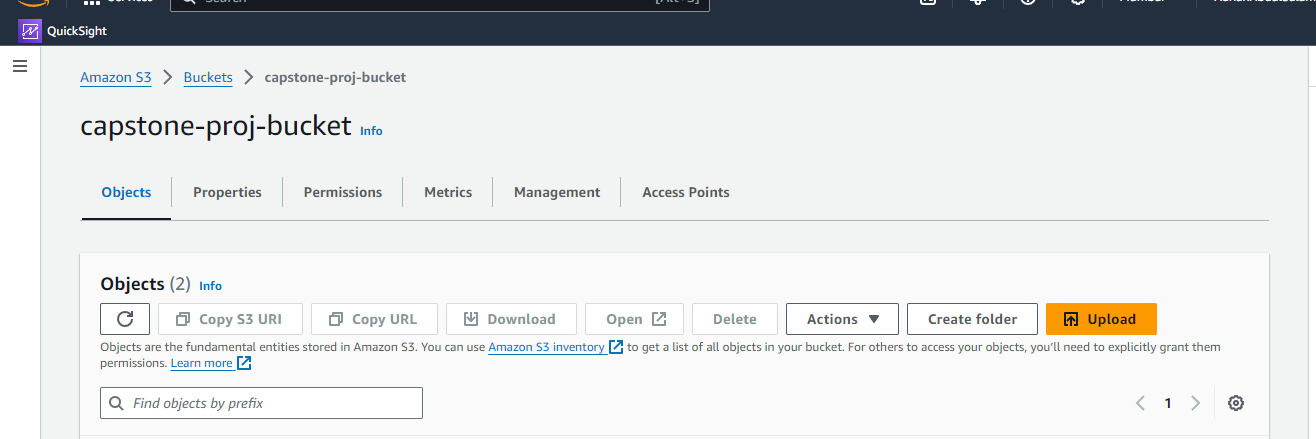
GANIT TIMES

API Provided: <https://inshorts.com/api/en/news?category=all_news>

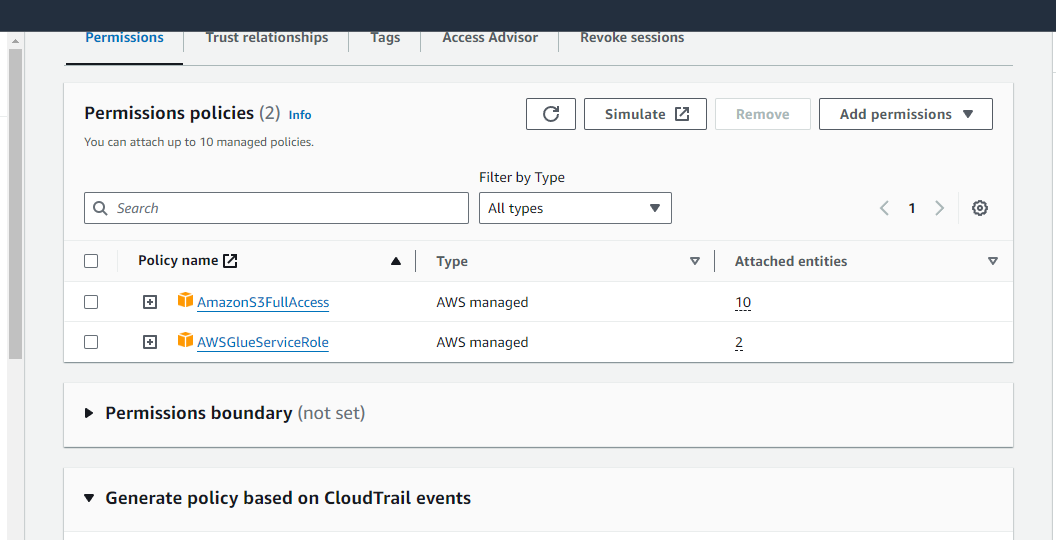
Objective: Do data analysis, build a pipeline to extract the daily trending news and have a Ganit newspaper built

Data Fetching and cleaning part:

