Guided Lab: Retrieving Data using Amazon S3 Select

Description

Amazon S3 Select is a powerful feature that enhances the capabilities of Amazon Simple Storage Service (S3) by offering efficient and selective data retrieval. S3 Select allows you to retrieve data from objects stored in your S3 buckets without downloading and processing the entire file. You can apply SQL-like queries to semi-structured data in JSON, CSV, and Parquet, enabling you to filter, transform, and aggregate data on the fly. This makes S3 Select an ideal tool for extracting valuable insights from massive datasets and improving data analytics.

S3 Select has substantial performance benefits as it minimizes data transfer and processing overhead, which reduces costs and speeds up data retrieval and analysis tasks. Whether working with log files, sensor data, or large datasets, Amazon S3 Select empowers you to access and process only the data you need efficiently, improving your data analytics and reducing the time and resources required for complex data manipulation tasks. It's also seamlessly integrated with AWS Glue and Amazon Athena, extending its utility for a comprehensive and streamlined data analysis experience.

In this hands-on lab, you'll explore the capabilities of Amazon S3 Select. This powerful feature efficiently retrieves, filters, and processes data from your S3 objects, making data analysis faster, cost-effective, and more precise. This practical experience will empower you to streamline your data workflows and easily extract valuable insights from your datasets.

Prerequisite

To guarantee a successful completion of this lab, you must possess prior experience creating Amazon S3 buckets and have a solid understanding of their core components. If you believe that your knowledge in this regard is lacking, we strongly advise you to consider taking the following to acquire the required proficiency:

- Creating an Amazon S3 bucket

Objectives

In this lab, you will:

- Execute SQL-like queries on your S3 data using S3 Select
- Understand the benefits of using S3 Select over traditional data retrieval methods

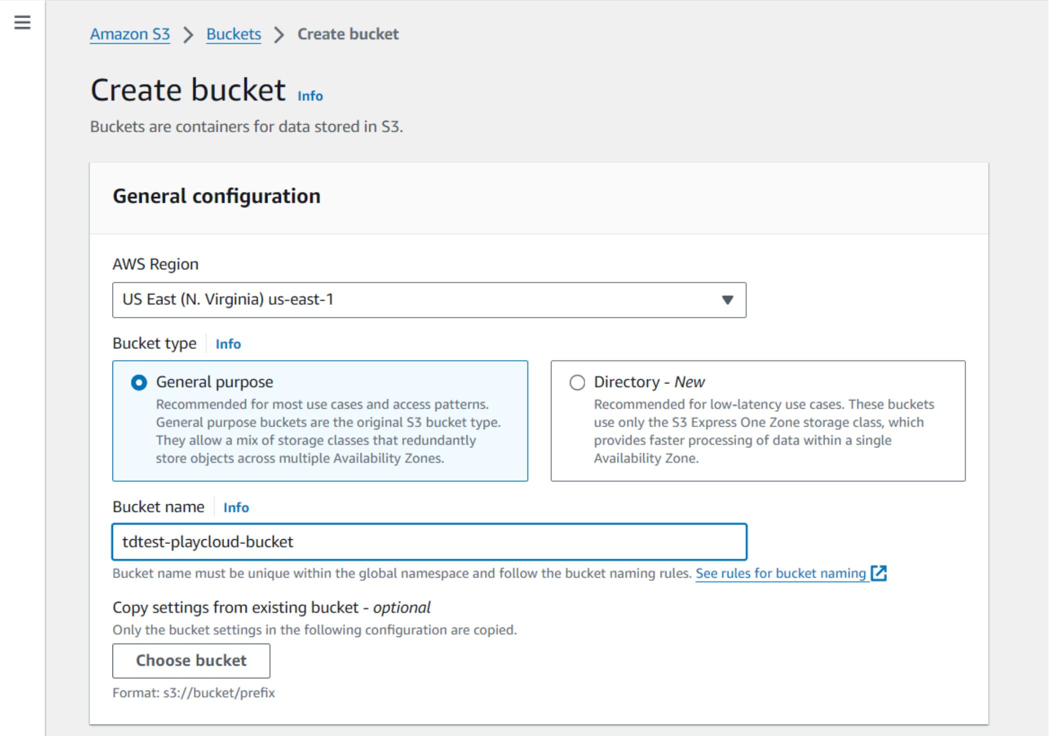
Subscribe to access AWS PlayCloud Labs

Lab Steps

Creating an S3 Bucket

1. Create an S3 bucket using the following configurations:

    - Add a **Bucket Name**
      Note: Provide a unique and descriptive name for your bucket. Remember that bucket names must be globally unique across all AWS accounts. Try a different combination if you receive an error message regarding your selected name. After that, set the AWS Region to N. Virginia to place the bucket to be created there.



Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region

US East (N. Virginia) us-east-1 ▼

Bucket type | Info

● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory - *New*
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name | Info

tdtest-playcloud-bucket

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

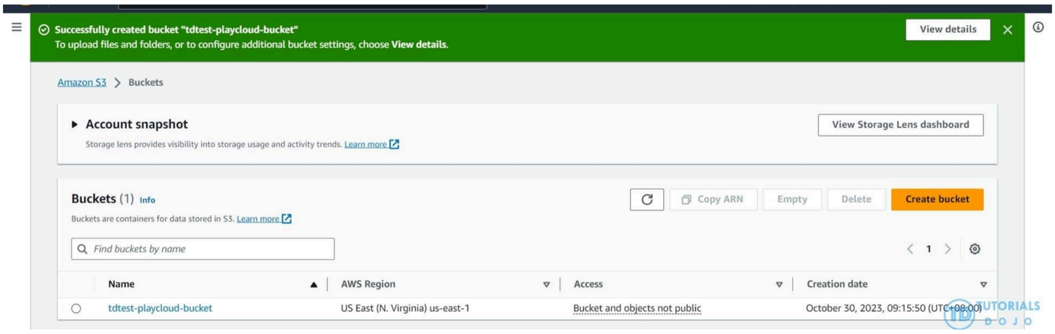
Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

2. Leave other settings at their default value.

3. Click on the “**Create bucket**” button. You should see a green notification that your bucket was created successfully.



Successfully created bucket "tdtest-playcloud-bucket"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

Amazon S3 > Buckets

Account snapshot

Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (1) Info

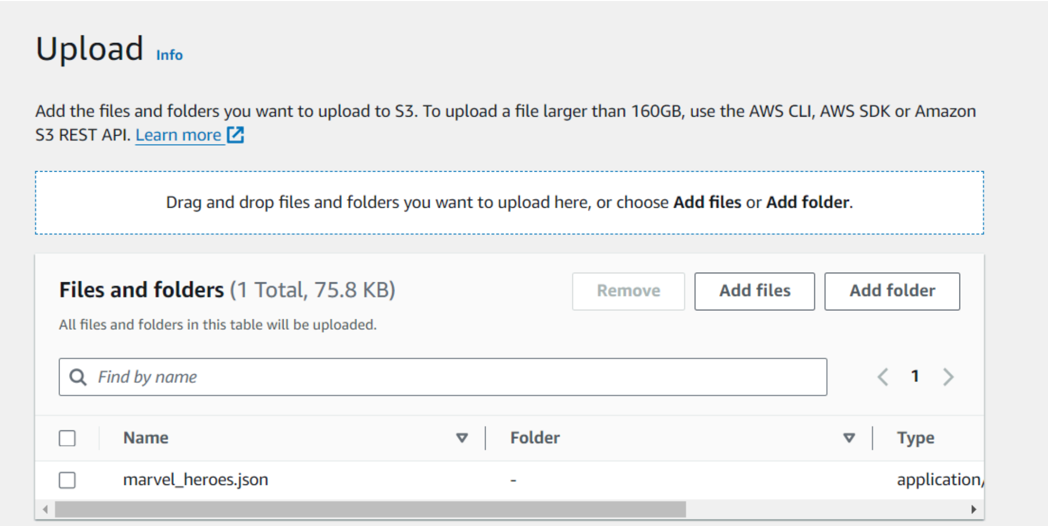
Buckets are containers for data stored in S3. Learn more

	Name	AWS Region	Access	Creation date
○	tdtest-playcloud-bucket	US East (N. Virginia) us-east-1	Bucket and objects not public	October 30, 2023, 09:15:50 (UTC+08:00)

