《大数据综合项目实训》练习01

一、背景

某电商平台产生用户行为日志（浏览/收藏/加购/下单），需要完成一次完整流程：
日志采集 → Kafka 缓冲 → Spark on YARN 处理 → 写入 Hive → 统计分析 → 保存汇总表

日志文件：用户行为日志`test01.log`

数据格式如下：

用户ID	商品ID	行为类型	行为时间	商品类目
U003	P2001	浏览	2025-01-06 09:12:08	图书
U001	P2003	收藏	2025-01-06 09:15:42	服装
U005	P2002	浏览	2025-01-06 09:18:30	食品
U002	P2004	加购	2025-01-06 09:21:55	美妆
U004	P2006	浏览	2025-01-06 09:25:10	运动用品
U003	P2002	加购	2025-01-06 09:28:46	食品
U006	P2005	浏览	2025-01-06 09:32:19	办公用品
U001	P2002	下单	2025-01-06 09:35:04	食品
U002	P2004	下单	2025-01-06 09:38:27	美妆
U007	P2003	浏览	2025-01-06 09:42:50	服装

二、内容

第 1 题：启动与检查环境

1.1 启动 HDFS

在 master 执行：

```
start-dfs.sh
```

1.2 启动 YARN

在 slave1 执行：

```
start-yarn.sh
```

1.3 启动 Zookeeper

在三台节点：

```
zkServer.sh start
```

1.4 启动 Kafka

在三台节点：

```
kafka-server-start.sh -daemon /opt/apps/kafka/config/server.properties
```

第 2 题：Kafka 创建 Topic 并验证

2.1 创建 Topic（10 分）

在 slave1 节点创建 Kafka 主题，主题名称: `test_topic_01`

```
kafka-topics.sh \
--create \
--topic test_topic_01 \
--bootstrap-server master:9092 \
--partitions 3 \
--replication-factor 1
```

2.2 验证 Topic

```
kafka-topics.sh --list --bootstrap-server master:9092
```

2.3 在 slave1 节点启动Kafka消费者

```
kafka-console-consumer.sh \
--bootstrap-server master:9092,slave1:9092,slave2:9092 \
--topic test_topic_01 \
--from-beginning
```

第 3 题：Kafka生产者发送消息到主题

3.1 准备日志文件

把日志文件`test01.log`复制到`slave2`指定目录`/opt/data/flume/logs`：

```
# 在slave2创建日志存放目录
mkdir -p /opt/data/flume/logs
# 把日志文件`test01.log`复制到刚才创建的目录下
```

3.2 在slave2将用户数据发送到 Kafka 主题

```
# 在slave2下操作
cat /opt/data/flume/logs/test01.log | \
kafka-console-producer.sh \
--broker-list master:9092,slave1:9092,slave2:9092 \
--topic test_topic_01
```

3.3 在slave1节点查看 Kafka 消费主题结果

第 4 题：Hive 建表

创建`test_table_01`用户行为日志表

表含义说明： 用户行为日志表用于记录用户在电商平台上的行为操作信息，包括浏览、收藏、加购和下单等行为。

字段说明

中文字段	字段名	数据类型	说明
用户ID	user_id	STRING	用户唯一标识
商品ID	product_id	STRING	商品唯一标识
行为类型	action_type	STRING	浏览 / 收藏 / 加购 / 下单
行为时间	action_time	STRING	行为发生时间
商品类目	category	STRING	商品所属分类

完成下面的Hive任务

4.1 在 master节点启动Hive环境：

```
# 启动元数据服务
hive --service metastore &
# 启动Hive
hive
```

4.2 选择数据库

```
use default;
```

4.3 删除数据表

```
DROP TABLE IF EXISTS test_table_01;
```

4.4 创建 用户行为数据表：`test_table_01`

```
CREATE TABLE IF NOT EXISTS test_table_01 (
  user_id     STRING,
  product_id  STRING,
  action_type STRING,
  action_time STRING,
  category    STRING
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE;
```

4.5 查看数据表

```
show tables;
```

第 5 题：在 spark-shell 中完成 Kafka → Spark → Hive 数据处理任务

在 FinalShell 新建一个master节点窗口，启动 spark-shell：

```
spark-shell \
--master yarn \
--deploy-mode client \
--driver-memory 512m \
--conf spark.driver.maxResultSize=256m \
--conf spark.executor.instances=1 \
--conf spark.executor.cores=1 \
--conf spark.executor.memory=512m \
--conf spark.sql.shuffle.partitions=2 \
--conf spark.sql.catalogImplementation=hive \
--conf spark.driver.extraClassPath=/opt/apps/hive/conf \
--conf spark.executor.extraClassPath=/opt/apps/hive/conf \
--packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.1.1
```

在 spark-shell 完成下面Spark任务：

5.1 从 Kafka 读取主题数据

```
val kafkaDF = (
    spark.read
  .format("kafka")
  .option("kafka.bootstrap.servers", "master:9092")
  .option("subscribe", "test_topic_01")
  .load()
)
```

5.2 提取 Kafka 中的 value 字段

```
val lines = kafkaDF.selectExpr("CAST(value AS STRING) AS line")
```

5.3 解析用户 CSV 数据

```
val dataDF = lines.selectExpr(
  "split(line, ',')[0] AS user_id",
  "split(line, ',')[1] AS product_id",
  "split(line, ',')[2] AS action_type",
  "split(line, ',')[3] AS action_time",
  "split(line, ',')[4] AS category"
)
```

5.4 切换数据库并查看表

```
spark.sql("USE default")
spark.sql("show tables").show(false)
```

5.5 将 DataFrame 写入 Hive 表

```
dataDF.coalesce(1).write.mode("append").insertInto("test_table_01")
```

5.6 查询 Hive 表前 5 行数据，验证写入结果

```
spark.sql("SELECT * FROM test_table_01 LIMIT 5").show(false)
```