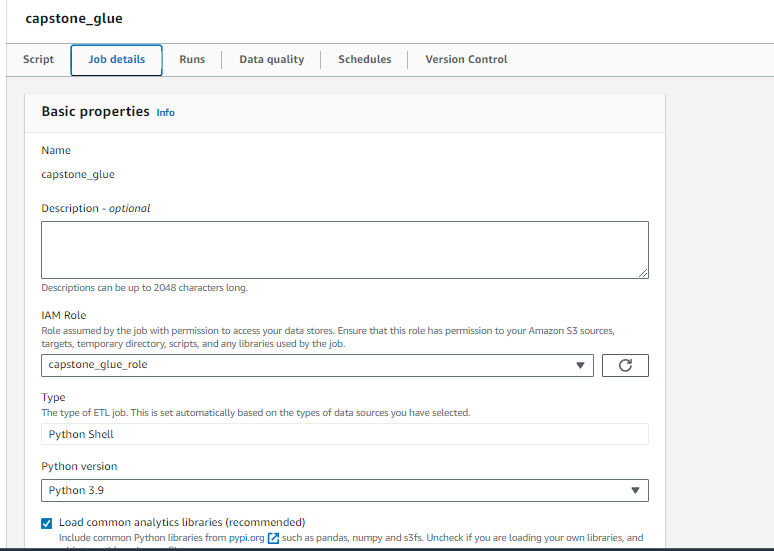
Step 1: Create a s3 bucket in Amazon Web Services (AWS) to store the data that is fetched from the API.



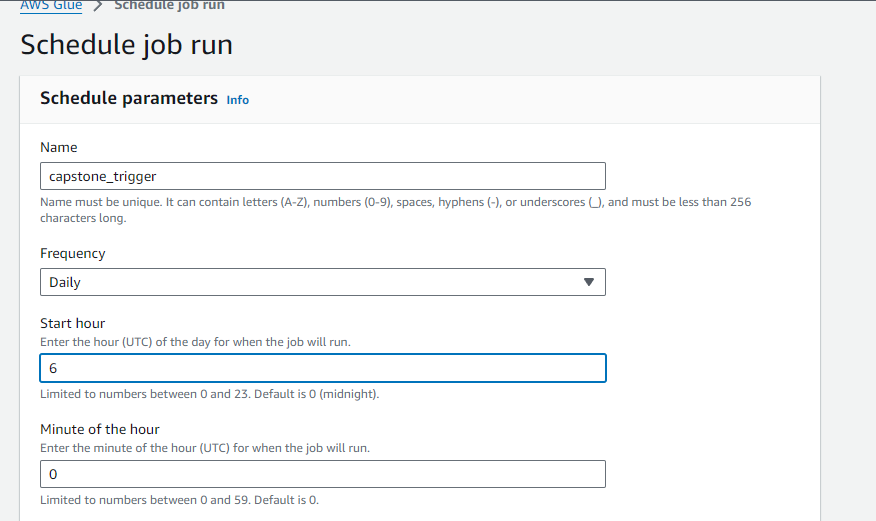
Step 2: Create an IAM role for the S3 bucket giving necessary permissions



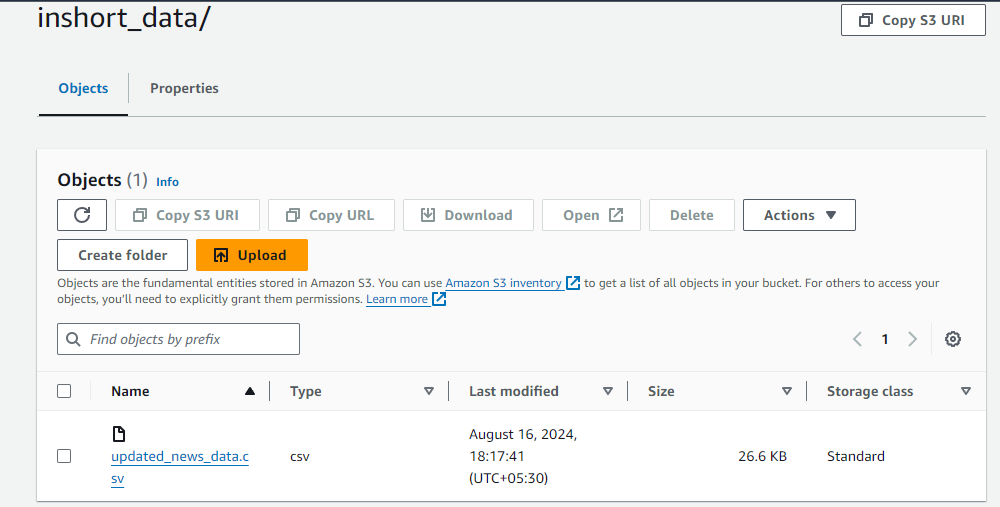
Step 3: Write a Glue function to fetch the news data.



