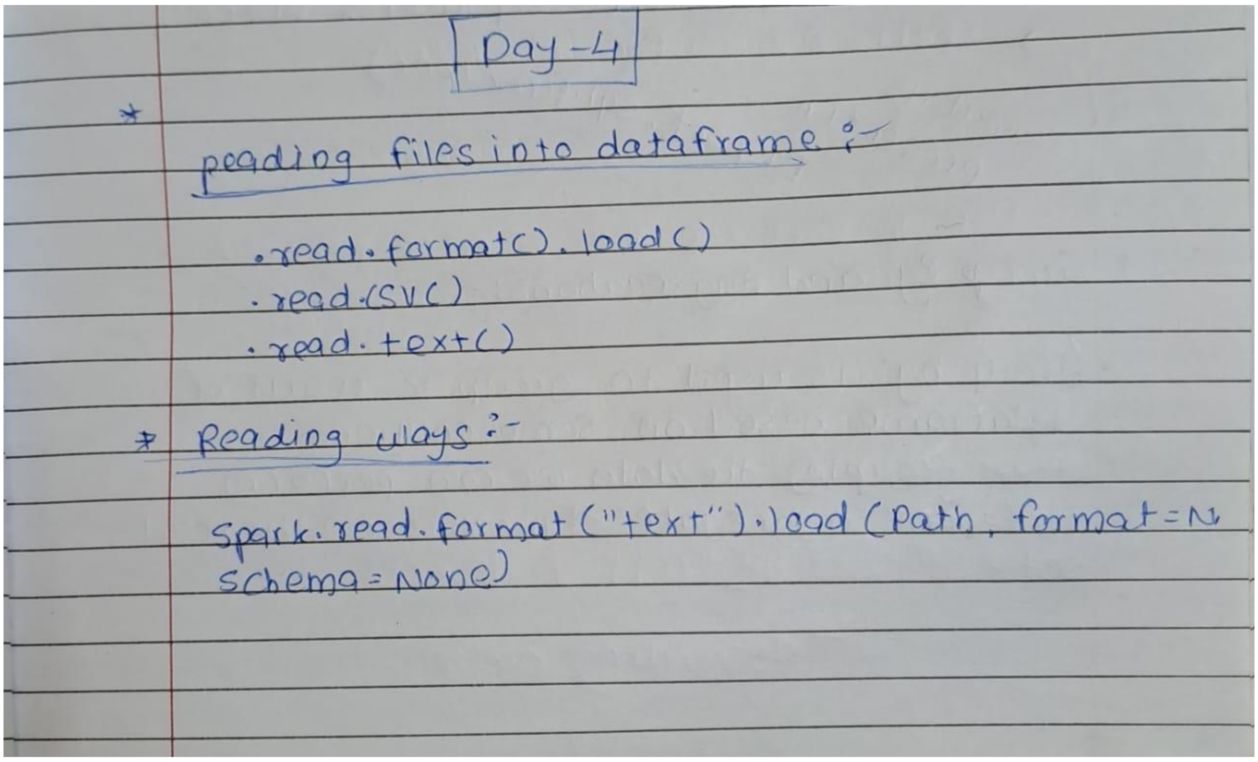
PySpark Assignment 4

Pratik Wani

- Notes



$\boxed{\text{Day - 4}}$

* Reading files into dataframe :-

.read.format().load()
.read.csv()
.read.text()

* Reading ways :-

```
Spark.read.format("text").load(Path, format=N
schema=None)
```

PAGE NO.:

* Adding new column

* we have to use withcolumn() function with lit() ~~tas~~ as parameter to give constant values to the column.

dataframe.withcolumn("column-name, lit(vale)

Lit funtion is present in pyspark.sql.function library.

* concate two columns

concat_ws()

Syntax → concat ws ("seprator", 'onecol', 'twocol'

concat-ws present in pyspark. sql.function()

* Group By and Aggregations:-

- group by is used to group to result of dataframe based on some columns.
- after grouping the data we can perform different aggregate Functions on the result set.

• such as sum(), min(), count(), max(), Avg()

Syntax:-

dataframe.groupBy('Department').sum('salary')

$\nearrow$

grouping the data based on department

Note → ① inferschema = True → df infers the data according to different datatypes.

* pivot() → It can convert some row values into the columns.

* Missing value:-

1) To drop the NULL values we can use drop() function.

dataframe.NA.drop()

3 parameters:-

① How = 'all'
→ only drop if all the values are NULL

② How = 'any'
→ drop if any value is NULL

③ thresh = 2
→ means if at least two values must be Not NULL then only data is not droped.

④ subset = ['col name']
→ only check null values for these column.

* Fill NULL values :-

- .fill('string') ~ for string value
- .fill(0) ~ for int values

* Sort :-

use to sort the data based on perticular column.

3 ways :-

ⓐ
df.sort("col1") → ascending order

ⓑ df.sort(df["col1"].desc())
↳ descending order

PAGE NO.:

(c) multiple column

```
df. sort ("col1", "col2")
```

* <u>order by</u>

we can also use orderby to sort values

```
df.orderby ("col")
```

* <u>joins</u>

similar as SQL

<u>Syntax :-</u>

- df1.join (df2, df1.col_name = df2.col_name,
 'type')

