**AMOS Group 6**

**Assignment - 2**   
Osborne Lopes  
Akshitha Pathania  
Smithi Parthiban  
Manimanya Reddy

1. **Data sources -**
2. <https://www.cfainstitute.org/membership/professional-development/refresher-readings/time-series-analysis>
3. Extracted PDF Files

**II. Details on the assignment -**

1. Python libraries used - Selenium, Beautifulsoup, Pandas
2. Tools used - Grobid, PYPDF, PDFMiner, SQLAlchemy  
   Databases - Snowflake
3. Cloud software - AWS S3

Platforms the assignment is performed on

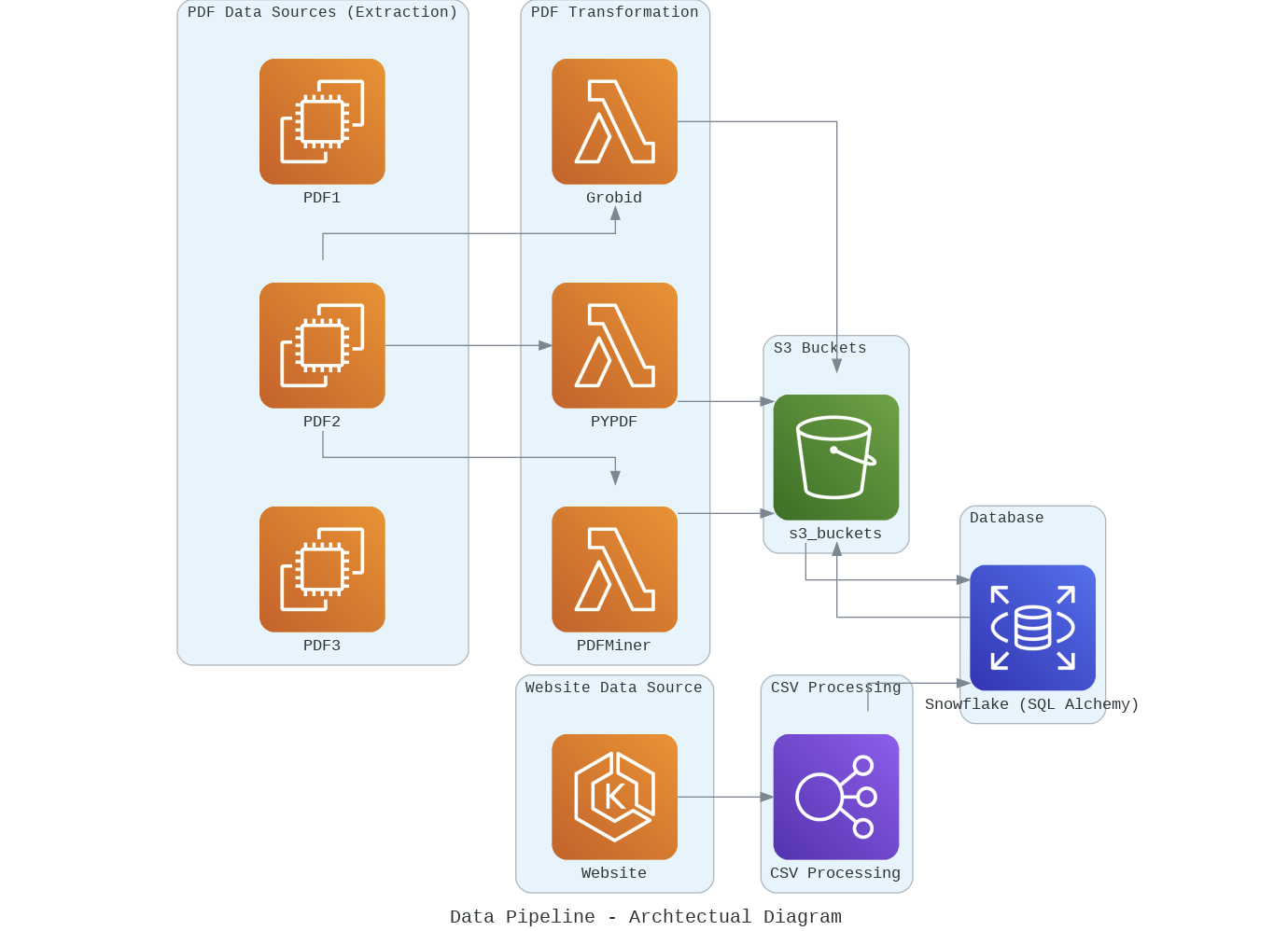
-Jupyter Notebooks

-Google Collab Notebooks

