Project 2: Requirements

Step 1: In an EC2 instance, download code from github

```
git clone https://github.com/aws-training/orderhistory.git
```

Step 2: Go in to the orderhistory directory on the EC2 instance and give execute permission to LogGenerator.py

```
$ cd orderhistory
$ chmod a+x LogGenerator.py
```

Step 3: Create a log directory to place the generated Log files

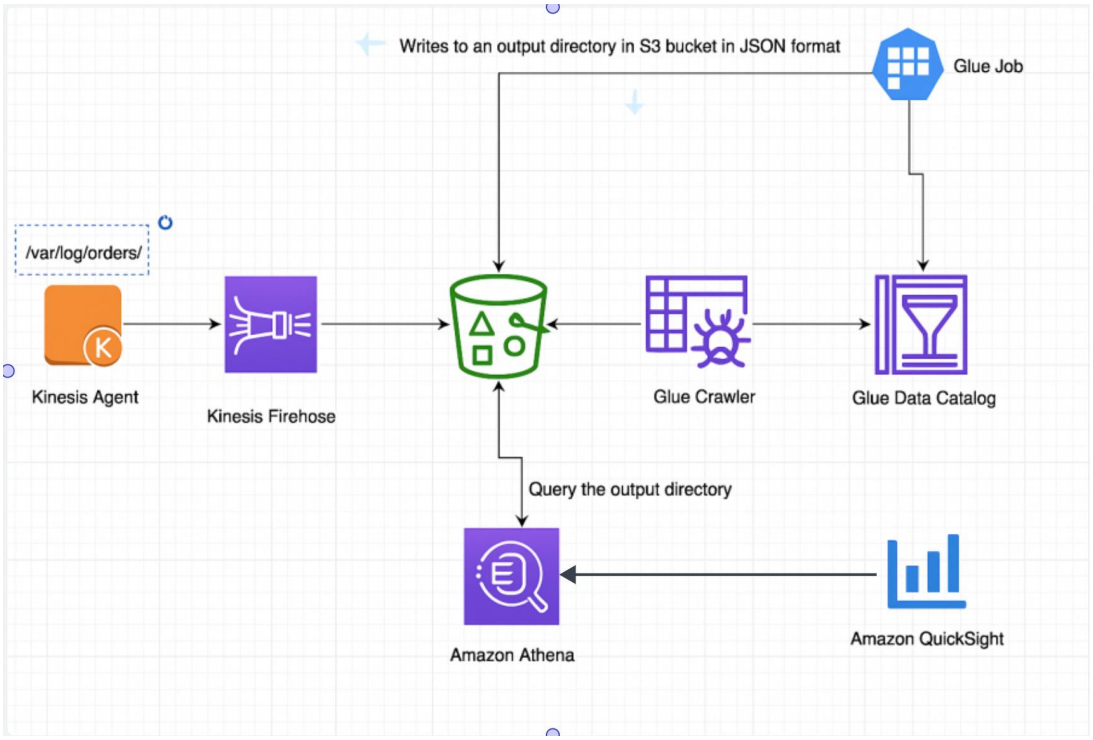
```
$ sudo mkdir /var/log/orders
```

Step 4: Modify the agent.json file and configure the properties.

Step 5: Run the LogGenerator and check the agent logs to public orders to Kinesis firehose.

```
$ sudo ./LogGenerator.py 500000
$ tail -f /var/log/aws-kinesis-agent/aws-kinesis-agent.log
```

Step 6: Follow the below diagram to create the data pipeline



Input Order history Schema (From Kinesis)

InvoiceNo	String
CustomerID	Integer
InvoiceDate	String
Quantity	float
Description	String
UnitPrice	float

Country	String
StockCode	String

Output Order History Schema (From Glue job to S3 output bucket)
Apply the following transformations to the data in the glue job

1. **Generate OrderID with Hashing or UUID for Uniqueness(include both Invoice no and stockcode)**
2. **Compute TotalAmount and Apply Currency Conversion**
3. **Parse and Split InvoiceDate into Date and Time Columns**
4. **Map Countries to Continents or Regions**
5. **Filter Out Invalid Records**
6. **Filter out rows with:**
   - **Negative quantities or unit prices**
   - **Missing CustomerID**
7. **Tag Orders as High if total order value is more than 100**
8. **Write Partitioned Output to S3**
   Partition by Country and Date:

Output of Glue job with the following fields

CustomerID	Integer
OrderID	String
InvoiceDate	String
Quantity	float
Description	String
UnitPrice	float
Country	String

and other generated columns

In Athena run the following queries and also visualize some of them in QuickSight

1. **List Top 10 High Value Orders**
2. **Average Order Value by Country**
3. **Order Trends Over Time(How many orders on each day and how was the revenue like?)**
4. **Region-wise Distribution of Orders**
5. **Peak Order Time (Hour-wise Aggregation)**
6. **Query a Specific Partition (e.g., UK orders on 2025-07-25)**