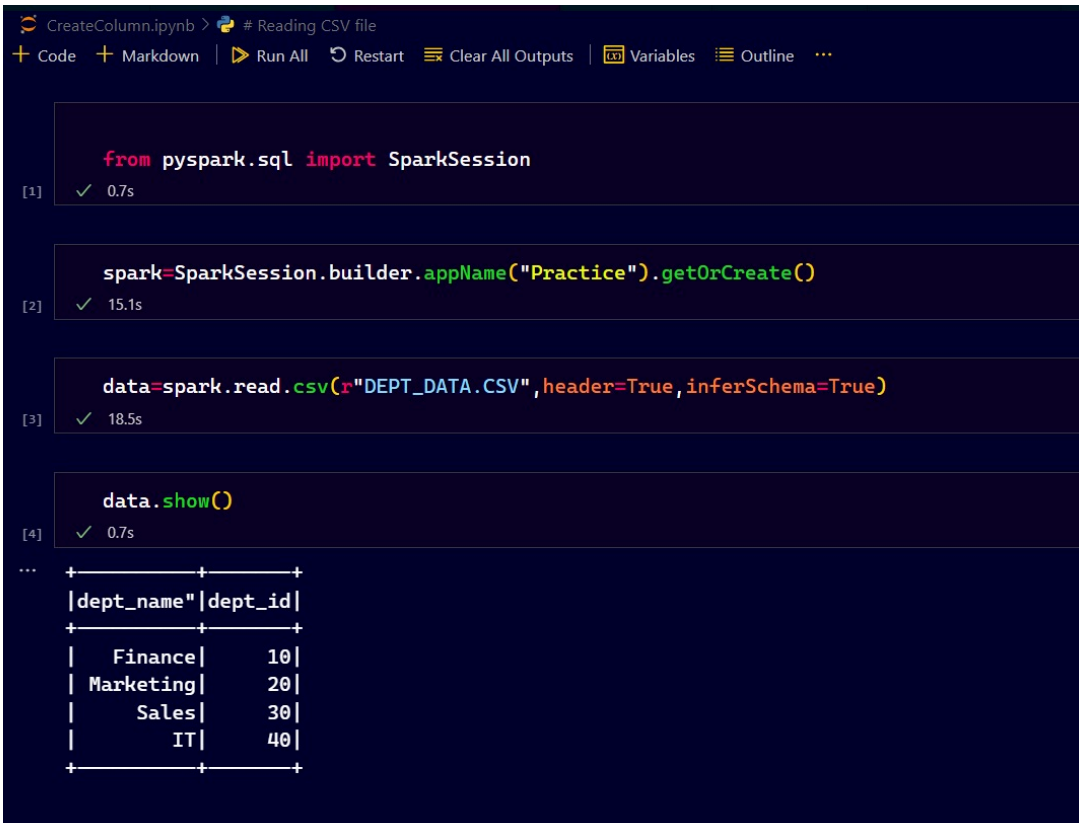
- <u>New types</u>

(a) <u>Left semi :-</u> ~~Add~~ Match from right

(b) <u>Left Anti :-</u> unmatch from right

(c) <u>Left outer :-</u> semi + Anti

- Reading Text File and Adding New Column:


CreateColumn.ipynb > # Reading CSV file
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

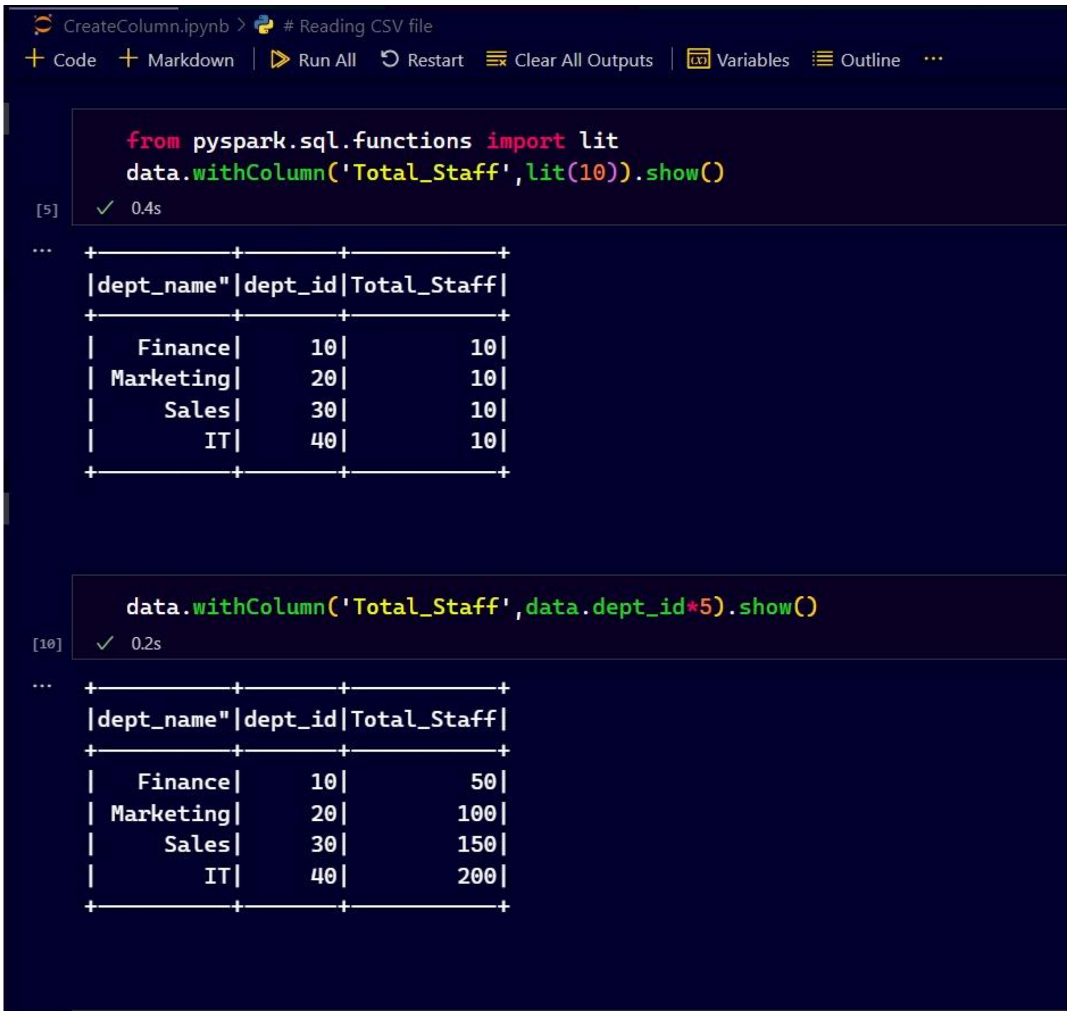
```
[1]
from pyspark.sql import SparkSession
✓ 0.7s
```

```
[2]
spark=SparkSession.builder.appName("Practice").getOrCreate()
✓ 15.1s
```

```
[3]
data=spark.read.csv(r"DEPT_DATA.CSV",header=True,inferSchema=True)
✓ 18.5s
```

```
[4]
data.show()
✓ 0.7s
```

```
+----------+-------+
|dept_name"|dept_id|
+----------+-------+
|   Finance|     10|
| Marketing|     20|
|     Sales|     30|
|        IT|     40|
+----------+-------+
```

