
61
62 //male who died
63 SELECT COUNT (*) AS male_died FROM titanic WHERE sex='male' AND survived=0;
64
65 //female who died
66 SELECT COUNT (*) AS female_died FROM titanic WHERE sex='female' AND survived=0;
67
```

Results Chart

	MALE_DIED
1	468

Query Details

Query duration 148ms

Rows 1

Query ID 01ac5735-0000-5a4b...

MALE_DIED #

100% filled

The screenshot shows the Snowflake web interface. On the left, there's a sidebar with 'Databases' and 'Worksheets'. The 'Worksheets' tab is active, showing a list of worksheets. The main area displays a SQL query in a text editor. The query is as follows:

```

61
62 //male who died
63 SELECT COUNT (*) AS male_died FROM titanic WHERE sex='male' AND survived=0;
64
65 //female who died
66 SELECT COUNT (*) AS female_died FROM titanic WHERE sex='female' AND survived=0;
67

```

Below the query editor, there's a 'Results' tab. It shows a table with one row and one column:

FEMALE_DIED
81

On the right side of the 'Results' tab, there's a 'Query Details' panel. It shows the following information:

- Query duration: 134ms
- Rows: 1
- Query ID: 01ac5735-0000-5a4b...
- FEMALE_DIED: 100% filled

SQL Query:

//creating storage integration

create or replace storage integration awsStorageIntegration100

type = external_stage

storage_provider = s3

storage_aws_role_arn = 'arn:aws:iam::638952873632:role/assessmenttroshankarkirole'

enabled = true

storage_allowed_locations = ('s3://assessmenttroshankarkibucket')

-- storage_blocked_locations = ('s3://<location1>', 's3://<location2>')

-- comment = '<comment>';

//(it will give External id and User ARN)

desc integration awsStorageIntegration100

//creating table

CREATE or Replace TABLE titanic (

PassengerId INT NOT NULL,

```
Survived INT NOT NULL,  
Pclass INT NOT NULL,  
Name VARCHAR(100) NOT NULL,  
Sex VARCHAR(10) NOT NULL,  
Age FLOAT,  
SibSp INT,  
Parch INT,  
Ticket VARCHAR(20),  
Fare FLOAT,  
Cabin VARCHAR(20),  
Embarked VARCHAR(10));
```

```
//creating format
```

```
create or replace FILE FORMAT FILE_FORMAT_Roshan  
TYPE = 'CSV'  
SKIP_HEADER = 1  
FIELD_DELIMITER = ','  
FIELD_OPTIONALLY_ENCLOSED_BY = ''
```

```
//creating external stage
```

```
CREATE or Replace STAGE my_external_stage  
STORAGE_INTEGRATION = awsStorageIntegration100  
URL = 's3://assessmenttroshankarkibucket'  
FILE_FORMAT = FILE_FORMAT;
```

```
//copying from stage to table.
```

```
COPY INTO titanic  
FROM @my_external_stage  
FILE_FORMAT = (FORMAT_NAME='FILE_FORMAT_Roshan');
```

```
SELECT * FROM titanic;
```

```
//Those people, who are survived
```

```
select count(*) as Survived from titanic where SURVIVED = 1;
```

```
//Those people, who died
```

```
select count(*) as Survived from titanic where SURVIVED = 0;
```

```
//those people, who survive and does not have sibling
```

```
SELECT COUNT (*) AS SURVIVED_NO_SIBLING from titanic where survived=1 AND sibsp=0;
```

```
//those people, who died and does not have sibling
```

```
SELECT COUNT (*) AS Died_NO_SIBLING from titanic where survived=0 AND sibsp=0;
```

```
//removing the nan data from cabin and displaying updated list
```

```
SELECT * FROM titanic WHERE CABIN IS NOT NULL;
```

```
Delete from titanic WHERE CABIN IS NULL;
```

```
//male who died
```

```
SELECT COUNT (*) AS male_died FROM titanic WHERE sex='male' AND survived=0;
```

```
//female who died
```

```
SELECT COUNT (*) AS female_died FROM titanic WHERE sex='female' AND survived=0;
```