In [1]:
```
from pyspark.sql import SparkSession
spark=SparkSession.builder.appName("DfApp").getOrCreate()
```

In [4]:
```
spark
```

Out[4]:

**SparkSession - in-memory**

**SparkContext**

Spark UI

| | |
|---|---|
| **Version** | `v3.1.2` |
| **Master** | `local[*]` |
| **AppName** | `DfApp` |

In [10]:
```
pdf=spark.read.option("header","true").csv('D:\items.csv',inferSchema=True)
pdf.show()
```

```
+-------+---------+---------+--------+-----+----+
|Item Id|Item Name|Item Cost|Supplier|Grade| Qty|
+-------+---------+---------+--------+-----+----+
|      4|    Chock|    65.76|       X|    A|  34|
|      5|   Pencil|    45.65|       Y|    B|null|
|      6|      Pen|    76.87|       X|    A|  23|
|      7|   Duster|    54.0|        Y|    C|  25|
|      8|     null|     null|       Z| null|  34|
|      9|     book|     53.0|    null| null|  65|
|      7|   Duster|    54.0|        Y|    C|  25|
|      8|     null|     null|       Z| null|  34|
+-------+---------+---------+--------+-----+----+
```

In [25]:
```
from pyspark.sql.functions import col
```

In [29]:
```
pdf1=pdf.select(col('Item Id').alias('ItemId'),
                col('Item Name').alias('ItemName'),
                col('Item Cost').alias('ItemCost'))
```

In [30]:
```
pdf1.show()
```

```
+------+---------+--------+
|ItemId|ItemName|ItemCost|
+------+---------+--------+
|     4|    Chock|   65.76|
|     5|   Pencil|   45.65|
|     6|      Pen|   76.87|
|     7|   Duster|    54.0|
|     8|     null|    null|
|     9|     book|    53.0|
|     7|   Duster|    54.0|
|     8|     null|    null|
+------+---------+--------+
```

In [31]:
```
pdf1.printSchema()
```

```
root
 |-- ItemId: integer (nullable = true)
 |-- ItemName: string (nullable = true)
 |-- ItemCost: double (nullable = true)
```

In [32]:
```
pdf.head(2)
```

Out[32]:
```
[Row(Item Id=4, Item Name='Chock', Item Cost=65.76, Supplier='X', Grade='A', Qty=34),
 Row(Item Id=5, Item Name='Pencil', Item Cost=45.65, Supplier='Y', Grade='B', Qty=None)]
```

In [33]:
```
pdf.limit(3).toPandas()
```

Out[33]:

| | Item Id | Item Name | Item Cost | Supplier | Grade | Qty |
|---|---|---|---|---|---|---|
| **0** | 4 | Chock | 65.76 | X | A | 34.0 |
| **1** | 5 | Pencil | 45.65 | Y | B | NaN |
| **2** | 6 | Pen | 76.87 | X | A | 23.0 |

In [34]:
```
pdf.printSchema()
```

```
root
 |-- Item Id: integer (nullable = true)
 |-- Item Name: string (nullable = true)
 |-- Item Cost: double (nullable = true)
 |-- Supplier: string (nullable = true)
 |-- Grade: string (nullable = true)
 |-- Qty: integer (nullable = true)
```

In [35]:
```
# Temporary view
pdf1.createOrReplaceTempView("tempview")
```

In [36]:
```
result=spark.sql("select * from tempview").limit(3).toPandas()
```

In [37]:
```
result
```

Out[37]:

| | ItemId | ItemName | ItemCost |
|---|---|---|---|
| **0** | 4 | Chock | 65.76 |
| **1** | 5 | Pencil | 45.65 |
| **2** | 6 | Pen | 76.87 |

In [41]:
```
res=spark.sql("select * from tempview where ItemName == 'Pen'")
res.show()
```

```
+------+--------+--------+
|ItemId|ItemName|ItemCost|
+------+--------+--------+
|     6|     Pen|   76.87|
+------+--------+--------+
```

In [43]:
```
pdf=spark.read.option("header","true").csv('D:\items.csv',inferSchema=True)
pdf.show()
```

```
+-------+---------+---------+--------+-----+----+
|Item Id|Item Name|Item Cost|Supplier|Grade| Qty|
+-------+---------+---------+--------+-----+----+
|      4|    Chock|    65.76|       X|    A|  34|
|      5|   Pencil|    45.65|       Y|    B|null|
|      6|      Pen|    76.87|       X|    A|  23|
|      7|   Duster|    54.0|        Y|    C|  25|
|      8|     null|     null|       Z| null|  34|
|      9|     book|     53.0|    null| null|  65|
|      7|   Duster|    54.0|        Y|    C|  25|
|      8|     null|     null|       Z| null|  34|
+-------+---------+---------+--------+-----+----+
```

In [45]:
```
pdf.createOrReplaceTempView("tempview")
```

In [46]:
```
result=spark.sql("select * from tempview").limit(3).toPandas()
result
```

Out[46]:

| | Item Id | Item Name | Item Cost | Supplier | Grade | Qty |
|---|---|---|---|---|---|---|
| **0** | 4 | Chock | 65.76 | X | A | 34.0 |
| **1** | 5 | Pencil | 45.65 | Y | B | NaN |
| **2** | 6 | Pen | 76.87 | X | A | 23.0 |