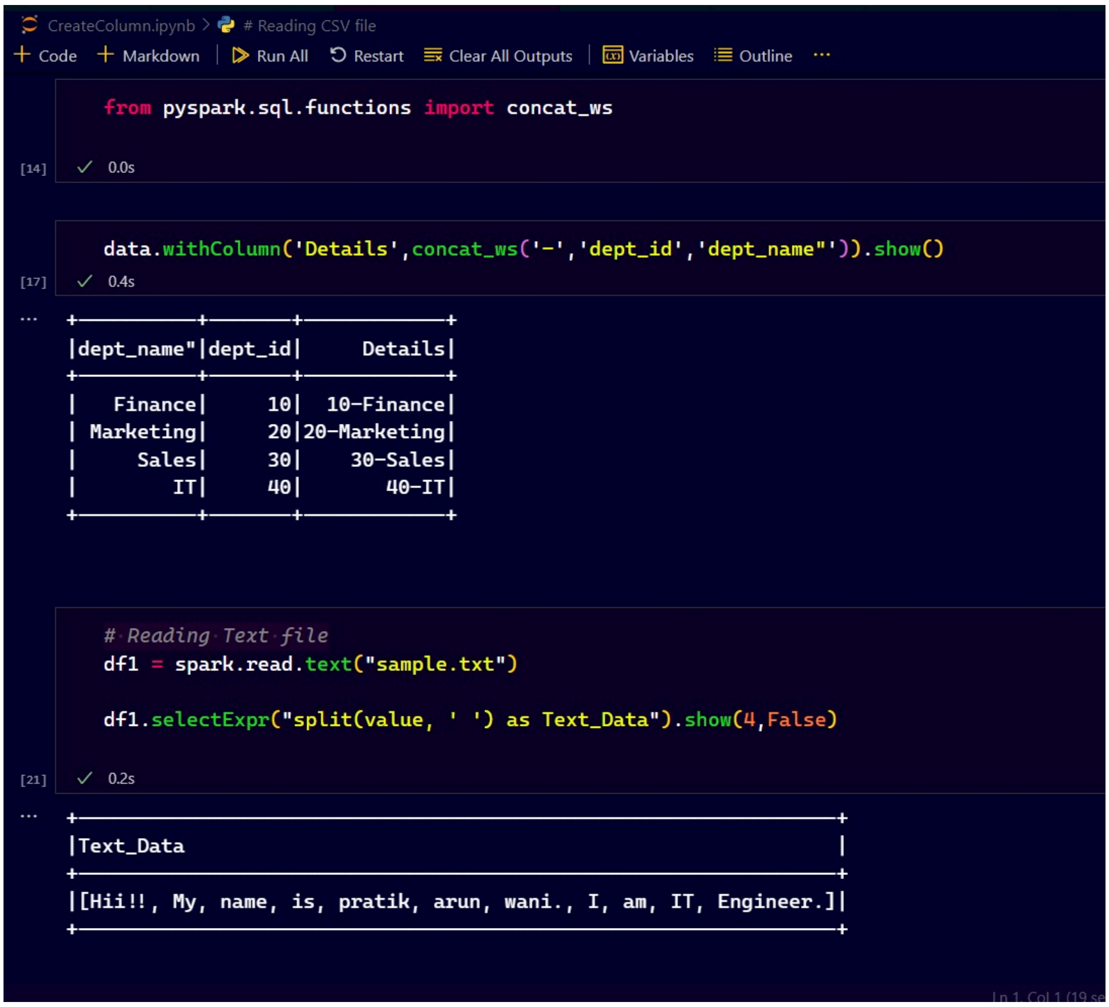

CreateColumn.ipynb > # Reading CSV file
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

```
[5]
from pyspark.sql.functions import lit
data.withColumn('Total_Staff',lit(10)).show()
✓ 0.4s
```

```
+----------+-------+-----------+
|dept_name"|dept_id|Total_Staff|
+----------+-------+-----------+
|   Finance|     10|         10|
| Marketing|     20|         10|
|     Sales|     30|         10|
|        IT|     40|         10|
+----------+-------+-----------+
```

```
[10]
data.withColumn('Total_Staff',data.dept_id*5).show()
✓ 0.2s
```

```
+----------+-------+-----------+
|dept_name"|dept_id|Total_Staff|
+----------+-------+-----------+
|   Finance|     10|         50|
| Marketing|     20|        100|
|     Sales|     30|        150|
|        IT|     40|        200|
+----------+-------+-----------+
```


CreateColumn.ipynb > # Reading CSV file
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

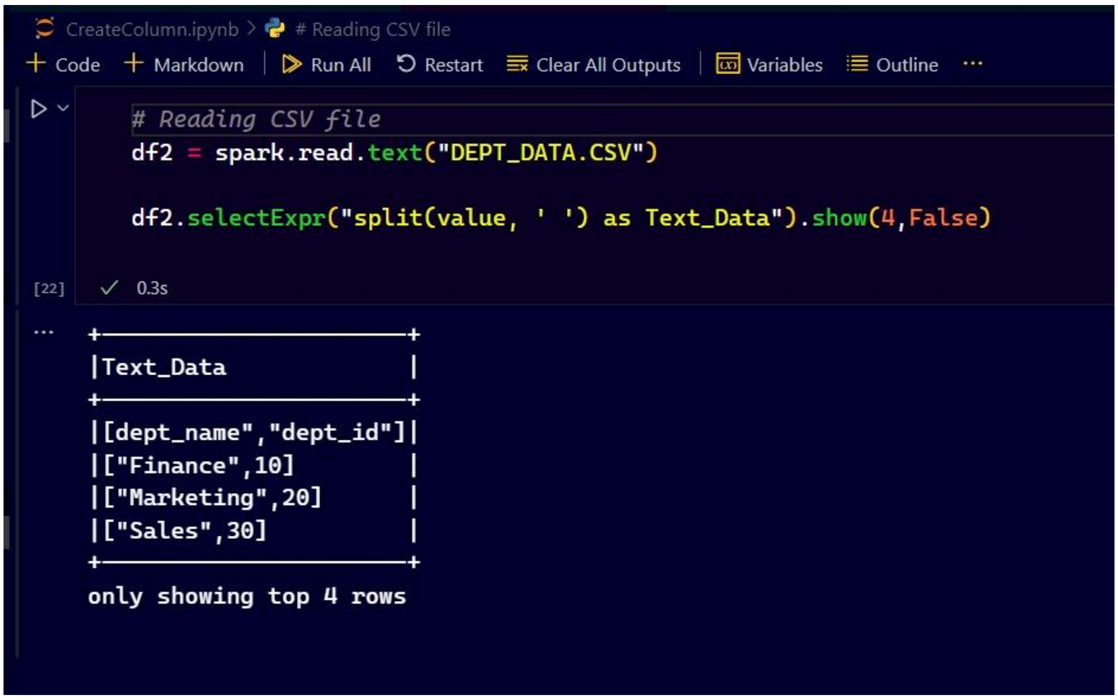
```
from pyspark.sql.functions import concat_ws
```
[14] ✓ 0.0s

```
data.withColumn('Details',concat_ws('-','dept_id','dept_name"')).show()
```
[17] ✓ 0.4s

```
+---------+-------+------------+
|dept_name"|dept_id|     Details|
+---------+-------+------------+
|   Finance|     10|  10-Finance|
| Marketing|     20|20-Marketing|
|     Sales|     30|    30-Sales|
|        IT|     40|       40-IT|
+---------+-------+------------+
```

```
# Reading Text file
df1 = spark.read.text("sample.txt")

df1.selectExpr("split(value, ' ') as Text_Data").show(4,False)
```
[21] ✓ 0.2s