Uploading a file into the S3 Bucket

1. Download and save the following JSON file into your computer and upload it to your S3 bucket.

marvel_heroes.json



Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more

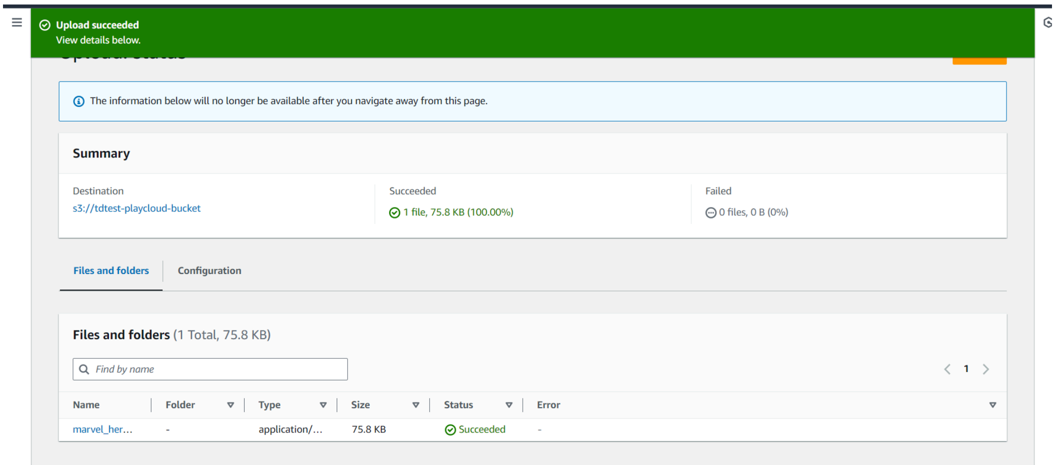
Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (1 Total, 75.8 KB)

All files and folders in this table will be uploaded.

□	Name	Folder	Type
□	marvel_heroes.json	-	application/

2. Scroll down to the page's bottom and click **Upload** to initiate the file upload process.



Upload succeeded
View details below.

The information below will no longer be available after you navigate away from this page.

Summary

Destination
s3://tdtest-playcloud-bucket

Succeeded
1 file, 75.8 KB (100.00%)

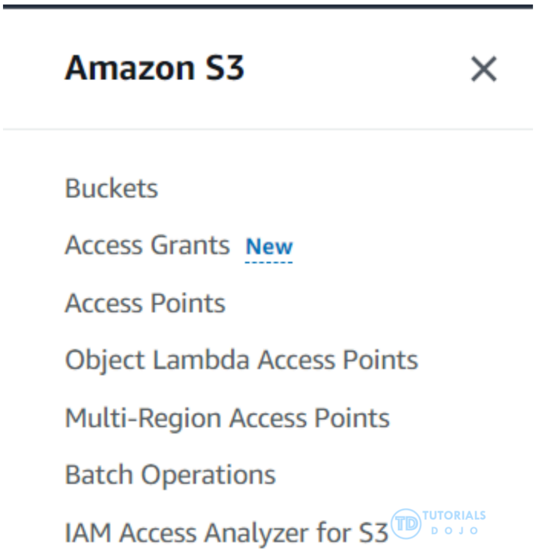
Failed
0 files, 0 B (0%)

Files and folders (1 Total, 75.8 KB)

Name	Folder	Type	Size	Status	Error
marvel_her...	-	application/...	75.8 KB	Succeeded	-