-Local CLI

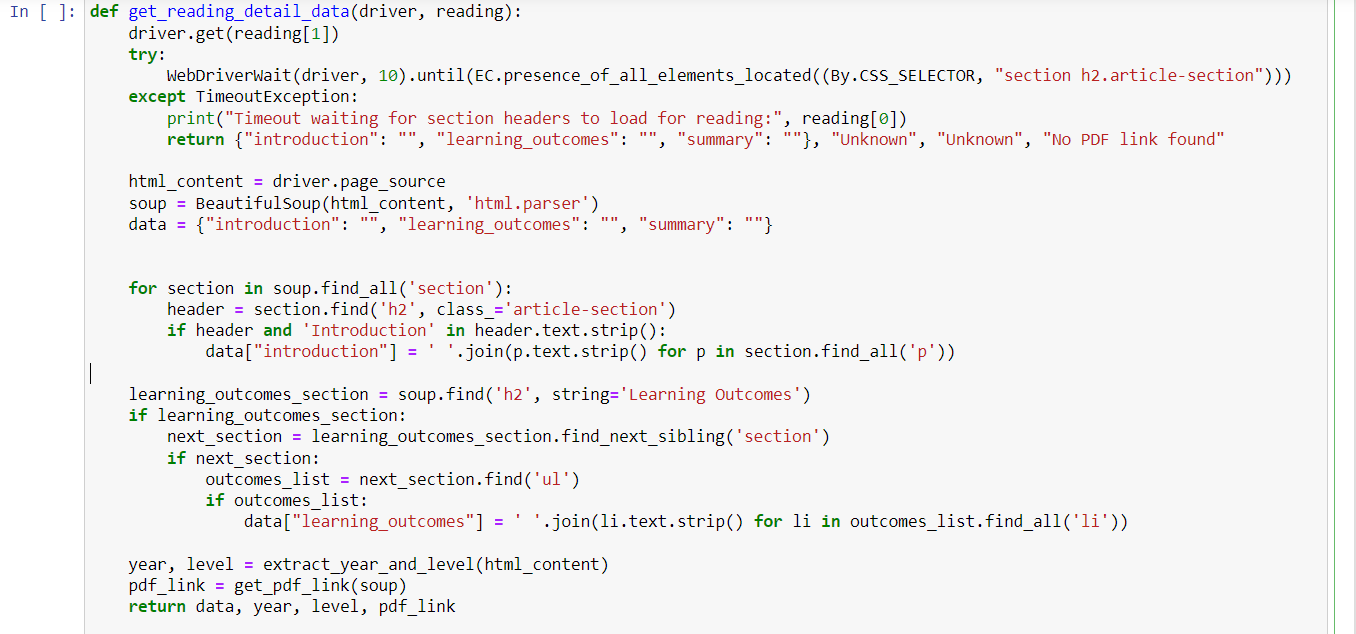
-AWS - Amazon Web Service

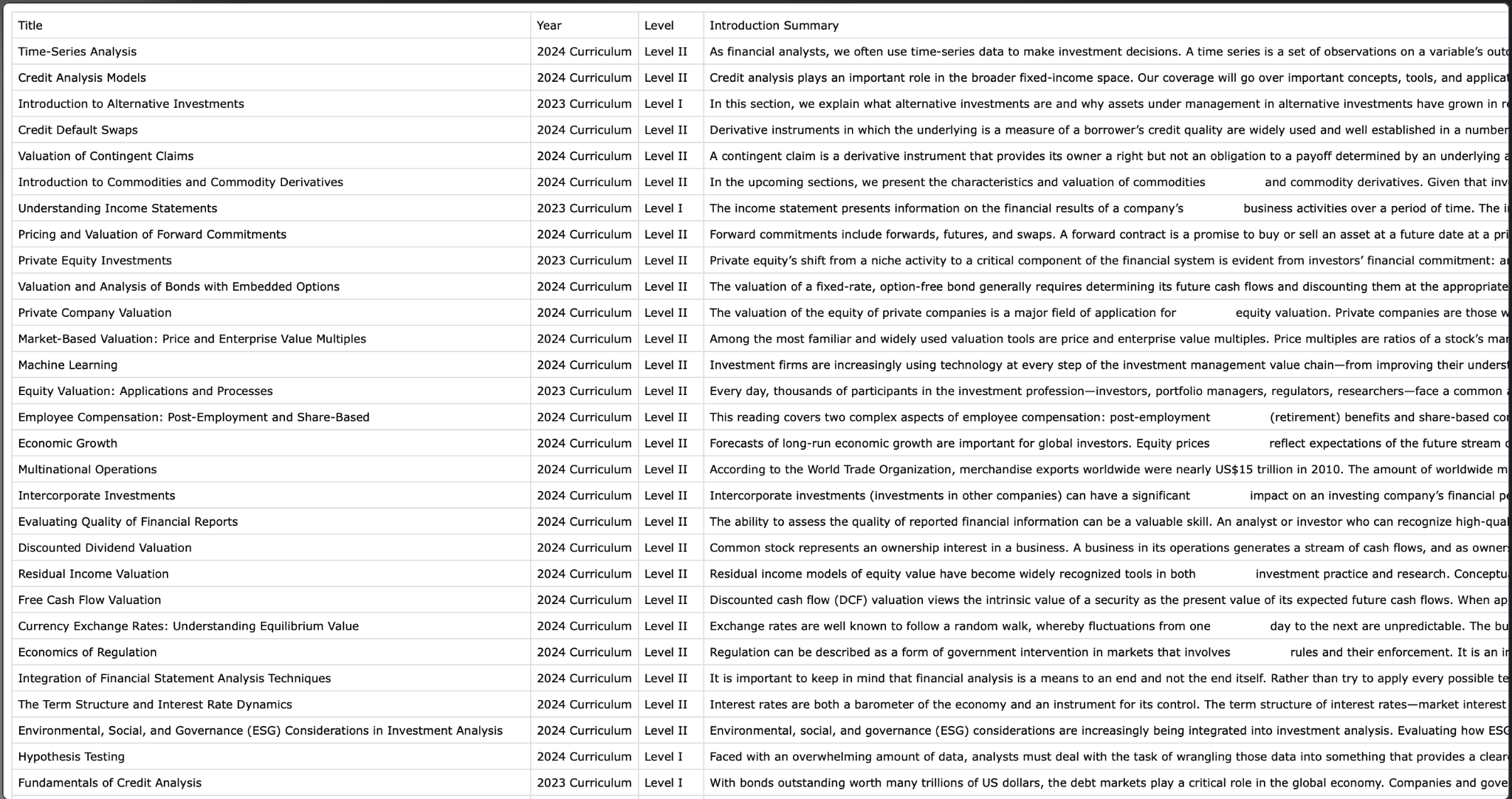
**Architectural Diagram -**



**III. Data Pipeline Orchestration**  
**3.1** **Step 1 - Web Scraping and Dataset Creation:**

Web scraping - Python libraries (Selenium and Beautifulsoup, pandas)   
224 Pages of Links were extracted

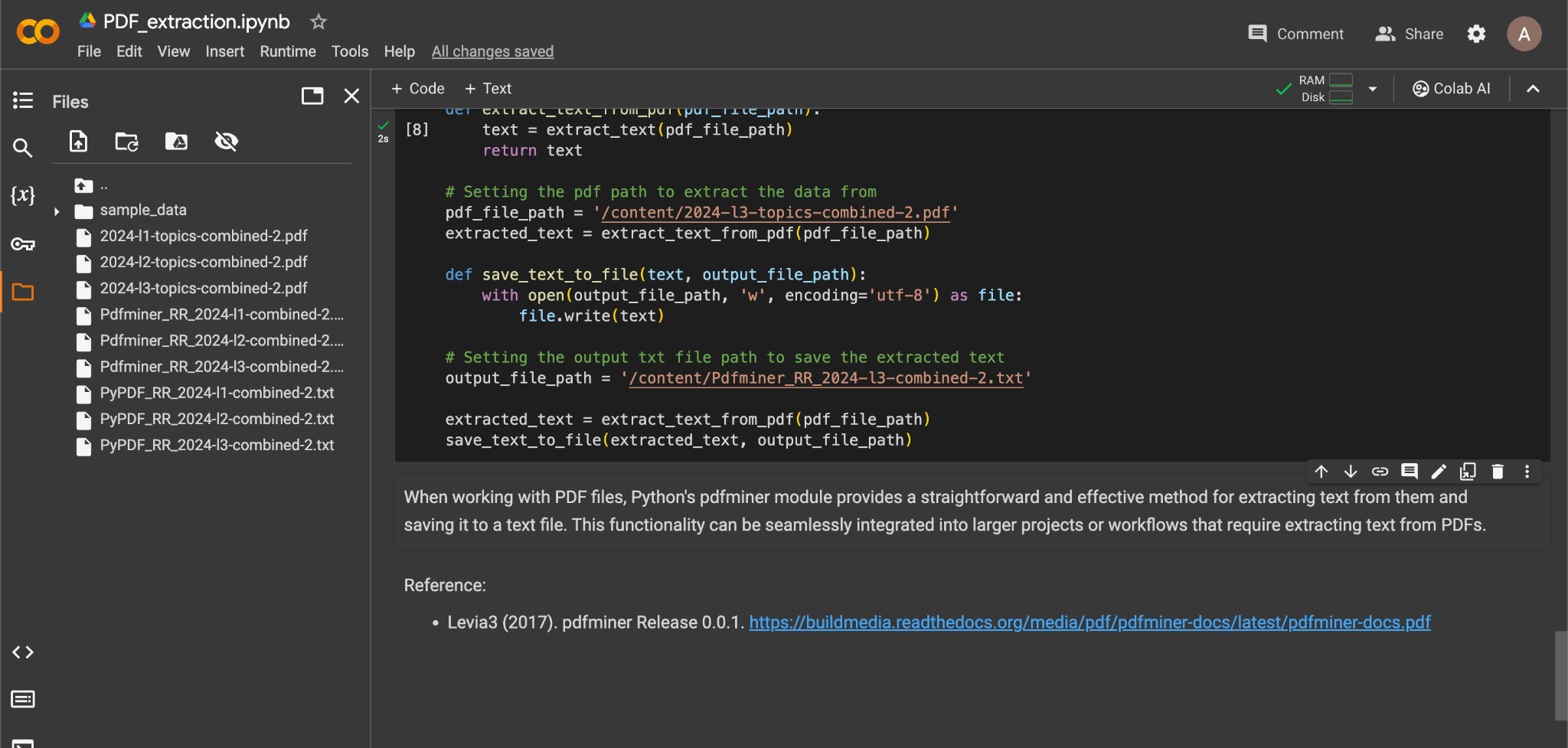


**Python notebook and Dataset available to web scrape the information:**  
  


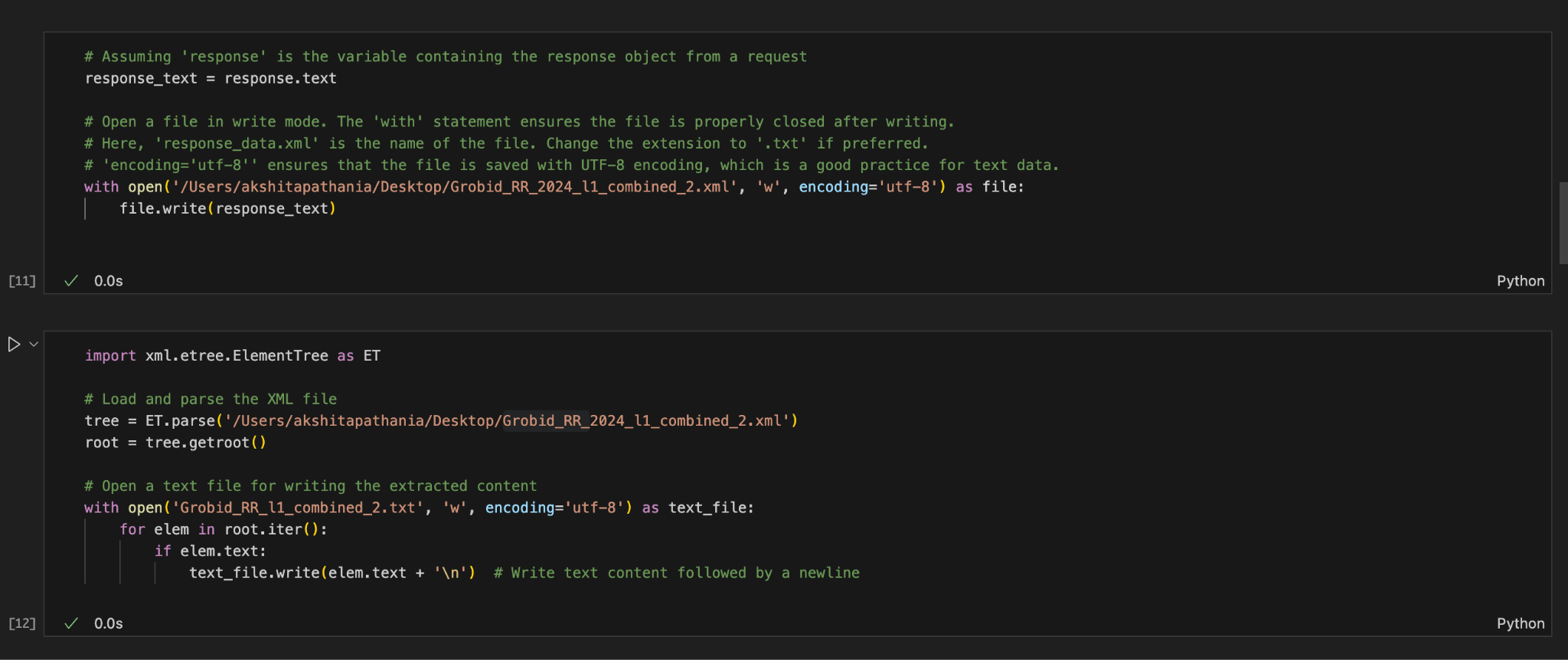
**3.2 PDFS**Tools used to extract data from provided PDFs