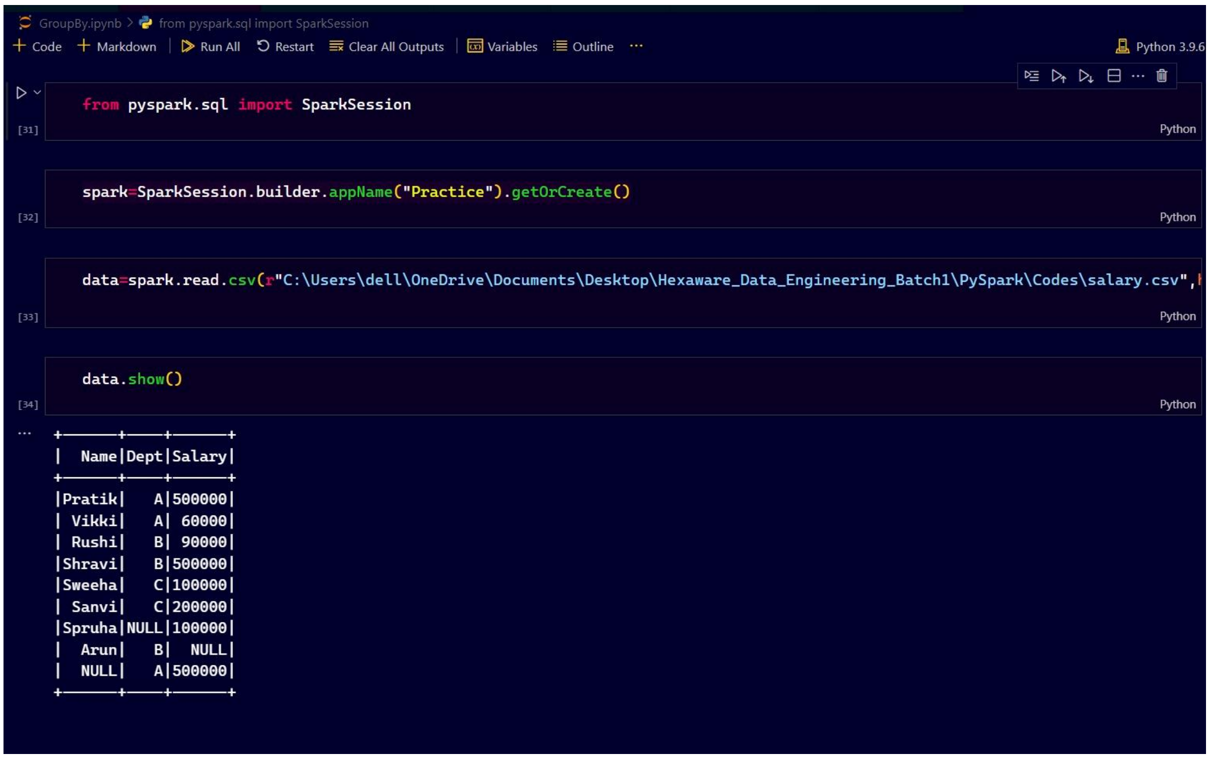
```
+--------------------------------------------------------------+
|Text_Data                                                     |
+--------------------------------------------------------------+
|[Hii!!, My, name, is, pratik, arun, wani., I, am, IT, Engineer.]|
+--------------------------------------------------------------+
```


CreateColumn.ipynb > # Reading CSV file
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

```
# Reading CSV file
df2 = spark.read.text("DEPT_DATA.CSV")

df2.selectExpr("split(value, ' ') as Text_Data").show(4,False)
```
[22] ✓ 0.3s

```
+----------------------+
|Text_Data             |
+----------------------+
|[dept_name","dept_id"]|
|["Finance",10]        |
|["Marketing",20]      |
|["Sales",30]          |
+----------------------+
only showing top 4 rows
```

- Group By, Handling Missing Values


GroupBy.ipynb > from pyspark.sql import SparkSession
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...
Python 3.9.6