Retrieving Data using Amazon S3 Select

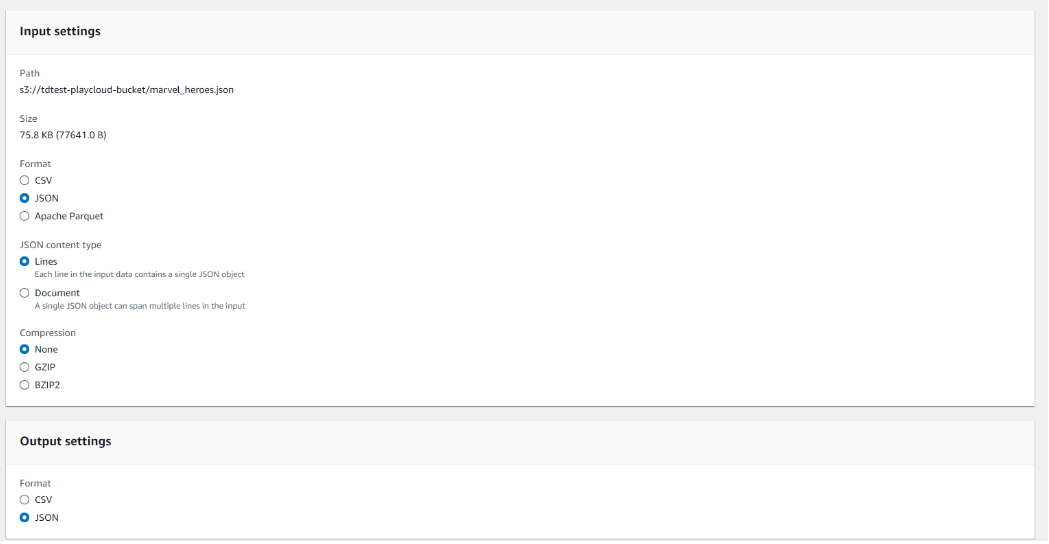
1. In the left navigation pane, choose **Buckets**.



Amazon S3
Buckets
Access Grants New
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3

2. Choose the bucket containing the object that you want to select content from, and then choose the name of the object **JSON** file that you've uploaded. Then, on the **Actions** drop-down, click **Query with S3 Select**.

3. Select **JSON** as the **Input and Output format**.



Input settings

Path
s3://tdtest-playcloud-bucket/marvel_heroes.json

Size
75.8 KB (77641.0 B)

Format
○ CSV
● JSON
○ Apache Parquet

JSON content type
● Lines
Each line in the input data contains a single JSON object
○ Document
A single JSON object can span multiple lines in the input

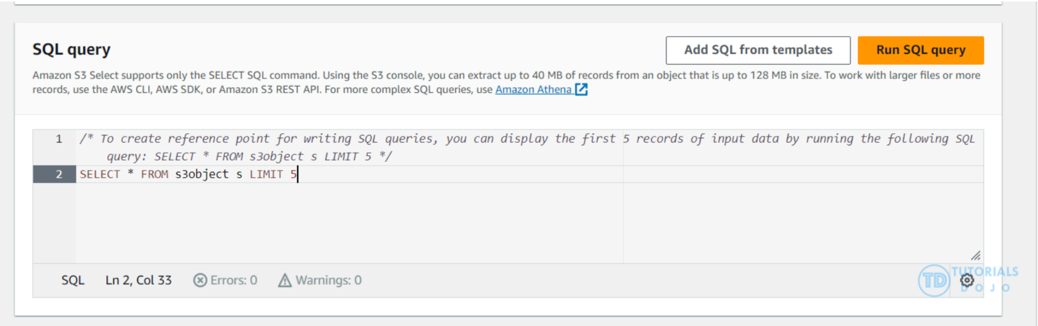
Compression
● None
○ GZIP
○ BZIP2

Output settings

Format
○ CSV
● JSON

4. To extract records from the chosen object, under **SQL query**, run the SELECT SQL query provided by the s3.

```
SELECT * FROM s3object s LIMIT 5
```

