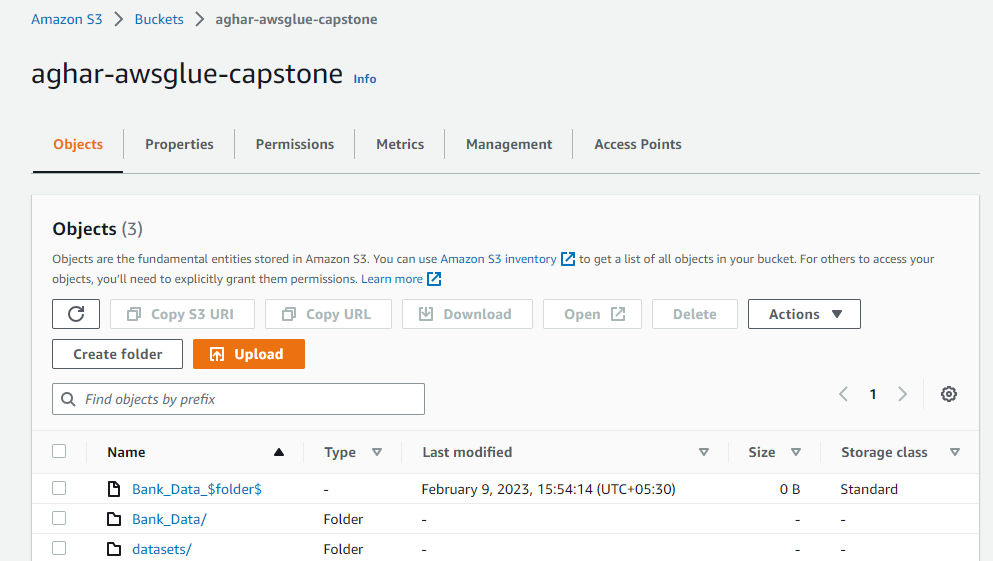
**Step-By-step process with images**

**AIM: Create an orchestration pipeline for training and inference in spark using AWS Glue**

step 1: created a s3 bucket and stored bank\_dataset.



Step 2: created crawler name Aghar\_Bankdata\_1 for loading the dataset from s3

Graphical user interface, application

Description automatically generated

Step 3:The result from step 2 is loaded data from s3 is stored in database table created in glue

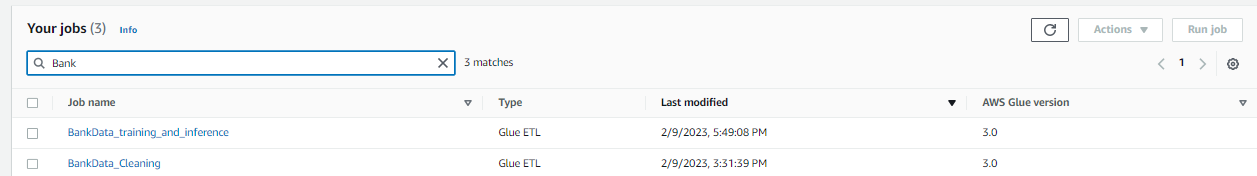
Table Name: raw\_bankdata

Graphical user interface, text, application, email

Description automatically generated

Step 4: created a job for cleaning the dataset.

Job name: BankData\_cleaning. (the data for cleaning is loaded from table created using crawler)



Step 5: The cleaned data is automatically stored to s3 using configuration in notebook.

Graphical user interface, text, application, email

Description automatically generated

Step 6: created crawler name Aghar\_Bankdata\_2 for loading the cleaned\_dataset from s3.

Graphical user interface, application

Description automatically generated

Step 7: The result from step 6 is loaded cleaned data from s3 is stored in database table created in glue

Table Name: cleaned\_bankdata

Graphical user interface, text, application, email

Description automatically generated

Step 8: created a job for preprocessing the data and inferences.

Job name: BankData\_training\_and\_inference

Graphical user interface, text, application

Description automatically generated

Step 9: The preprocessed(train and test) data and output metrics(inference data) is automatically stored to s3 using configuration in notebook.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Step 10: created a conditional and on demand trigger for the jobs and crawlers.

Graphical user interface, text, application

Description automatically generated

Step 11: created a workflow and added the triggers in order and allow it run and checked whether it is succeeded.

A picture containing graphical user interface

Description automatically generated