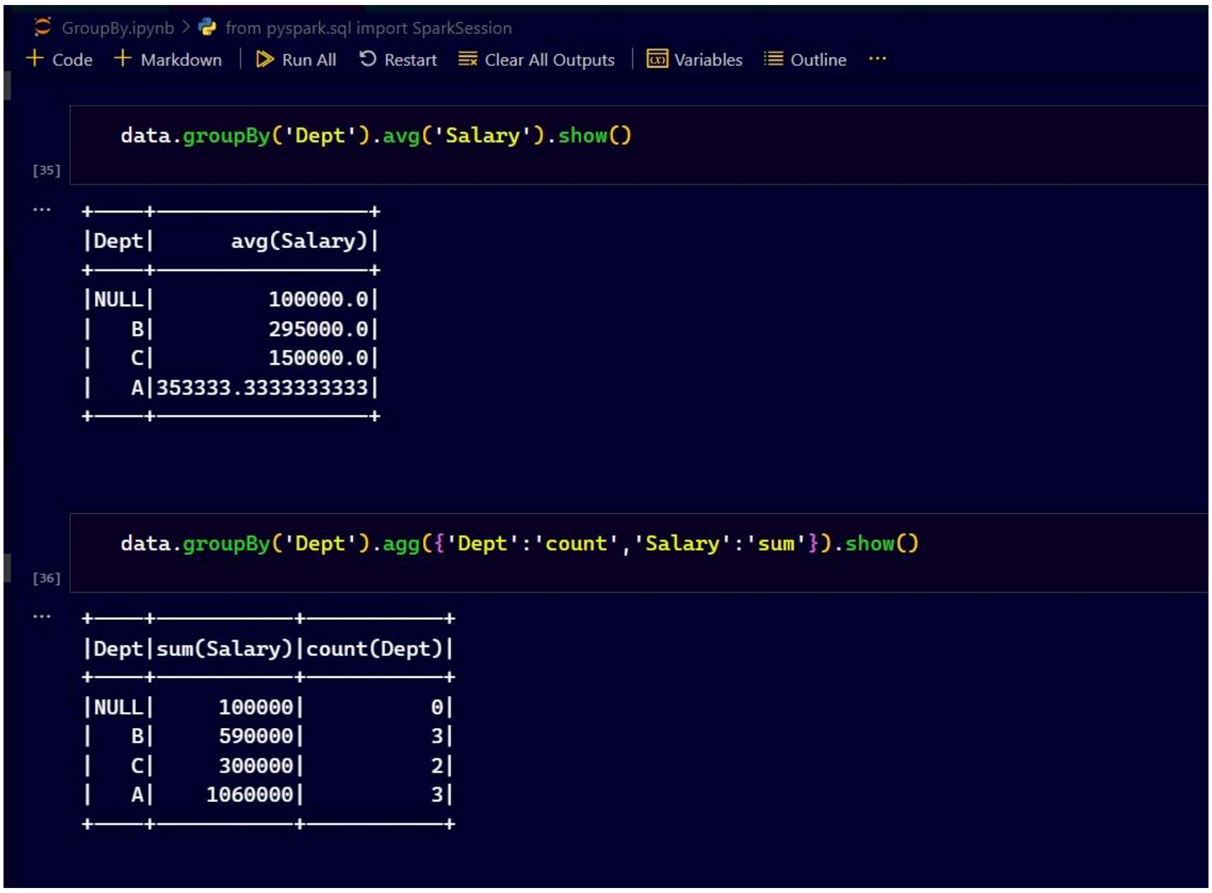
```
[31]
from pyspark.sql import SparkSession
```

```
[32]
spark=SparkSession.builder.appName("Practice").getOrCreate()
```

```
[33]
data=spark.read.csv(r"C:\Users\dell\OneDrive\Documents\Desktop\Hexaware_Data_Engineering_Batch1\PySpark\Codes\salary.csv",
```

```
[34]
data.show()
```

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
|Spruha|NULL|100000|
|  Arun|   B|  NULL|
|  NULL|   A|500000|
+------+----+------+
```

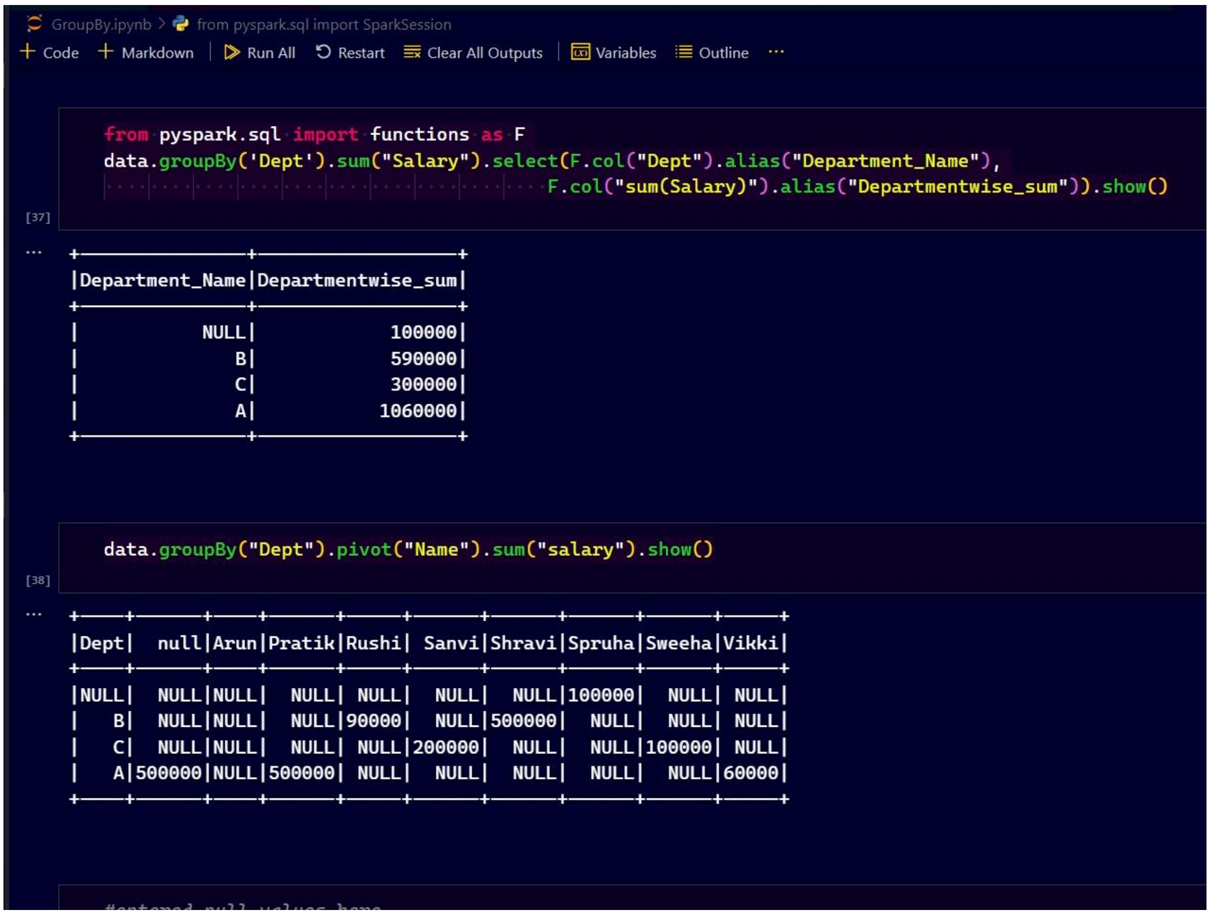

GroupBy.ipynb > from pyspark.sql import SparkSession
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