* PYPDF2
* PDFMiner
* GROBID

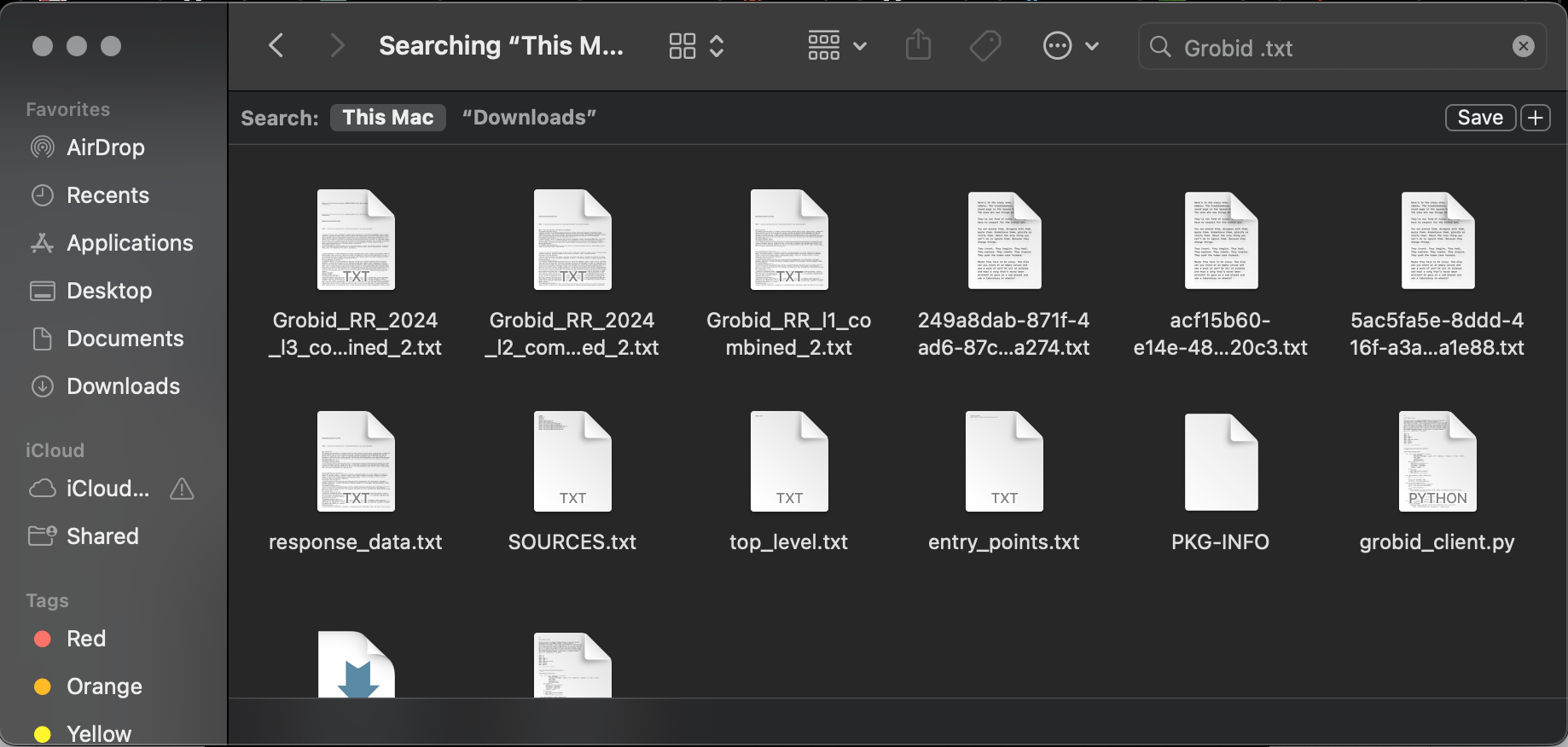
In the screenshot below, we can see the downloaded TXT files of PDFMiner and PyPDF files of the three given PDF files on the left.



For Grobid, we had to run the file locally on VSCode and download the files locally onto our system.



Below we can see the downloaded Txt files on the system.

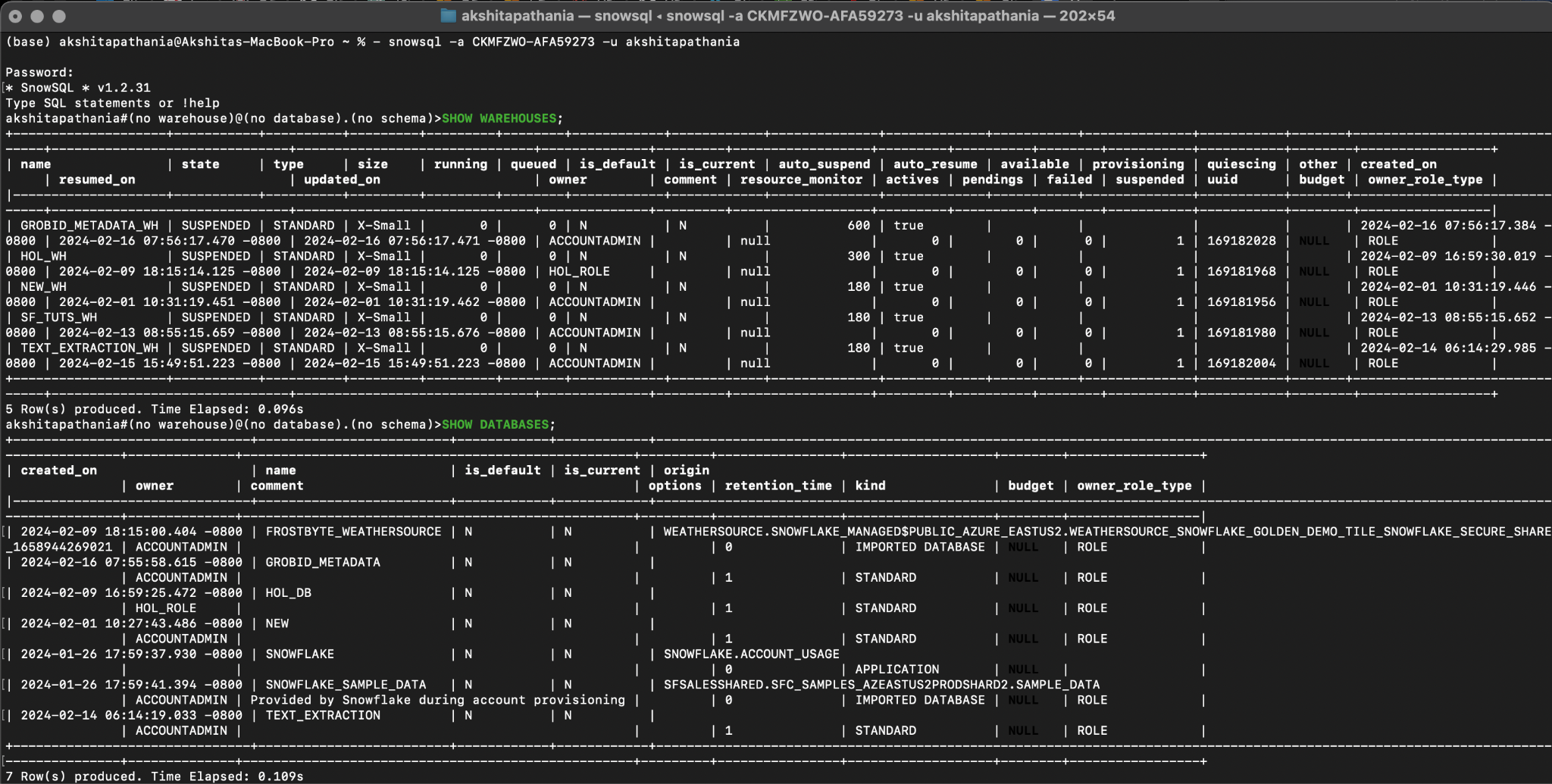


**IV. Database Upload:**

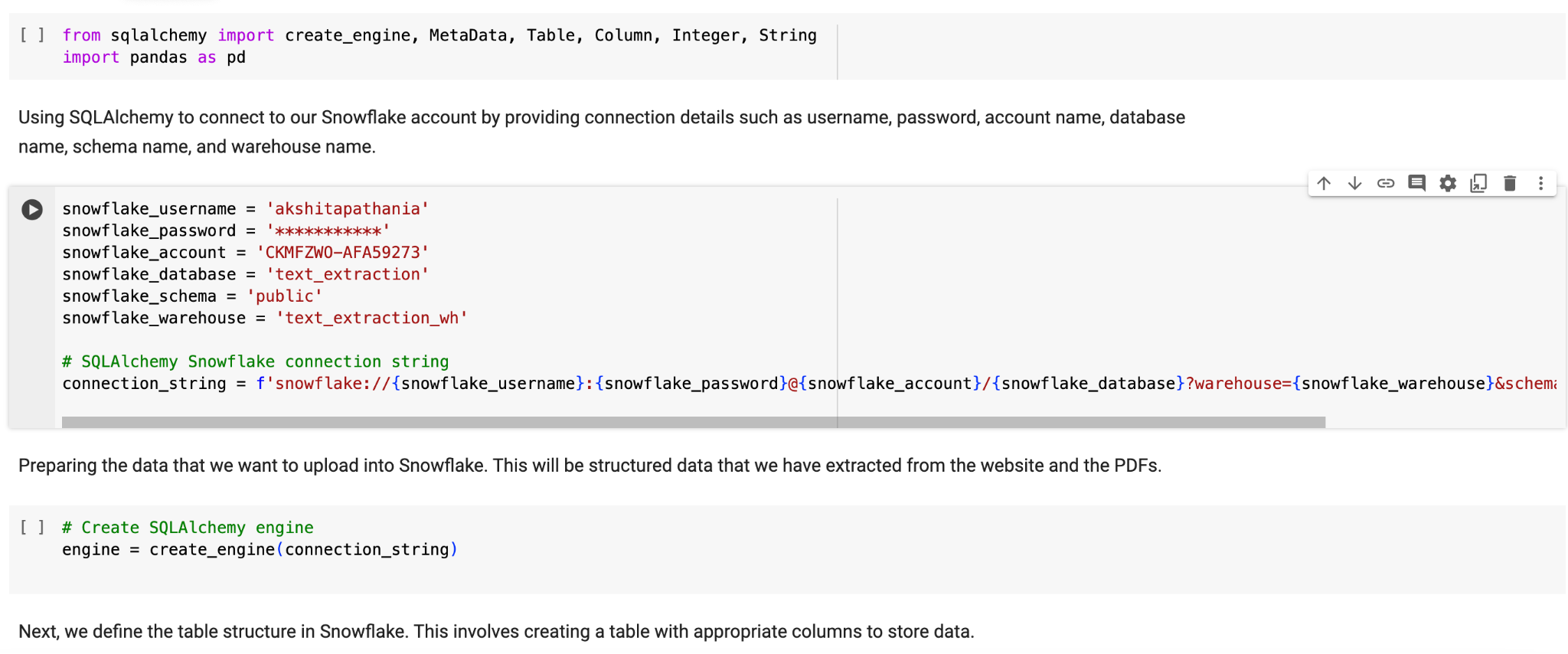
4.1 Uploading the Extracted (Webscrapped CSV file from 3.1) into Snowflake via

