**STEPS**

**1. Fetch Data:**

* First, we **fetch stock data** using **yfinance** and **news data** using **NewsAPI**.
* The data is then **merged** into a single dataset, which is done in the **get\_data.py** script.

**2. Store the Data in S3:**

* The merged data, specifically **tesla\_balanced\_training\_data.csv**, is **uploaded to Amazon S3**. This file will be used as the input for training the machine learning model.

**3. Train the Model with SageMaker:**

* We use **AWS SageMaker** to **train** the model. The trained model, **tesla\_model\_fixed.pkl**, is then **saved in S3** for future use.

**4. Make Predictions:**

* In the **Prediction.ipynb** notebook, we use the trained model to **predict** whether the Tesla stock will go **up or down** based on the latest data.

**5. Launch Web Application:**

* The **streamlit.py** script is run in the **command prompt** to launch a **web application**.
* Users can enter the **open**, **close**, and **volume** values for today's stock data to get a prediction of tomorrow’s movement.

**6. Save Predictions to S3:**

* The **predictions** are automatically **saved in a CSV file** and **uploaded to S3**. This file is named **predictions\_log.csv**.

**7. Visualize in QuickSight:**

* The **predictions\_log.csv** file stored in S3 is used to **plot insights** and **graphs** in **AWS QuickSight**, helping visualize stock predictions over time.