```
[35]
data.groupBy('Dept').avg('Salary').show()
```

```
+----+-----------------+
|Dept|      avg(Salary)|
+----+-----------------+
|NULL|         100000.0|
|   B|         295000.0|
|   C|         150000.0|
|   A|353333.3333333333|
+----+-----------------+
```

```
[36]
data.groupBy('Dept').agg({'Dept':'count','Salary':'sum'}).show()
```

```
+----+-----------+-----------+
|Dept|sum(Salary)|count(Dept)|
+----+-----------+-----------+
|NULL|     100000|          0|
|   B|     590000|          3|
|   C|     300000|          2|
|   A|    1060000|          3|
+----+-----------+-----------+
```



```
from pyspark.sql import functions as F
data.groupBy('Dept').sum("Salary").select(F.col("Dept").alias("Department_Name"),
                                          F.col("sum(Salary)").alias("Departmentwise_sum")).show()
```

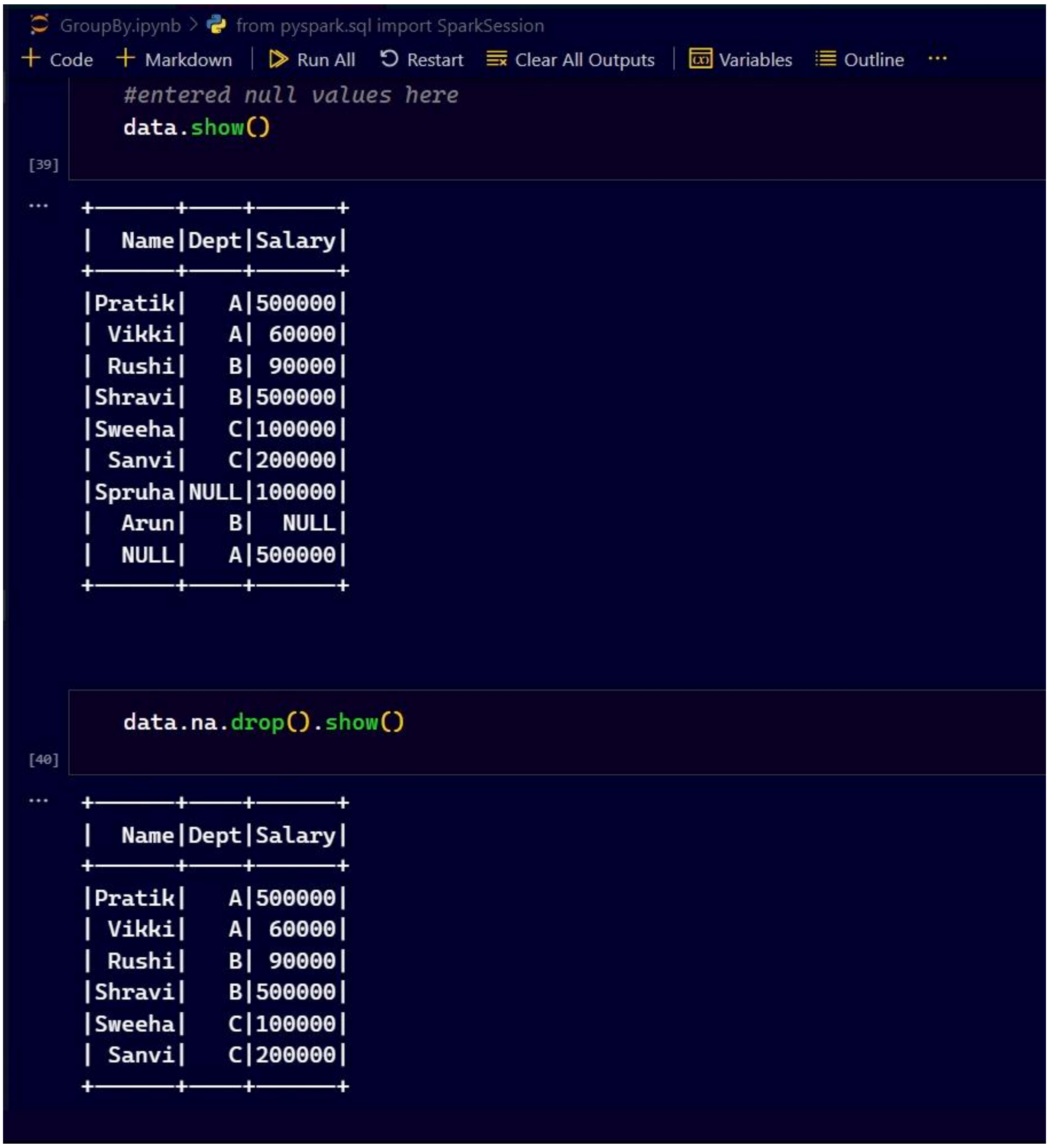
[37]

```
+---------------+------------------+
|Department_Name|Departmentwise_sum|
+---------------+------------------+
|           NULL|            100000|
|              B|            590000|
|              C|            300000|
|              A|           1060000|
+---------------+------------------+
```

```
data.groupBy("Dept").pivot("Name").sum("salary").show()
```

[38]

```
+----+------+----+------+-----+------+------+------+------+-----+
|Dept|  null|Arun|Pratik|Rushi| Sanvi|Shravi|Spruha|Sweeha|Vikki|
+----+------+----+------+-----+------+------+------+------+-----+
|NULL|  NULL|NULL|  NULL| NULL|  NULL|  NULL|100000|  NULL| NULL|
|   B|  NULL|NULL|  NULL|90000|  NULL|500000|  NULL|  NULL| NULL|
|   C|  NULL|NULL|  NULL| NULL|200000|  NULL|  NULL|100000| NULL|
|   A|500000|NULL|500000| NULL|  NULL|  NULL|  NULL|  NULL|60000|
+----+------+----+------+-----+------+------+------+------+-----+
```


GroupBy.ipynb > from pyspark.sql import SparkSession
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