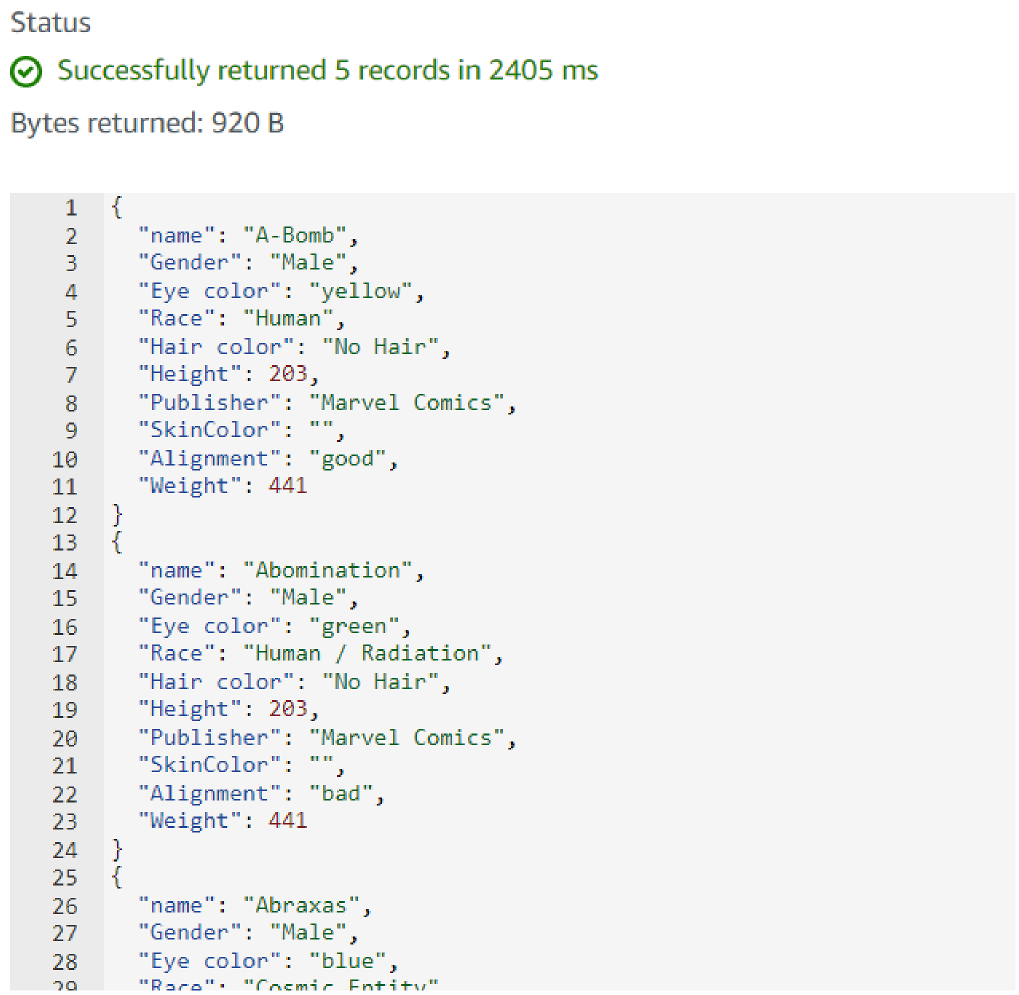


SQL query

Amazon S3 Select supports only the SELECT SQL command. Using the S3 console, you can extract up to 40 MB of records from an object that is up to 128 MB in size. To work with larger files or more records, use the AWS CLI, AWS SDK, or Amazon S3 REST API. For more complex SQL queries, use Amazon Athena

```
/* To create reference point for writing SQL queries, you can display the first 5 records of input data by running the following SQL
    query: SELECT * FROM s3object s LIMIT 5 */
SELECT * FROM s3object s LIMIT 5
```

The result should display the first 5 rows of your **.csv** file.



Status

Successfully returned 5 records in 2405 ms

Bytes returned: 920 B

```
{
  "name": "A-Bomb",
  "Gender": "Male",
  "Eye color": "yellow",
  "Race": "Human",
  "Hair color": "No Hair",
  "Height": 203,
  "Publisher": "Marvel Comics",
  "SkinColor": "",
  "Alignment": "good",
  "Weight": 441
}
{
  "name": "Abomination",
  "Gender": "Male",
  "Eye color": "green",
  "Race": "Human / Radiation",
  "Hair color": "No Hair",
  "Height": 203,
  "Publisher": "Marvel Comics",
  "SkinColor": "",
  "Alignment": "bad",
  "Weight": 441
}
{
  "name": "Abraxas",
  "Gender": "Male",
  "Eye color": "blue",
  "Race": "Cosmic Entity",
```