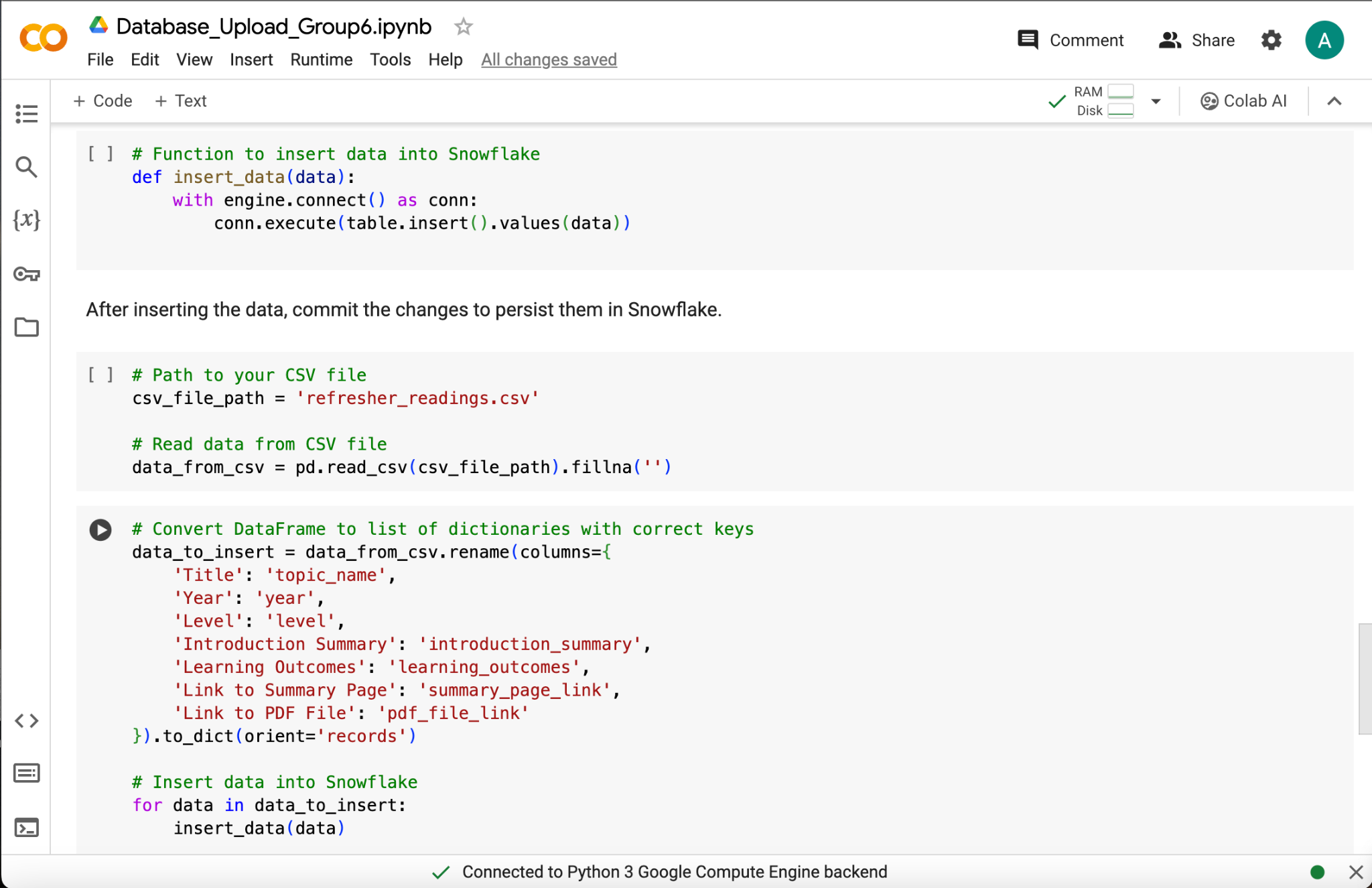
SQLAlchemy

* Terminal Commands where the Warehouses and Databases are shown after being created