In [ ]:
```
# Assignment
```

In [2]:
```
pdf1=spark.read.option("header","true").csv('D:\items.csv',inferSchema=True)
pdf1.show()
pdf2=spark.read.option("header","true").csv('D:\Supplier.csv',inferSchema=True)
pdf2.show()
```

```
+------+---------+--------+-------+----------+
|ItemId|ItemName|ItemCost|ItemQty|SupplierId|
+------+---------+--------+-------+----------+
|     4|    Chock|   65.76|     23|         X|
|     5|   Pencil|   45.65|     34|         Y|
|     6|      Pen|   76.87|     32|         X|
|     7|   Duster|    54.0|     10|         Z|
|     8|     null|    23.0|     45|         Y|
|     9|     book|    53.0|     25|      null|
+------+---------+--------+-------+----------+

+----------+-----+
|SupplierId|Grade|
+----------+-----+
|         X|    A|
|         Y|    B|
|         Z|    C|
+----------+-----+
```

In [3]:
```
pdf1.createOrReplaceTempView("tempview1")
pdf2.createOrReplaceTempView("tempview2")
```

In [8]:
```
# 1.Show item details supplied by MR.X
res=spark.sql("select * from tempview1 where SupplierId == 'X'")
res.toPandas()
```

Out[8]:

| | ItemId | ItemName | ItemCost | ItemQty | SupplierId |
|---|---|---|---|---|---|
| **0** | 4 | Chock | 65.76 | 23 | X |
| **1** | 6 | Pen | 76.87 | 32 | X |

In [4]:
```
# 2.Show supplier details who has supplied items cost>1000
pdf=pdf1.join(pdf2,pdf1.SupplierId==pdf2.SupplierId,'inner')
pdf.createOrReplaceTempView("tempview")
res=spark.sql("select * from tempview where ItemCost > 50")
res.toPandas()
```

Out[4]:

| | ItemId | ItemName | ItemCost | ItemQty | SupplierId | SupplierId | Grade |
|---|---|---|---|---|---|---|---|
| **0** | 4 | Chock | 65.76 | 23 | X | X | A |
| **1** | 6 | Pen | 76.87 | 32 | X | X | A |
| **2** | 7 | Duster | 54.00 | 10 | Z | Z | C |

In [10]:
```
# 3.show all item details whos supplier details are not available
pdf=pdf1.join(pdf2,pdf1.SupplierId==pdf2.SupplierId,'leftanti')
pdf.toPandas()
```

Out[10]:

| | ItemId | ItemName | ItemCost | ItemQty | SupplierId |
|---|---|---|---|---|---|
| **0** | 9 | book | 53.0 | 25 | None |

In [11]:
```
# 4.show item details whose supplier details are available
pdf=pdf1.join(pdf2,pdf1.SupplierId==pdf2.SupplierId,'leftsemi')
pdf.toPandas()
```

Out[11]:

| | ItemId | ItemName | ItemCost | ItemQty | SupplierId |
|---|---|---|---|---|---|
| **0** | 4 | Chock | 65.76 | 23 | X |
| **1** | 5 | Pencil | 45.65 | 34 | Y |
| **2** | 6 | Pen | 76.87 | 32 | X |
| **3** | 7 | Duster | 54.00 | 10 | Z |
| **4** | 8 | None | 23.00 | 45 | Y |

In [5]:
```
# 5.show item details along with supplier details for such items ,for which supplier details are  available
# and item name starts with 'P'
res=spark.sql("select * from tempview where ItemName LIKE 'P%'")
res.toPandas()
```

Out[5]:

| | ItemId | ItemName | ItemCost | ItemQty | SupplierId | SupplierId | Grade |
|---|---|---|---|---|---|---|---|
| **0** | 5 | Pencil | 45.65 | 34 | Y | Y | B |
| **1** | 6 | Pen | 76.87 | 32 | X | X | A |

In [15]:
```
# 6.Show supplier wise number of items supplied, sum ,min ,max total of itemcost . all item cost supplied
pdf=spark.sql("select min(ItemCost),max(ItemCost),sum(ItemCost) from tempview")
pdf.show()
```

```
+------------+------------+------------+
|min(ItemCost)|max(ItemCost)|sum(ItemCost)|
+------------+------------+------------+
|        23.0|       76.87|      265.28|
+------------+------------+------------+
```

In [15]:
```
# 7.join overall items available in 2 stores
pdf=pdf1.join(pdf2,pdf1.SupplierId==pdf2.SupplierId,'inner')
pdf.toPandas()
```

Out[15]:

| | ItemId | ItemName | ItemCost | ItemQty | SupplierId | SupplierId | Grade |
|---|---|---|---|---|---|---|---|
| **0** | 4 | Chock | 65.76 | 23 | X | X | A |
| **1** | 5 | Pencil | 45.65 | 34 | Y | Y | B |
| **2** | 6 | Pen | 76.87 | 32 | X | X | A |
| **3** | 7 | Duster | 54.00 | 10 | Z | Z | C |
| **4** | 8 | None | 23.00 | 45 | Y | Y | B |
| **5** | 9 | book | 53.00 | 25 | Z | Z | C |

In [ ]:
```

```