```python
#entered null values here
data.show()
```

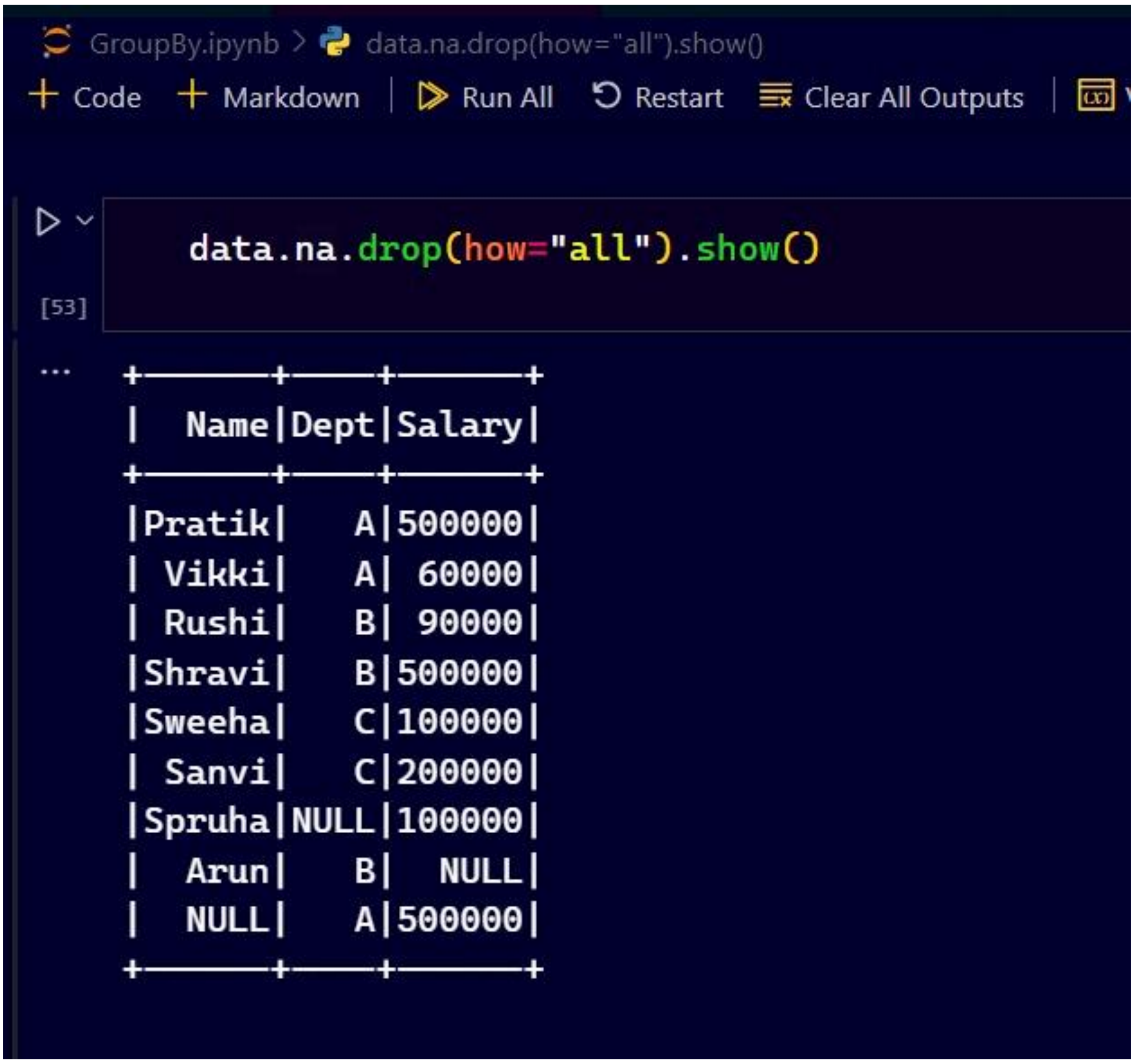
[39]

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
|Spruha|NULL|100000|
|  Arun|   B|  NULL|
|  NULL|   A|500000|
+------+----+------+
```

```python
data.na.drop().show()
```

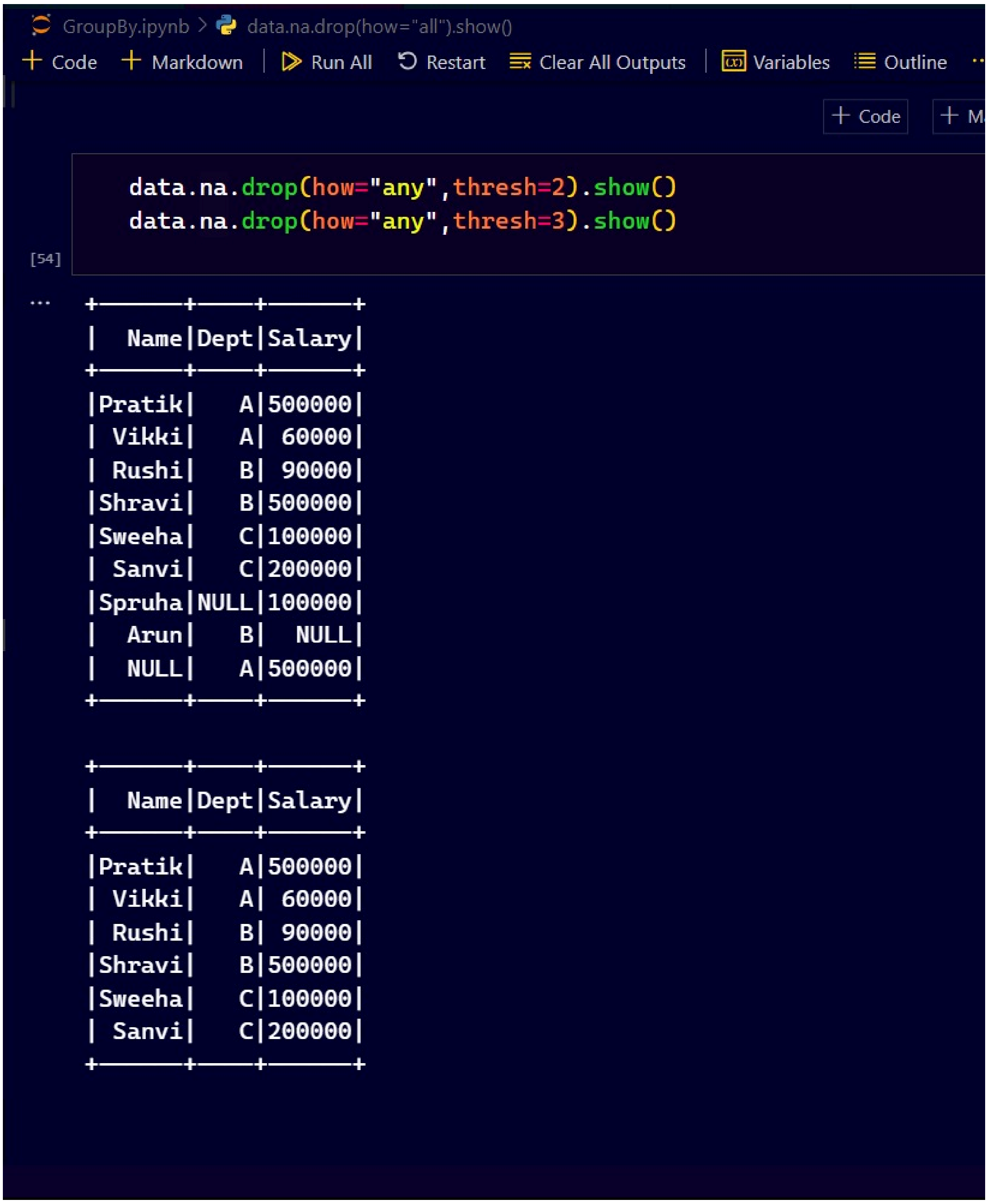
[40]