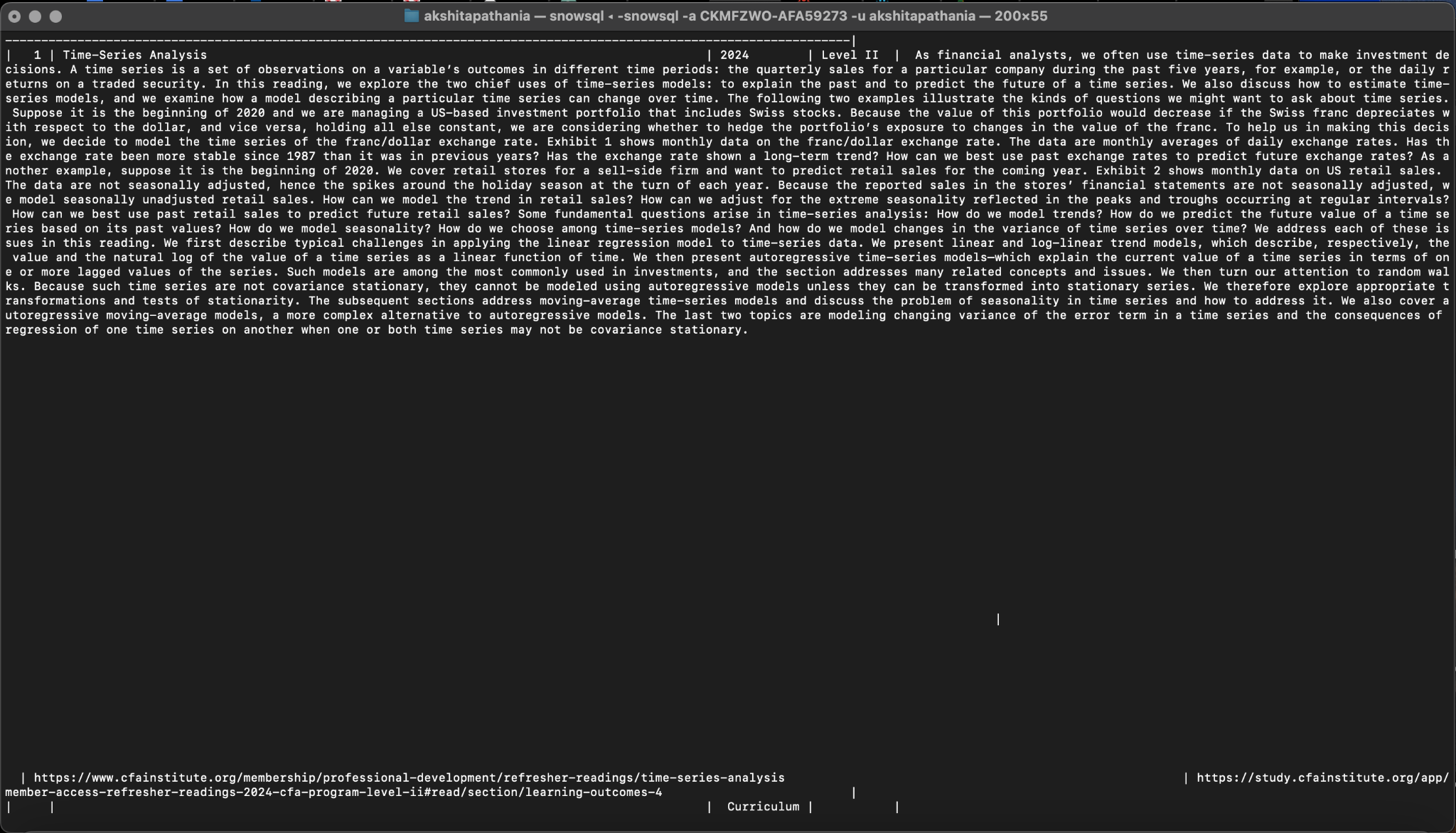


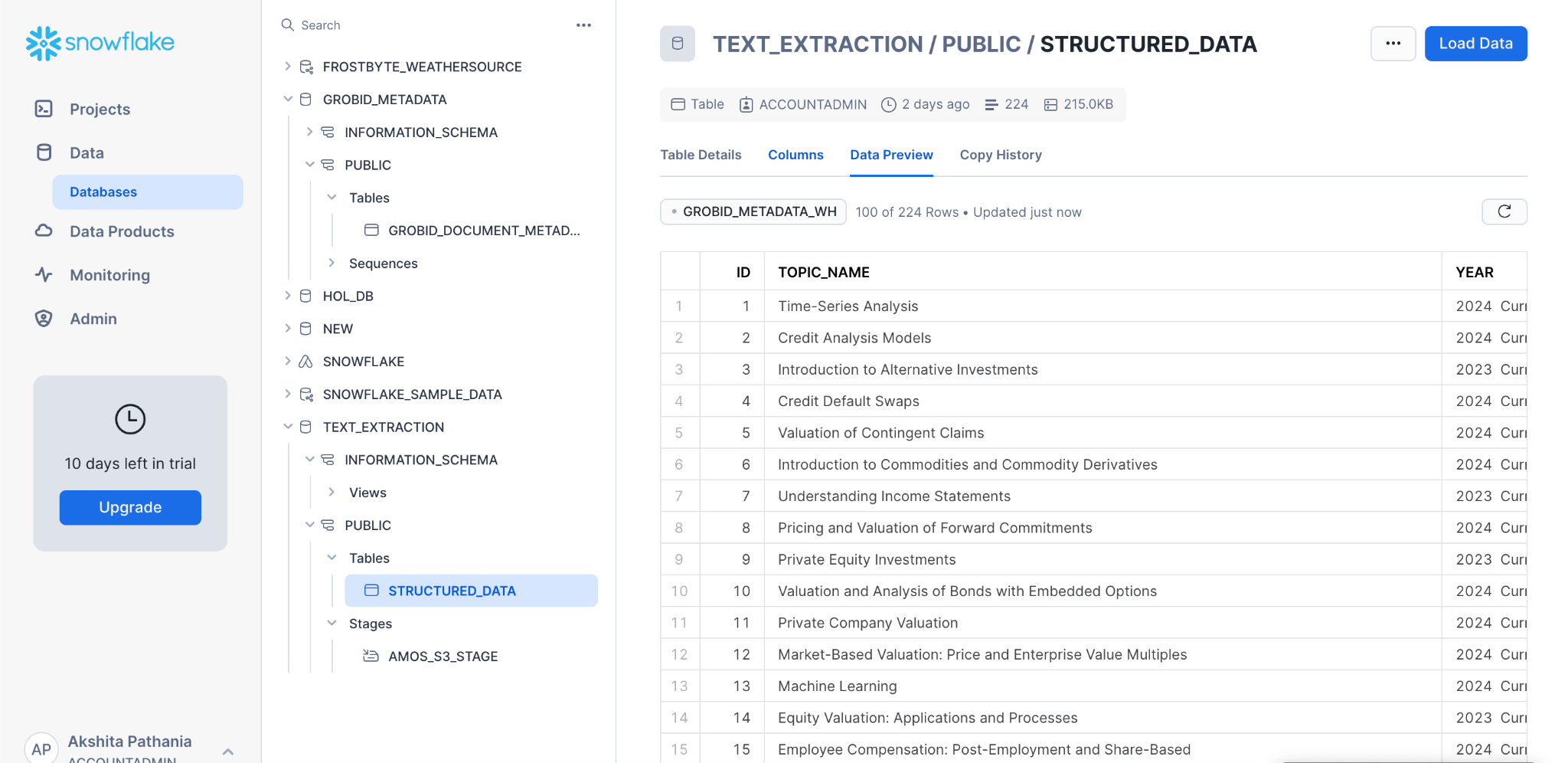
* Using SQLAlchemy to connect to Snowflake in the Google Colab file



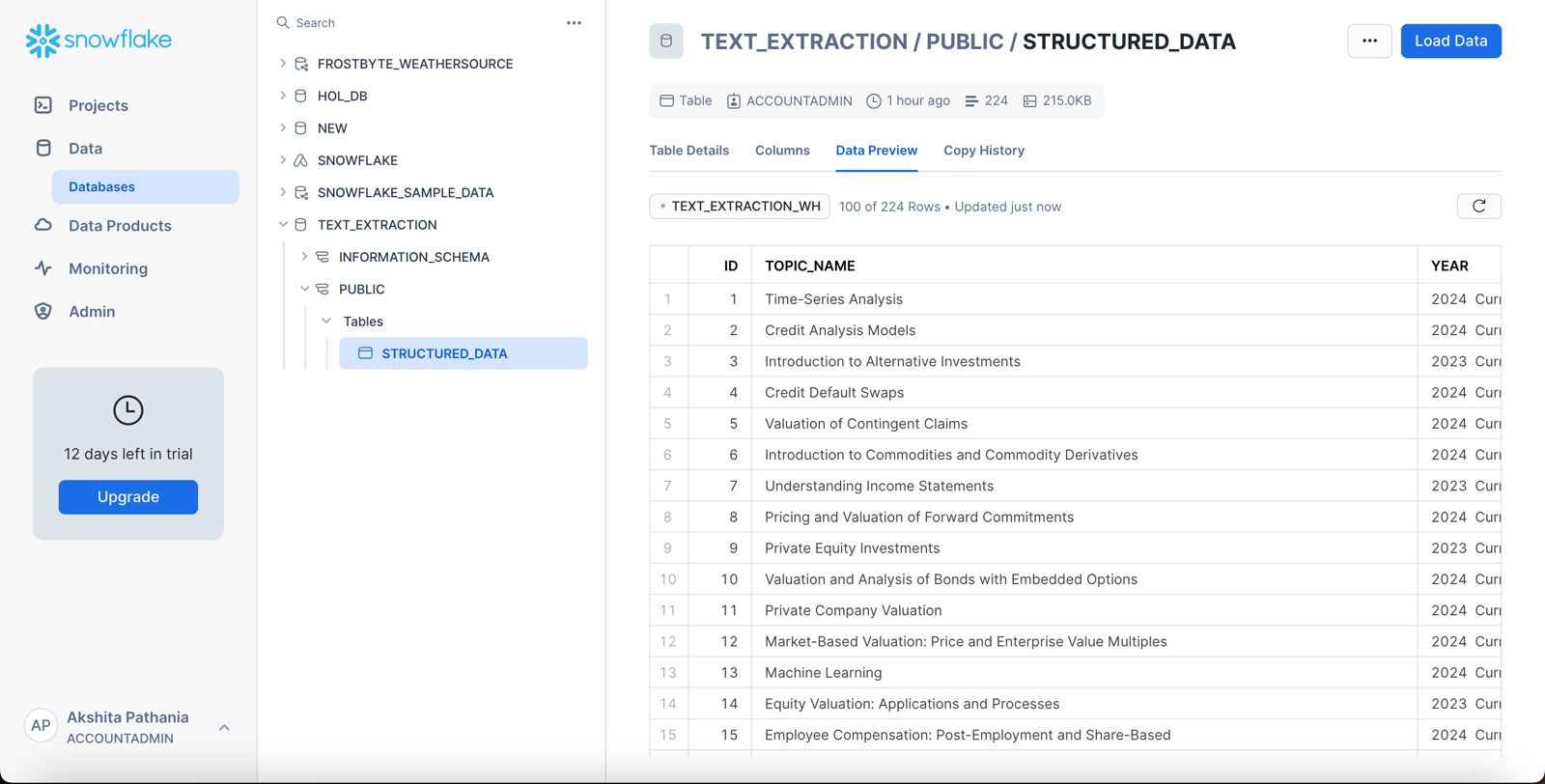


* Once the Data is inserted into the Snowflake from the CSV file extracted from the website. We can see the table being created along with the data inserted in it on the terminal and the Snowflake Web UI.





* Database populated with Structured CSV file



A notebook is available detailing this process.

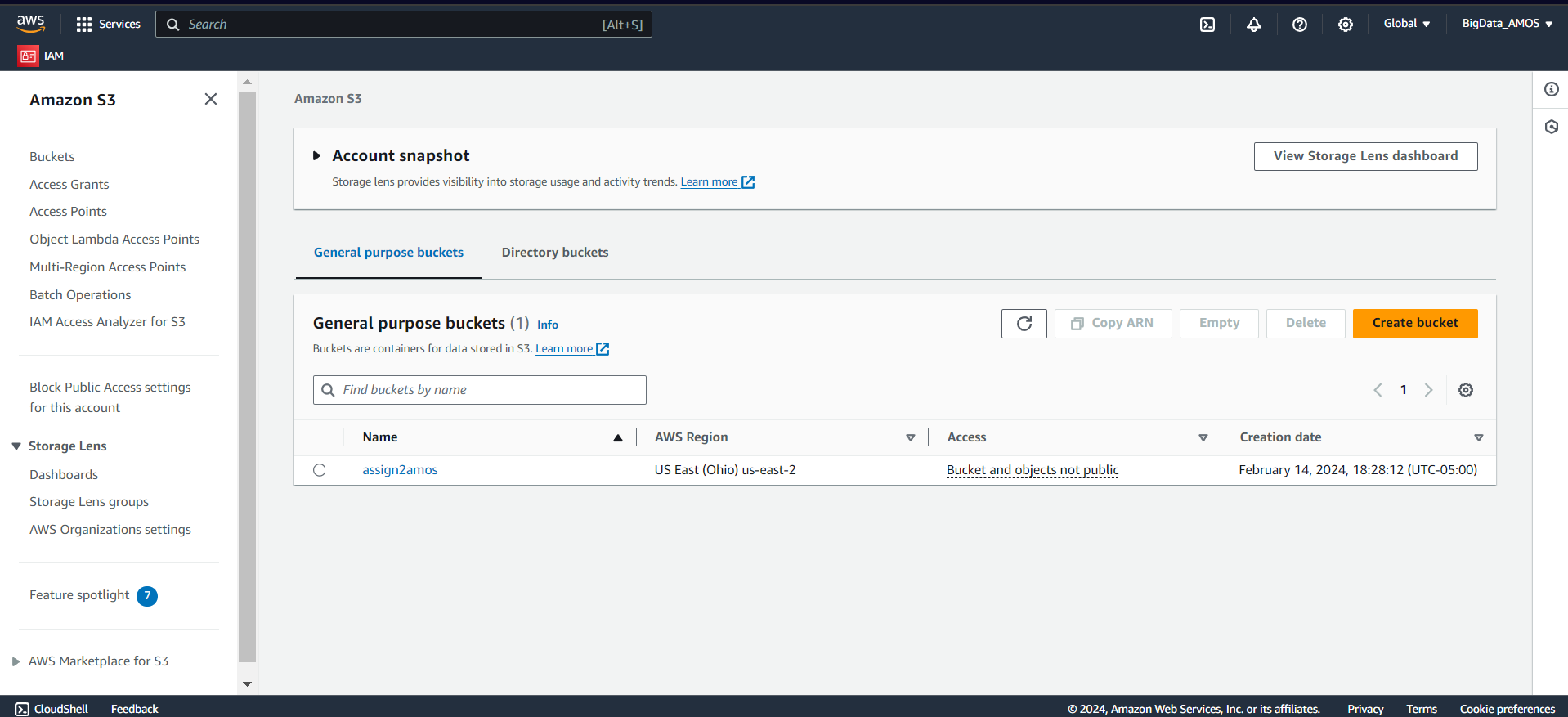
**V. Cloud Storage Integration**

### **Step 1: Sign Up for AWS**

1. **Create an AWS Account:** Go to AWS homepage ([https://aws.amazon.com/](https://colab.research.google.com/corgiredirector?site=https%3A%2F%2Faws.amazon.com%2F)) and sign up.
2. **Log in to the AWS Management Console:** Once the account is set up, log in to the AWS Management Console.

