《大数据综合项目实训》练习05

一、业务背景

某电商平台需要对**用户基础信息数据**进行集中管理与分析，需要完成一次完整流程：
用户数据采集 → Kafka 缓冲 → Spark on YARN 处理 → 写入 Hive → 查询验证

日志文件：用户信息数据文件 `test05.log`

数据格式如下：

用户ID	用户姓名	性别	年龄	注册时间
U5001	李强	男	24	2024-11-28 08:45:10
U5002	王芳	女	20	2024-11-29 10:12:35
U5003	张伟	男	22	2024-11-30 11:38:50
U5004	刘洋	男	21	2024-12-01 14:06:20
U5005	陈敏	女	23	2024-12-02 15:22:40
U5006	赵磊	男	25	2024-12-03 09:18:05
U5007	周婷	女	22	2024-12-04 10:55:30
U5008	黄俊	男	21	2024-12-05 11:47:15
U5009	徐静	女	24	2024-12-06 16:10:45
U5010	杨帆	男	23	2024-12-07 17:32:00

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

在三台节点执行：

```
zkServer.sh start
```

1.4 启动 Kafka

在三台节点执行：

```
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

在 slave1 节点创建 Kafka 主题，主题名称：`test_topic_05`

```
kafka-topics.sh \
--create \
--topic test_topic_05 \
--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_05 \
--from-beginning
```

第 3 题：Kafka 生产者发送用户数据

3.1 准备用户数据文件

把日志文件`test05.log`复制到`slave2`指定目录`/opt/data/flume/logs`：

```
mkdir -p /opt/data/flume/logs
```

3.2 在slave2将用户数据发送到 Kafka 主题

```
cat /opt/data/flume/logs/test05.log | \
kafka-console-producer.sh \
--broker-list master:9092,slave1:9092,slave2:9092 \
--topic test_topic_05
```

3.3 在slave1节点查看 Kafka 消费主题结果（8 分）

第 4 题：Hive 建表（共 15 分）

任务：创建 `test_table_05` 用户信息表

表含义说明：
用户信息表用于记录电商平台中用户的基础资料信息，包括用户姓名、性别、年龄及注册时间等内容，为后续用户画像与数据分析提供基础数据支持。

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
用户姓名	user_name	STRING	用户姓名
性别	gender	STRING	男 / 女
年龄	age	INT	用户年龄
注册时间	register_time	STRING	用户注册时间

完成下面的 Hive 任务

4.1 在 master 节点 启动 Hive 环境：

```
hive --service metastore &
hive
```

4.2 选择数据库

```
use default;
```

4.3 删除数据表

```
DROP TABLE IF EXISTS test_table_05;
```

4.4 创建用户信息表：`test_table_05`

```
CREATE TABLE IF NOT EXISTS test_table_05 (
  user_id     STRING,
  user_name   STRING,
  gender      STRING,
  age         INT,
  register_time 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 读取用户数据

```
val kafkaDF = (
    spark.read
  .format("kafka")
  .option("kafka.bootstrap.servers", "master:9092")
  .option("subscribe", "test_topic_05")
  .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（5 列）

```
val dataDF = lines.selectExpr(
  "split(line, ',')[0] AS user_id",
  "split(line, ',')[1] AS user_name",
  "split(line, ',')[2] AS gender",
  "cast(split(line, ',')[3] AS int) AS age",
  "split(line, ',')[4] AS register_time"
)
```

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_05")
```

5.6 查询 Hive 表前 5 行数据，验证写入结果

```
spark.sql("SELECT * FROM test_table_05 LIMIT 5").show(false)
```