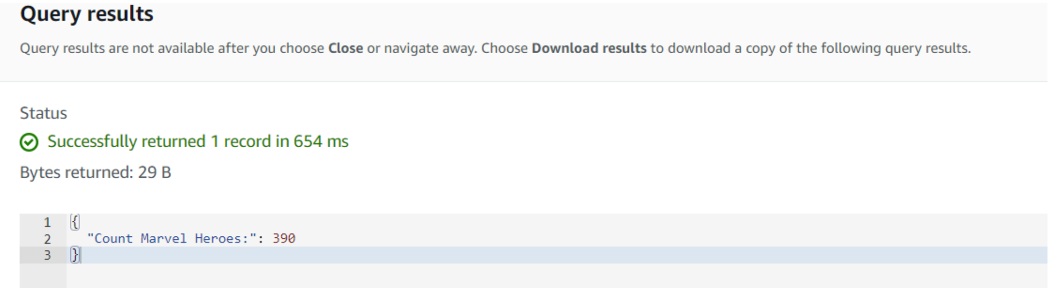
Using Aggregate Functions in Amazon S3 Select Query

Amazon S3 Select supports the following aggregate functions. You can use these aggregate functions in combination with the SELECT statement to process and analyze your data directly in Amazon S3 without having to retrieve the entire dataset.

1. Now, let's explore the Aggregate functions by using SQL Query. The first function we will use is **COUNT**, which will count the total number of records of the S3 file. Use the following query to run the function:

```
SELECT COUNT(s.gender) AS "Your Alias" FROM s3object s
```

The expected result should be as follows:



Query results

Query results are not available after you choose **Close** or navigate away. Choose **Download results** to download a copy of the following query results.

Status

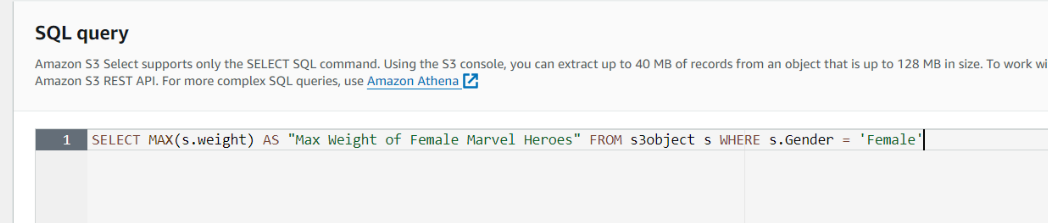
Successfully returned 1 record in 654 ms

Bytes returned: 29 B

```
{
  "Count Marvel Heroes:": 390
}
```

2. The next function is **MAX**, which will get the maximum value of a given attribute (e.g. weight) and also you can combine it with the **WHERE** clause to make it more specific. Use the following query to run the function:

```
SELECT MAX(s.your_attribute) AS "Your Alias" FROM
s3object s WHERE condition
```

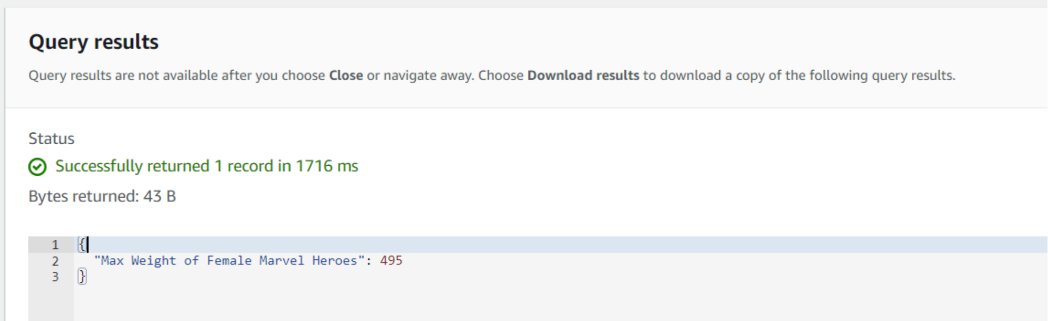


SQL query

Amazon S3 Select supports only the SELECT SQL command. Using the S3 console, you can extract up to 40 MB of records from an object that is up to 128 MB in size. To work wi
Amazon S3 REST API. For more complex SQL queries, use Amazon Athena

```
SELECT MAX(s.weight) AS "Max Weight of Female Marvel Heroes" FROM s3object s WHERE s.Gender = 'Female'
```

The expected result should be as follows:



Query results

Query results are not available after you choose **Close** or navigate away. Choose **Download results** to download a copy of the following query results.