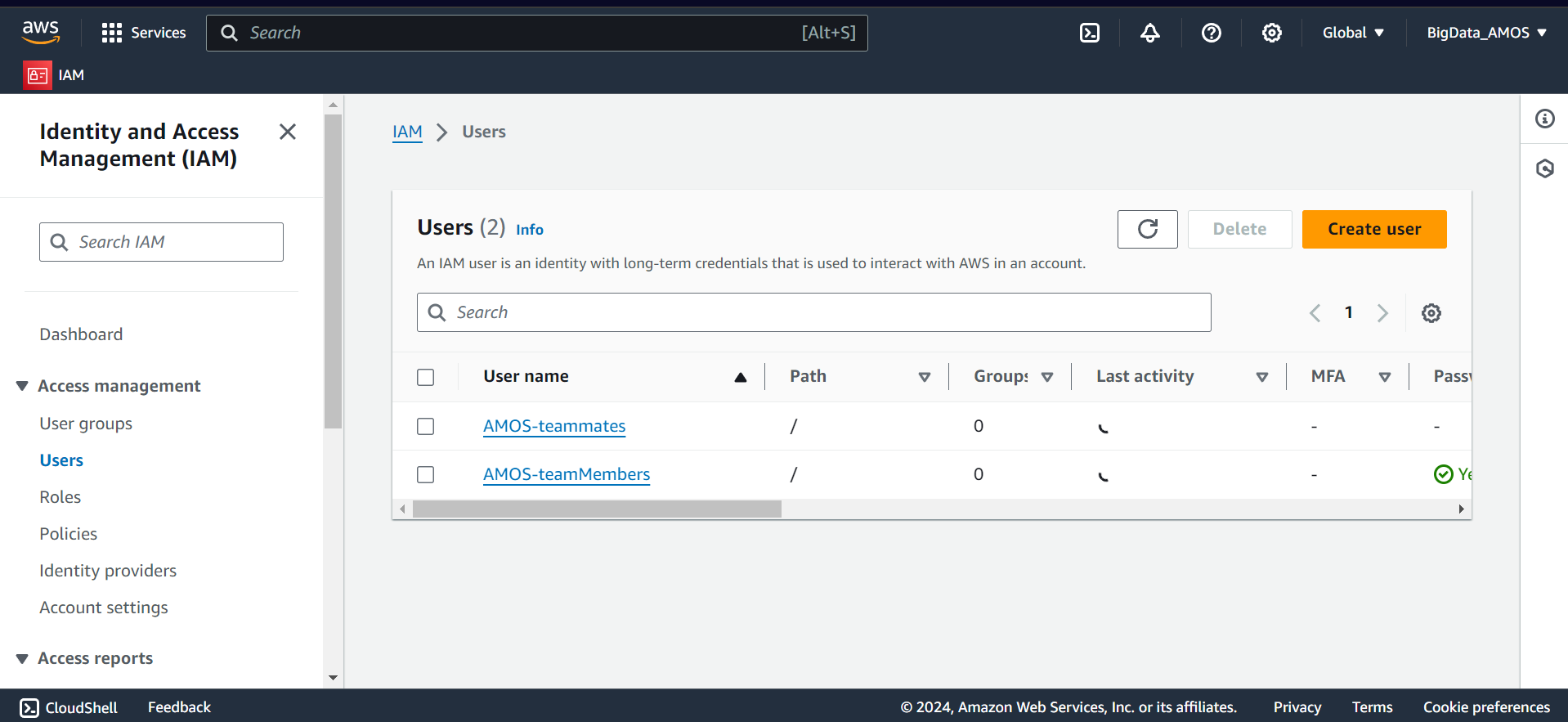
### **Step 2: Create an S3 Bucket**

1. **Go to S3 Service:** In the AWS Management Console, under the Service menu find **S3**
2. **Create a New Bucket:** Click **Create Bucket**. Complete the creation of Bucket by following the instructions on-screen
3. **Review and Create:** Review the settings and click **Create bucket**.
4. **Bucket Name:** assign2amos



### **Step 3: Create an IAM User and Assign Permissions**

1. **Go to the IAM Service:** In the AWS Management Console, under the Service menu, find **IAM**
2. **Create a New User:** In the navigation pane, click **Users**-->**Add user**. Provide a user name - **AMOS-teammates** and select Programmatic access for the AWS access type.
3. **Attach Policies for S3 Access:** On the permissions page, select **Attach existing policies directly**. Search and select the AmazonS3FullAccess policy, or select the customized policy created - ***Policy\_AMOS\_uploadData.***
4. **Review and Create:** Review the user details and permissions, then click on **Create user**.
5. **Download Credentials:** After the user is created, download the **Access Key ID** and **Secret Access Key**. Save these credentials securely.



### **Step 4: Configure AWS CLI with Your Credentials**

1. **Install AWS CLI:** Download and install the AWS Command Line Interface (CLI)
2. **Configure the CLI:**
   * Open a terminal or command prompt and run aws configure.
   * Enter the access key ID and secret access key when prompted. Also, specify the default region name and output format (e.g., us-west-2 and json).  
     aws configure  
     AWS Access Key ID [None]: YOUR\_ACCESS\_KEY\_ID  
     AWS Secret Access Key [None]: YOUR\_SECRET\_ACCESS\_KEY  
     Default region name [None]: YOUR\_PREFERRED\_REGION  
     Default output format [None]: json
3. **Verify Configuration:** Verify the configuration, run aws s3 ls which will list the S3 buckets in your AWS

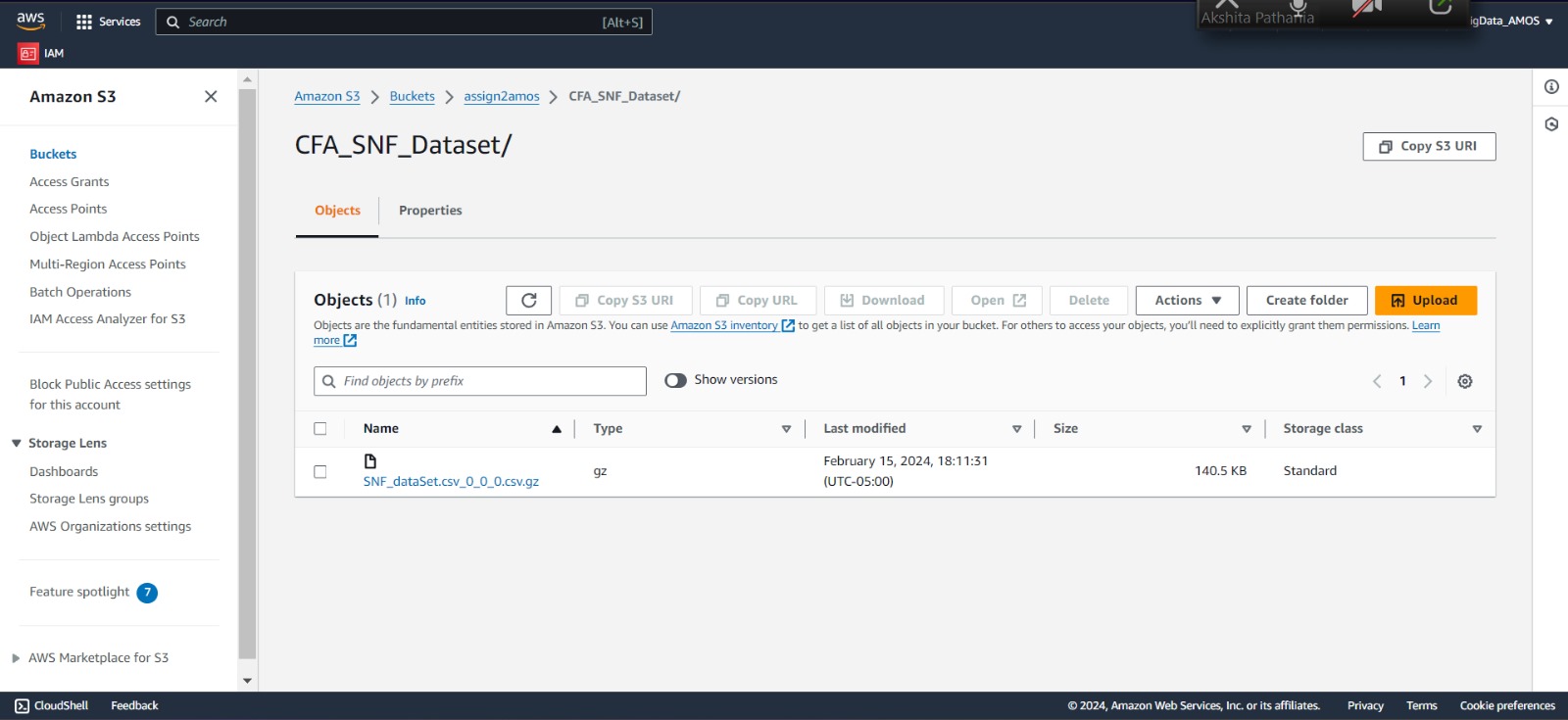
### **Step 5: Run Python Script**

After completion of the above steps, the AWS environment is ready. Now run the Python script below to upload files to your S3 bucket.  
  