Step 4: Create a schedule for the glue job to run at 11:30 IST.

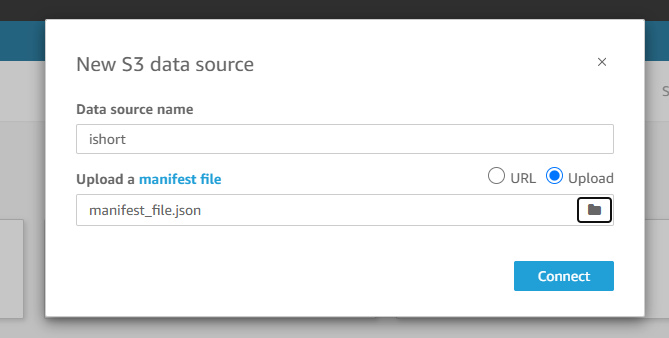


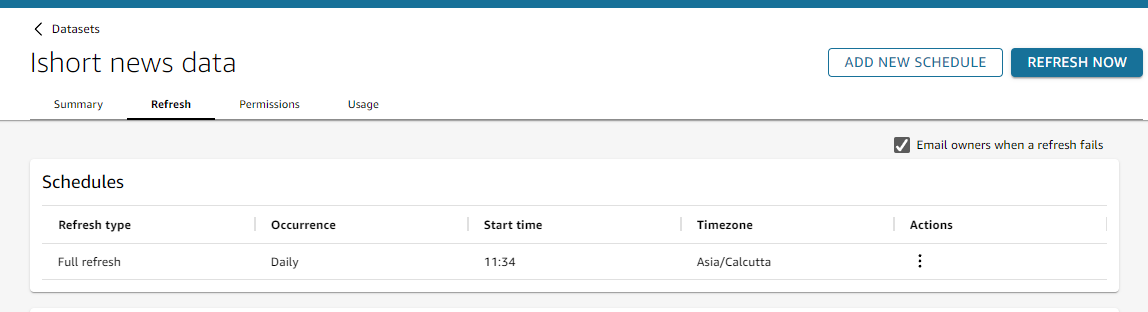
Step 5: The glue function will clean the fetched news data (remove unwanted and unnecessary columns) and store it in an S3 bucket.



We have formulated two approaches to this problem-statement. One is displaying the dashboard in Amazon Quicksight and the other in Tableau. The steps until uploading news data into S3 bucket are the same for both the approaches.

**Approach 1 : Amazon Quicksight**

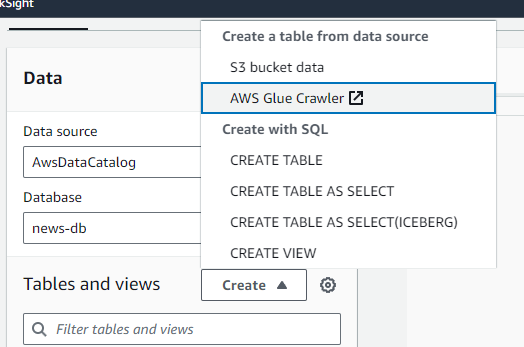
Step 1: Connect S3 data source to Quicksight.

Step 2: Schedule the dataset to refresh at 11:34 IST.

Step 3: Visualize the data and create the necessary calculated fields to create a newspaper dashboard.

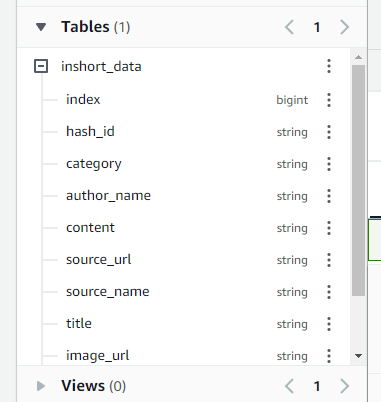
**Approach 2 : Tableau Desktop**

Step 1: Setting up AWS Crawler for creating table in Athena.

****

**A screenshot of a computer

Description automatically generated**Step 2: Providing Data Source as S3 and connecting it to our S3 bucket which stores the daily fetched data.

****Step 3: After Scheduling Crawler to refresh daily. The table is created in Athena.

Step 4: In the tableau, go to Connect A Data Source, then use ODBC to connect Athena to tableau.

**A screenshot of a computer

Description automatically generated**

Step 5: The table can then be visualized in Tableau.

**A screenshot of a computer

Description automatically generated**

Step 6: Visualize the data and create the necessary calculated fields to create a newspaper dashboard.