Status

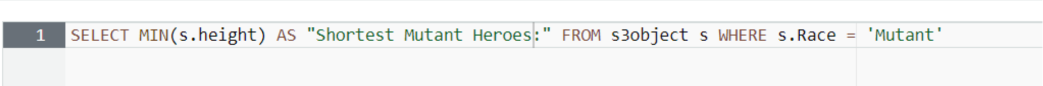
Successfully returned 1 record in 1716 ms

Bytes returned: 43 B

```
{
  "Max Weight of Female Marvel Heroes": 495
}
```

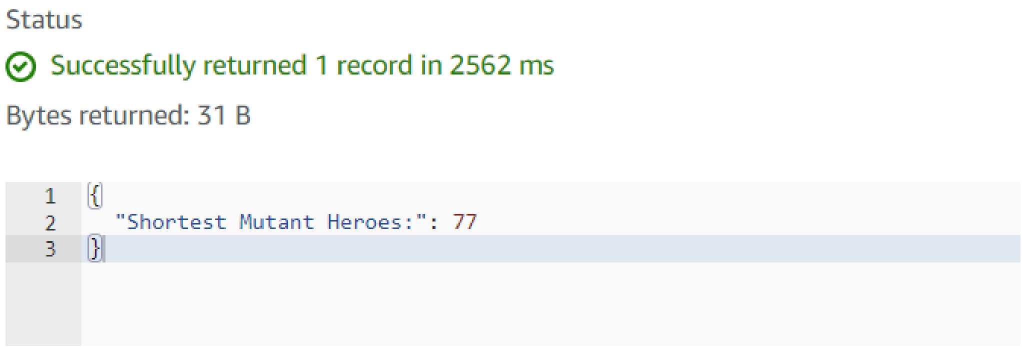
3. The third function is **MIN**, which will get the minimum value of a given attribute (e.g. height). Use the following query to run the function:

```
SELECT MIN(s.your_attribute) AS "Your Alias" FROM
s3object s WHERE condition
```



```
SELECT MIN(s.height) AS "Shortest Mutant Heroes:" FROM s3object s WHERE s.Race = 'Mutant'
```

The expected result should be as follows:



Status

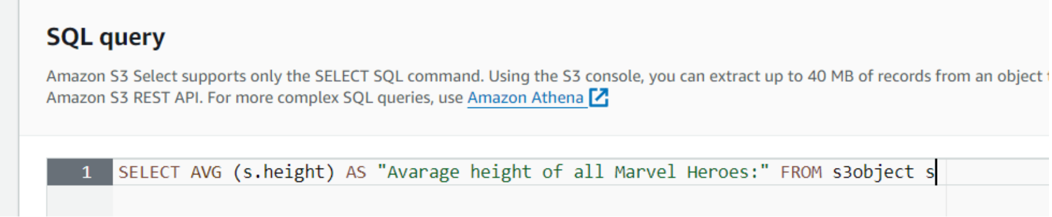
Successfully returned 1 record in 2562 ms

Bytes returned: 31 B

```
{
  "Shortest Mutant Heroes:": 77
}
```

5. The fourth function is **AVG**, which will return the average number of your selected attribute. Use the following query to run the function:

```
SELECT AVG (s.your_attribute) AS "Your Alias" FROM
s3object s
```