Now,

* Uploading CSV FILE from Snowflake to AWS S3 Bucket

#Code to upload the CSV file from SNF to the AWS S3 bucket

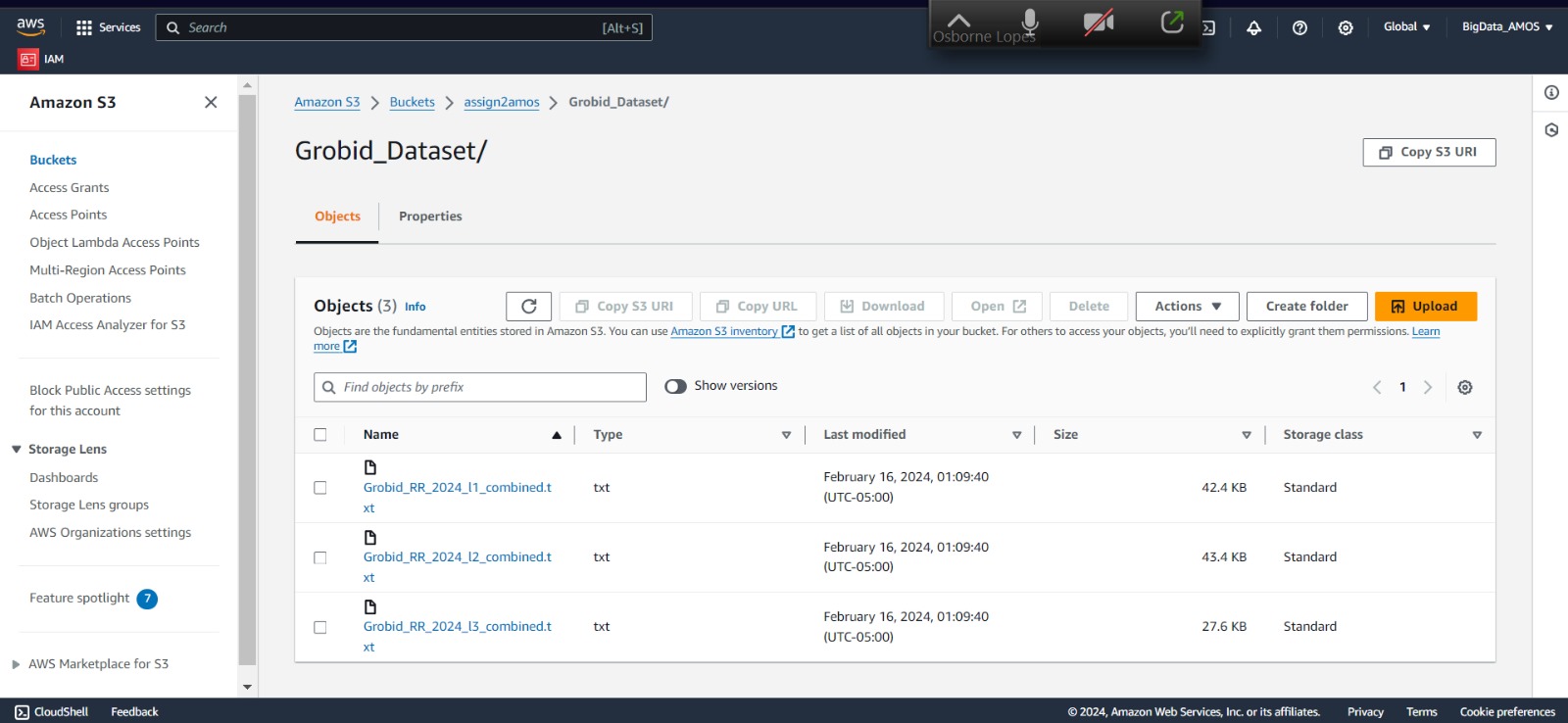




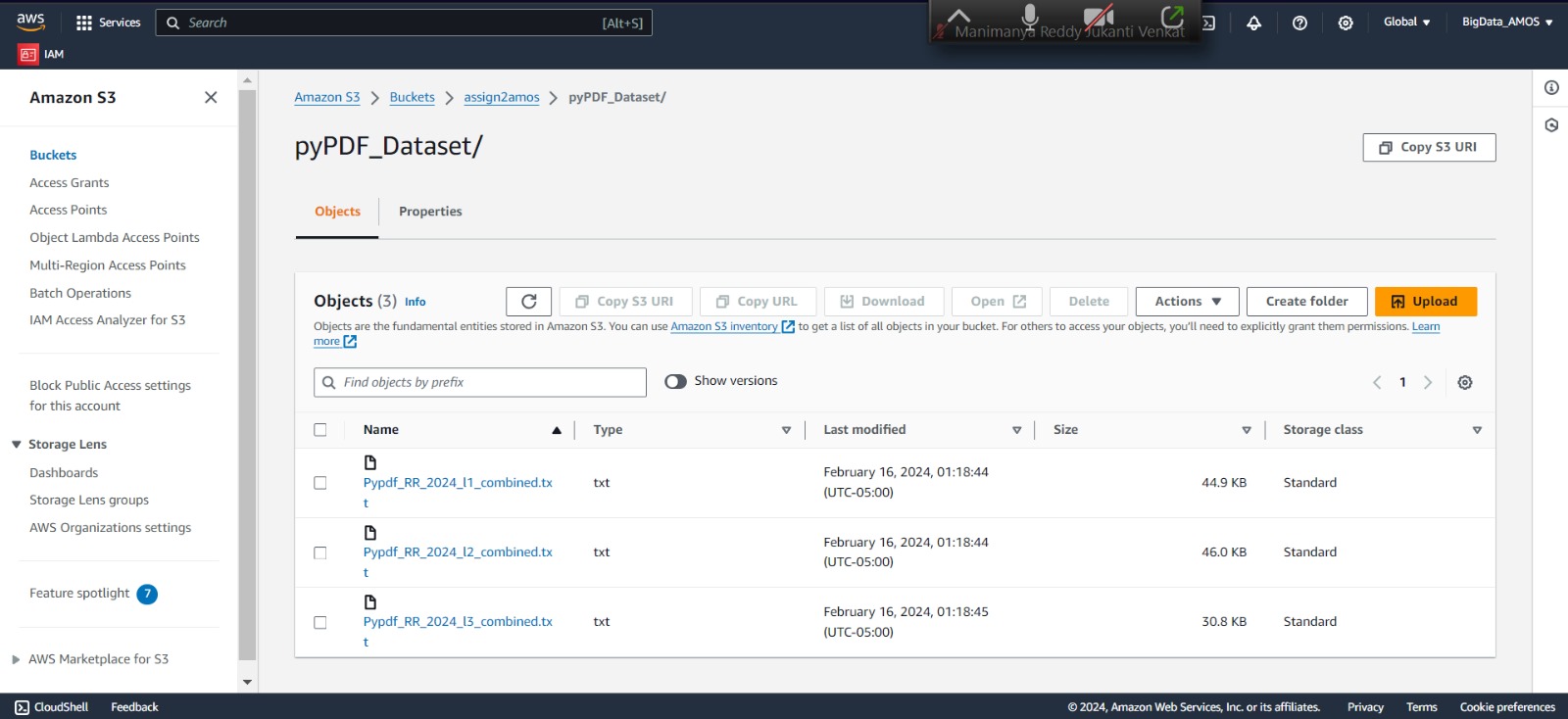
* **Uploading extracted text files from Grobid to AWS S3 Bucket**

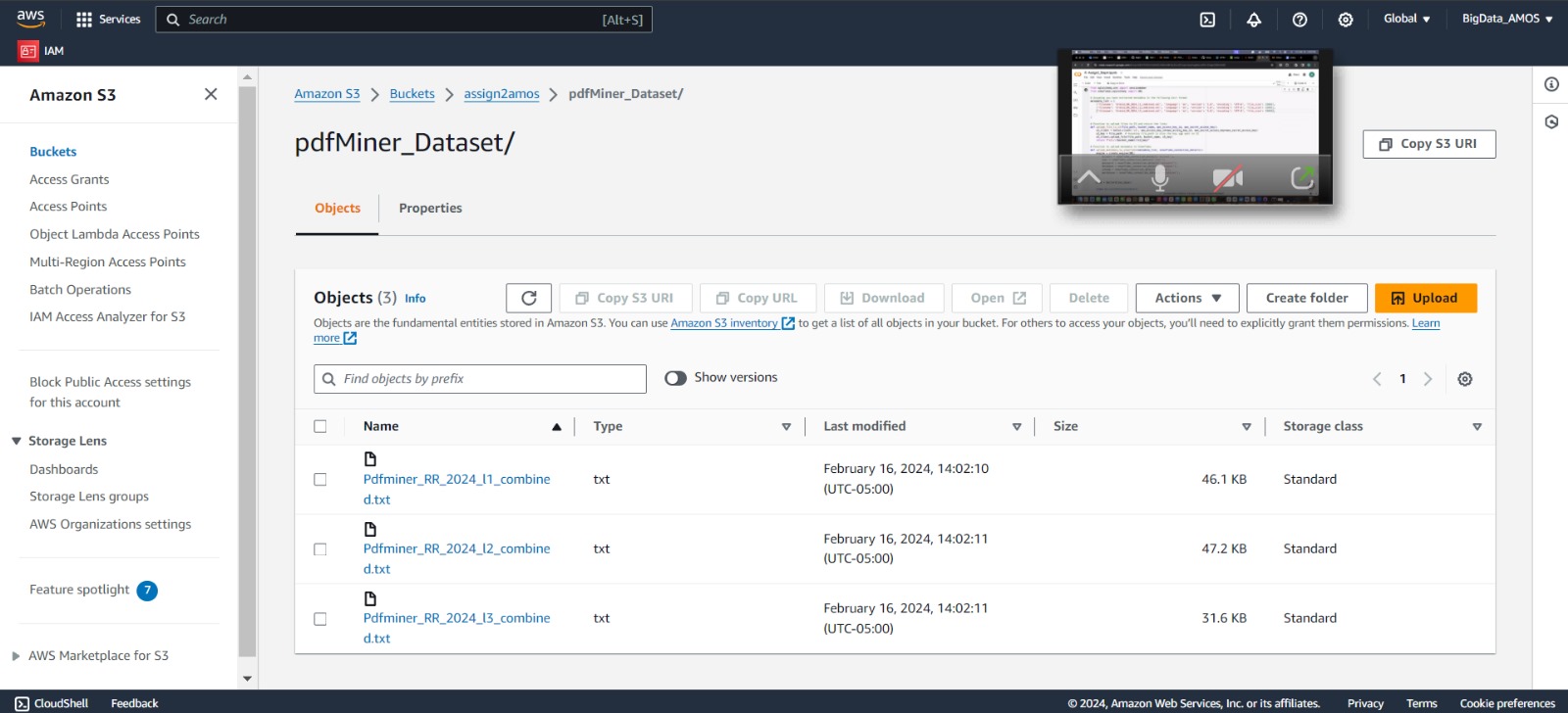
#Code to upload the extracted text file from Grobid to the AWS S3 bucket





