Remittance History Analysis using PySpark on EMR and Athena

Agenda

FinTech companies usually lend money to individual customers and businesses based on the borrower's eligibility. Criteria for determining the creditworthiness of borrowers for such loans can involve rigorous analysis of past remittances and transactions that had occurred to and from the borrower's accounts. Data preparation and staging from raw transactional logs is a vital aspect of the analysis before which an ML team of the company cannot proceed further with the modeling phase.

In this project, we will process and prepare raw transaction data by normalizing the amounts involved to a standardized currency using API data of past exchange rates.

Aim

To perform Spark Transformations on bank transactions using a real-time currency ticker API and loading the processed data to Athena using Glue Crawler.

Data Description

The data will be acquired and processed using two sources. The first source involves past currency rates from an API provided by 'Openexchangerates.' The second source is a static file of transaction data with the following fields:

- Account_ID
- Value_date
- Transaction_details
- Withdrawal_amount
- Deposit_amount
- Currency
- Balance_amount

Tech stack:

➔Language: Python

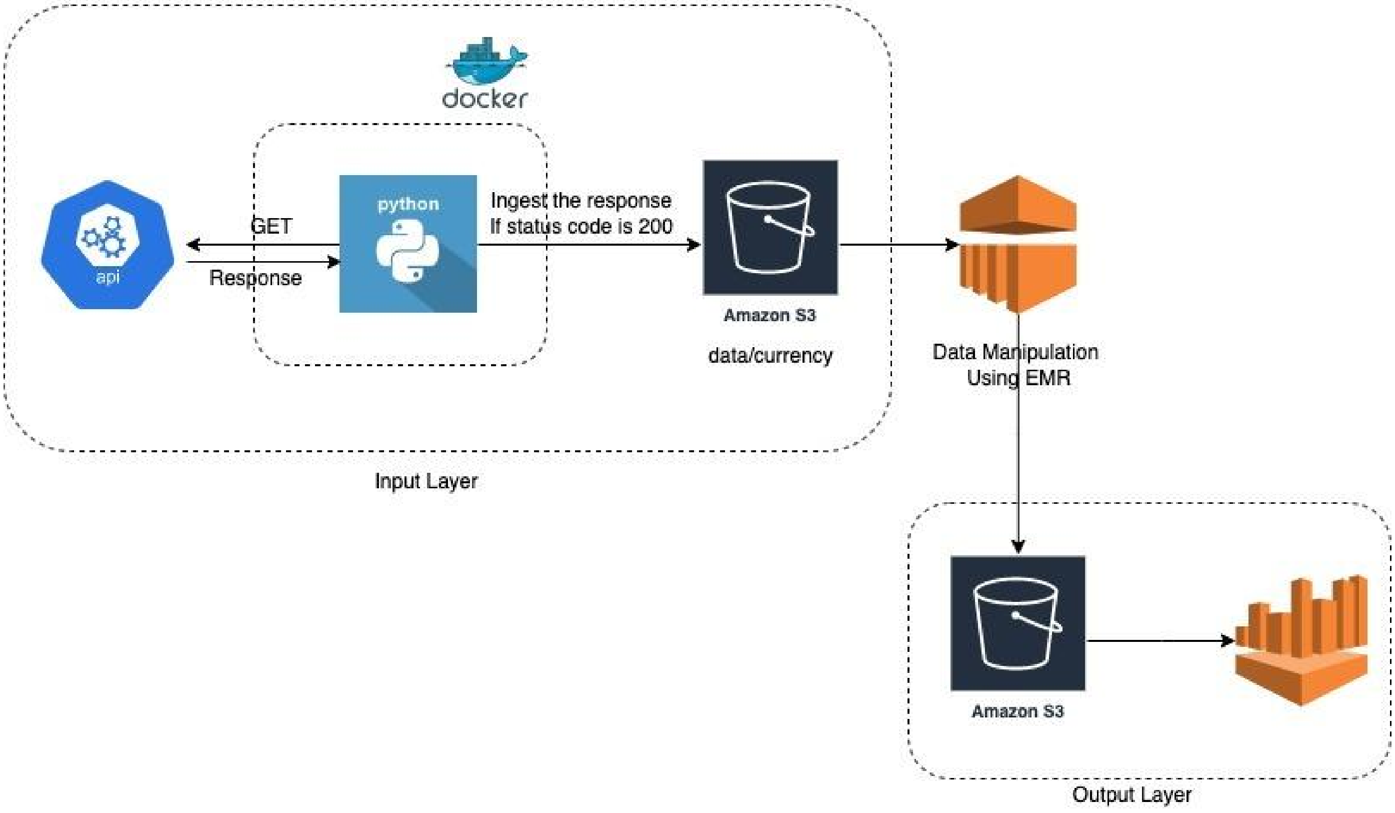
➔Package: PySpark

➔Services: Docker, Spark, AWS EMR, AWS S3, AWS Glue Crawler, AWS Athena, AWS EC2, Cron Job

Project Takeaways

- Understanding the Project Overview
- Creating an AWS EC2 instance
- Connecting to an AWS EC2 instance via SSH
- Introduction and Installation of Docker
- Visualizing the complete Data Pipeline
- Programmatically access S3 bucket
- Processing currency data from API
- Creating a cluster on AWS EMR
- Conversion of CSV data to Parquet format
- Using Pyspark on EMR for transformations
- Using crontab to schedule the code execution
- Using AWS Glue Crawler on S3 data
- Querying processed data using Athena

API Based Spark Data Pipeline


docker
api
GET
Response
python
Ingest the response
If status code is 200
Amazon S3
data/currency
Data Manipulation
Using EMR
Input Layer
Amazon S3
Output Layer