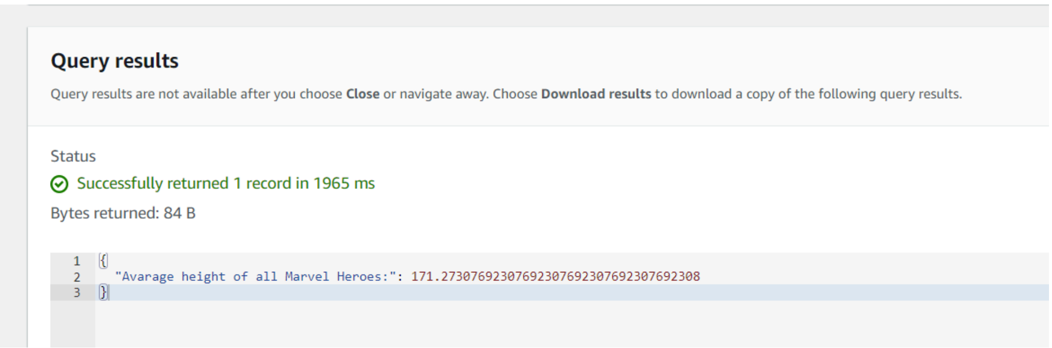


SQL query

Amazon S3 Select supports only the SELECT SQL command. Using the S3 console, you can extract up to 40 MB of records from an object
Amazon S3 REST API. For more complex SQL queries, use Amazon Athena

```
SELECT AVG (s.height) AS "Avarage height of all Marvel Heroes:" FROM s3object s
```

The expected result should be as follows:



Query results

Query results are not available after you choose **Close** or navigate away. Choose **Download results** to download a copy of the following query results.

Status

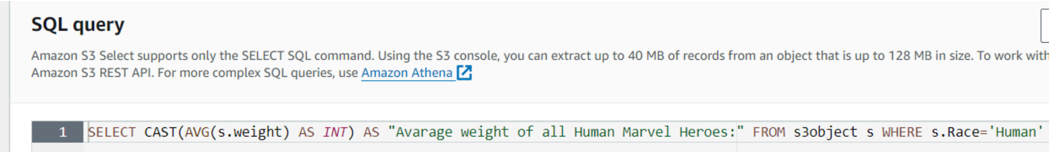
Successfully returned 1 record in 1965 ms

Bytes returned: 84 B

```
{
  "Avarage height of all Marvel Heroes:": 171.27307692307692307692307692307692308
}
```

5. Lastly, we will use the **CAST** conversion function with the AVG aggregate function, which will calculate the average Human Marvel hero weight and convert it to an INTEGER data type. Use the following query to run the function:

```
SELECT CAST(AVG(s.your_attribute) AS INT) AS "Your
Alias" FROM s3object s WHERE condition
```

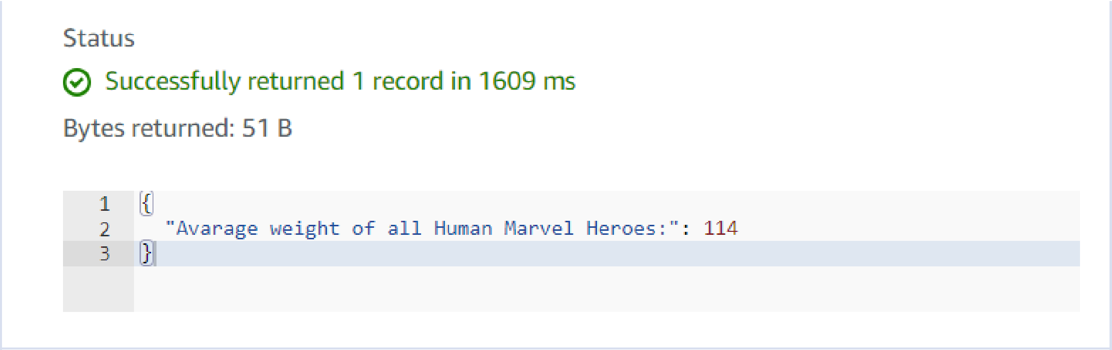


SQL query

Amazon S3 Select supports only the SELECT SQL command. Using the S3 console, you can extract up to 40 MB of records from an object that is up to 128 MB in size. To work with
Amazon S3 REST API. For more complex SQL queries, use Amazon Athena

```
SELECT CAST(AVG(s.weight) AS INT) AS "Avarage weight of all Human Marvel Heroes:" FROM s3object s WHERE s.Race='Human'
```

And the expected result should be as follows:



Status

Successfully returned 1 record in 1609 ms

Bytes returned: 51 B

```
{
  "Avarage weight of all Human Marvel Heroes:": 114
}
```