#### **Uploading extracted text files from PyPDF to AWS S3 Bucket**

#Code to upload the extracted text file from PYPDF to the AWS S3 bucket



#### 

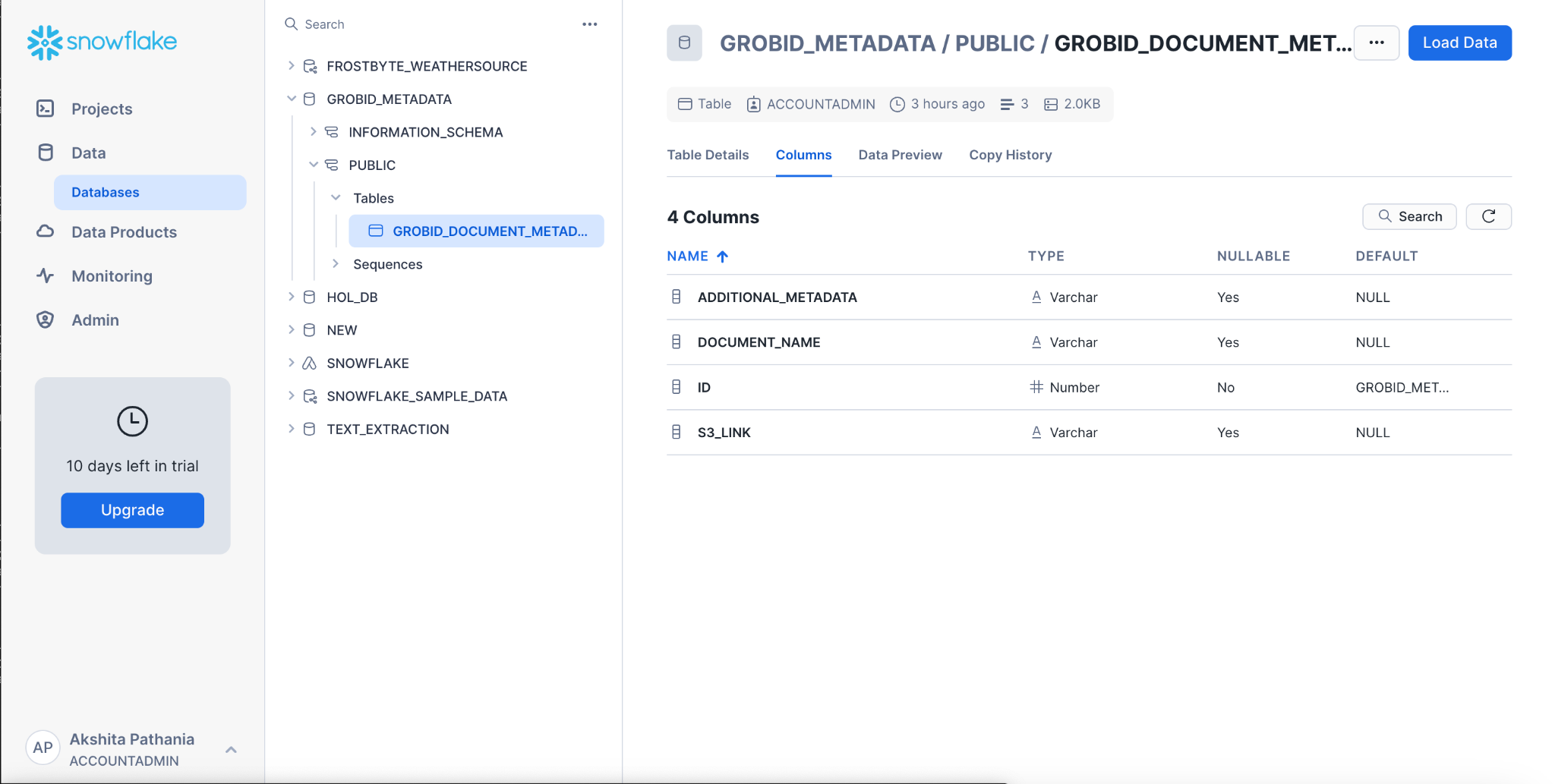
#### **Overall S3 population**

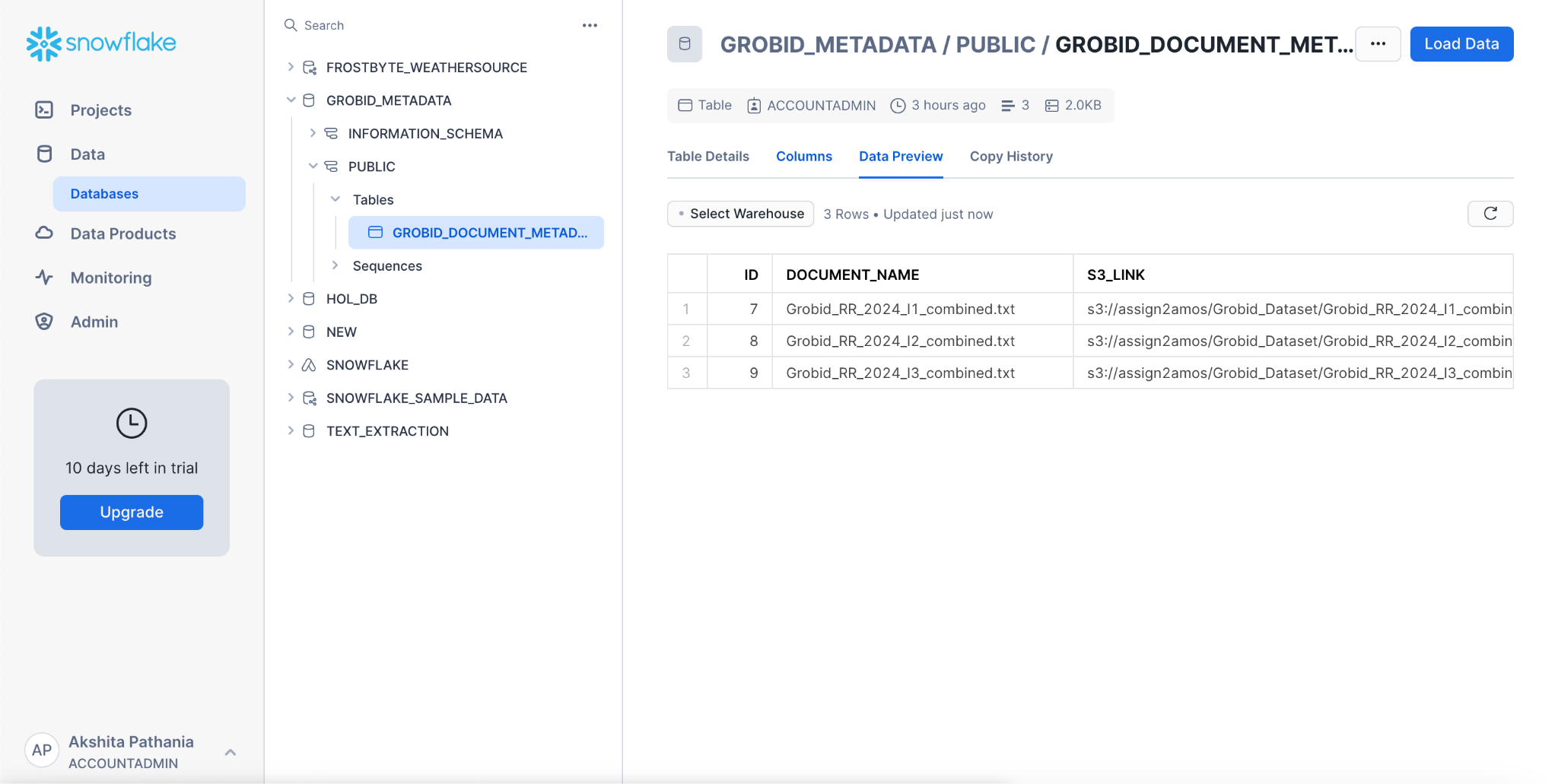


#### - Uploading files from AWS S3 Bucket to Snowflake -

#### 

* The following screenshot shows the Grobid Metadata uploaded on Snowflake





The Metadata of the structured files from Grobid has now been uploaded onto Snowflake.