



## I. Tóm tắt bài thực hành

### 1. Yêu cầu lý thuyết

Sinh viên đã được trang bị kiến thức:

- Cấu trúc hệ thống phân tán và framework lập trình Apache Spark
- Đối tượng RDD (Resilient Distributed Dataset) trong Apache Spark
- Lập trình Python với Apache Spark thông qua PySpark

...

### 2. Nội dung

#### ❖ Ôn tập lại những kiến thức cần thiết

- Lập trình Python với Apache Spark thông qua PySpark

#### ❖ Làm quen với DataFrame

#### ❖ Thao tác với DataFrame bằng Spark SQL

### 3. Kết quả cần đạt

- ✓ Hiểu rõ về đối tượng DataFrame trong Apache Spark.
- ✓ Sử dụng được Spark SQL để thao tác với các đối tượng DataFrame.

## II. Ôn tập lại những kiến thức đã học

Sinh viên tham khảo tài liệu ở buổi học trước về việc sử dụng PySpark để lập trình trong Apache Spark.

## III. Yêu cầu bài làm sinh viên

Nội dung thực hành buổi 04 được thực hiện theo từng cá nhân. Sinh viên upload một tập tin <MSSV>.doc hoặc <MSSV>.docx, nội dung trả lời các bài tập bên dưới.

**Lưu ý: Bài nộp không theo đúng quy định này sẽ không được tính.**

## IV. Làm quen với Spark DataFrame

### Bài tập 1: Tạo và thao tác với DataFrame từ JSON

a. Tạo DataFrame từ nội dung của file JSON với câu lệnh sau:

```
df = spark.read.json("examples/src/main/resources/people.json")
```

b. Hiển thị nội dung của DataFrame ra stdout

```
df.show()
```

c. In ra stdout cấu trúc của DataFrame theo dạng cây

```
df.printSchema()
```

d. Chọn duy nhất một cột “name” để hiển thị

```
df.select("name").show()
```

e. Hiển thị toàn bộ các cột với tất cả giá trị của cột “age” tăng thêm 1

```
df.select(df['name'], df['age'] + 1).show()
```

f. Lọc ra những người có tuổi lớn hơn 21

```
df.filter(df['age'] > 21).show()
```

g. Đếm số người theo từng độ tuổi

```
df.groupBy("age").count().show()
```

### **Bài tập 2: Tạo và thao tác với DataFrame từ file CSV**

Tập dữ liệu được sử dụng là từ các cuộc đấu giá trực tuyến của eBay. Bộ dữ liệu đấu giá trực tuyến của eBay chứa các trường sau:

- ✓ **auctionid** - số nhận dạng duy nhất của một cuộc đấu giá
- ✓ **bid** (giá đấu) - giá đấu được đặt bởi người đấu giá
- ✓ **bidtime** (thời gian dự đấu giá) - thời gian (tính theo ngày) mà giá đấu đã được đặt, ngay từ đầu phiên đấu giá
- ✓ **bidder** (người đặt giá đấu) - tên người dùng eBay của người đấu giá
- ✓ **bidderrate** - Đánh giá phản hồi của eBay đối với người đấu giá
- ✓ **openbid** - giá mở cửa của người bán
- ✓ **price** - giá đóng cửa của mặt hàng đã bán (trương đương với giá cao nhất thứ hai + mức tăng)

a. Tạo DataFrame từ nội dung của file CSV với câu lệnh sau:

```
df = spark.read.format("csv").option("header",
    "true").load("path/to/ebay.csv")
```

b. Hiển thị nội dung của DataFrame ra stdout

```
df.show()
```

c. In ra stdout cấu trúc của DataFrame theo dạng cây

```
df.printSchema()
```

d. Đếm số cuộc đấu giá đã được tổ chức bằng câu lệnh

```
df.select("auctionid").distinct().count()
```

e. Đếm số lượng giá đấu trên mỗi mặt hàng

```
df.groupBy("auctionid", "item").count().show()
```

f. Lọc và hiển thị tất cả những phiên đấu giá có giá đóng cửa lớn hơn 100

```
highprice = df.filter("price > 100")
highprice.show()
```

## **V. Thao tác với DataFrame bằng ngôn ngữ SQL**

### **Bài tập 3: Thao tác với DataFrame bằng ngôn ngữ SQL**

Với dữ liệu people ở Bài tập 1, sinh viên thực hiện các yêu cầu sau:

a. Tạo một khung nhìn tạm (SQL temporary view) từ DataFrame df

```
df.createOrReplaceTempView("people")
```

b. Truy vấn toàn bộ dữ liệu và hiển thị kết quả

```
sqlDF = spark.sql("SELECT * FROM people")
sqlDF.show()
```

c. Thực hiện lại toàn bộ các yêu cầu d, e, f, g trong Bài tập 1 bằng truy vấn SQL trong DataFrame.

### **Bài tập 4: Thao tác với DataFrame bằng ngôn ngữ SQL**

Với dữ liệu people ở Bài tập 2, sinh viên thực hiện các yêu cầu sau:

a. Tạo một khung nhìn tạm (SQL temporary view) từ DataFrame df

```
df.createOrReplaceTempView("ebay")
```

b. Truy vấn toàn bộ dữ liệu và hiển thị kết quả

```
sqlDF = spark.sql("SELECT * FROM ebay")  
sqlDF.show()
```

c. Thực hiện lại toàn bộ các yêu cầu d, e, f trong Bài tập 2 bằng truy vấn SQL trong DataFrame.

~ HẾT ~

