**Flink 项目介绍**

**Title:**

**Ecommerce real-time big data analytics (based on Apache Flink)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**One sentence description:**

Implemented Flink pipelines (using Flink's DataStream and Dataset APIs) for enabling (real-time) monitoring of key e-commerce metrics from streaming and batch processing data sources to provide immediate insights into user behavior and sales trends.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Bullet Points description:**

**One sentence concise version as the first bullet point.**

Utilized Flink's windowing and event-time processing capabilities to handle time-based aggregations and handle out-of-order data in the streaming pipeline.

**Option 1:**

Generated daily and weekly reports on website performance, revenue, and customer engagement, facilitating data-driven decision-making.

**Option 2:**

Leveraged Flink's stateful processing features, including KeyedState and ValueState, to maintain and update the application's state across time and data partitions.

**Prepare 30s; 1min; 3min version**

User behavior Analysis: Top N popular item

Product Click Count,

Pageview

Learning: Table API, SQL Query, Word Count/ Word Frequency,

Watermark, Windowing, Stream Processing, Operator, MapReduce

**Difficulty:**

**No Stateful Computation, make it impossible to handle real time data in real time**

**Data validation techniques:** Identified and resolved data quality issues by implementing data validation and cleansing techniques in the processing pipeline.

**STAR: Situation, Task, Action, Result**

**Map – Reduce vs Process Function**

**Situation, Task**

**Raw data**

**Bullet Points:**

implemented Flink pipelines using Flink's DataStream and DataSet APIs for real-time streaming and batch processing of e-commerce data.

Utilized Flink's windowing and event-time processing capabilities to handle time-based aggregations and handle out-of-order data in the streaming pipeline.

Implemented custom Flink functions, such as map, flatMap, and reduce, to transform and manipulate data during the processing stages.

Leveraged Flink's stateful processing features, including KeyedState and ValueState, to maintain and update the application's state across time and data partitions.

Integrated Flink with external systems, such as Apache Kafka, for data ingestion and Apache Hadoop/Hive for storing and managing large volumes of data for batch processing.

Utilized Flink's fault-tolerance mechanisms, such as checkpointing and state snapshots, to ensure reliable and consistent processing in the face of failures.

Optimized Flink jobs by tuning parallelism, buffer sizes, and resource allocation to maximize throughput and minimize latency.

Enabled real-time monitoring of key e-commerce metrics, providing immediate insights into user behavior and sales trends.

Generated daily and weekly reports on website performance, revenue, and customer engagement, facilitating data-driven decision-making.

Identified and resolved data quality issues by implementing data validation and cleansing techniques in the processing pipeline.

Reading:

Flink data source upstream: [Data Pipelines & ETL | Apache Flink](https://nightlies.apache.org/flink/flink-docs-master/docs/learn-flink/etl/)

[Apache Flink: Stream Processing for All Real-Time Use Cases (confluent.io)](https://www.confluent.io/blog/apache-flink-stream-processing-use-cases-with-examples/)

Flink data source downstream: [Building real-time dashboard applications with Apache Flink, Elasticsearch, and Kibana | Elastic Blog](https://www.elastic.co/blog/building-real-time-dashboard-applications-with-apache-flink-elasticsearch-and-kibana)  