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
+------+----+------+
```



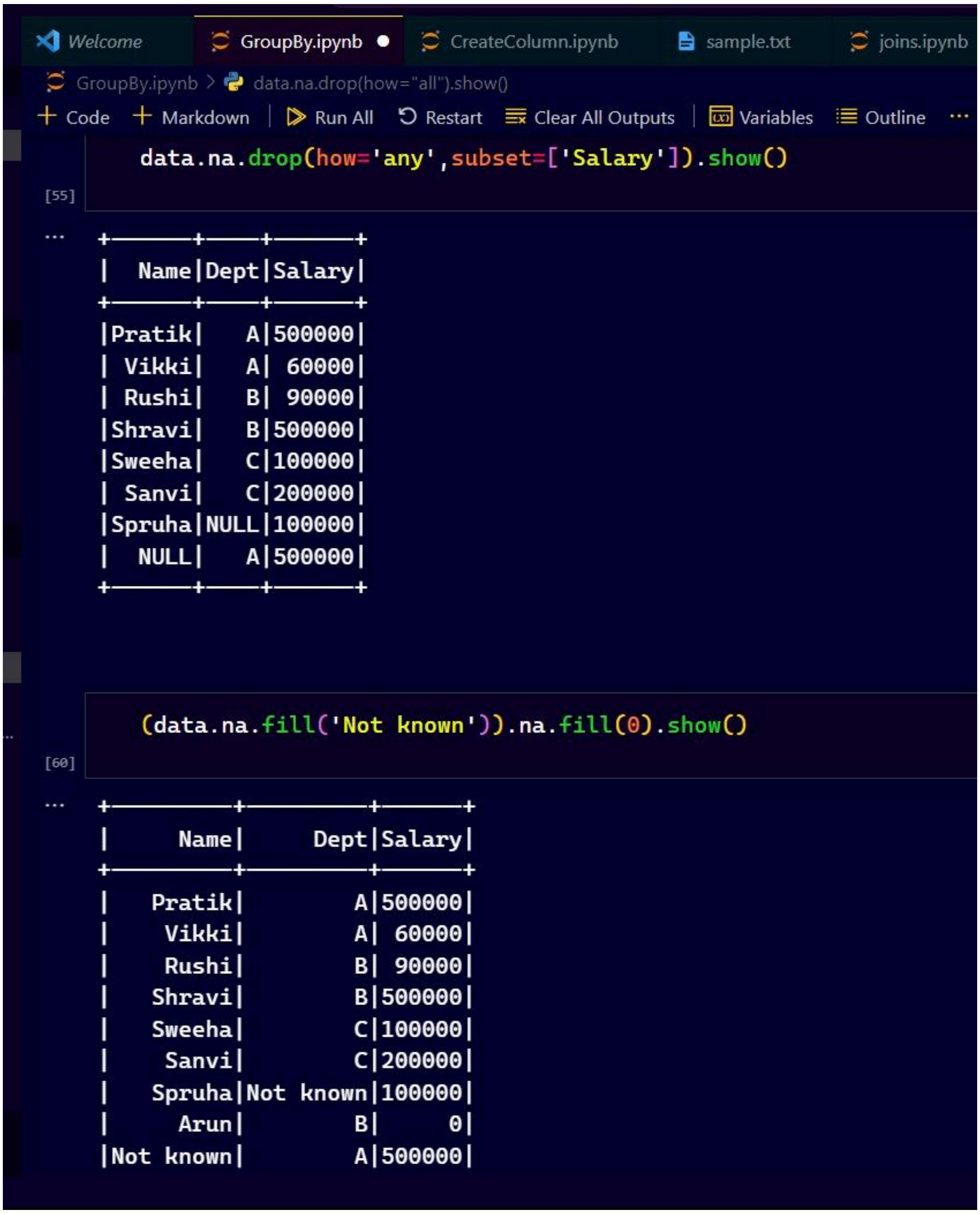
```
data.na.drop(how="all").show()
```

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
|Spruha|NULL|100000|
|  Arun|   B|  NULL|
|  NULL|   A|500000|
+------+----+------+
```



```
data.na.drop(how="any",thresh=2).show()
data.na.drop(how="any",thresh=3).show()
```

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
|Spruha|NULL|100000|
|  Arun|   B|  NULL|
|  NULL|   A|500000|
+------+----+------+

+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
+------+----+------+
```



```
data.na.drop(how='any',subset=['Salary']).show()
```

[55]

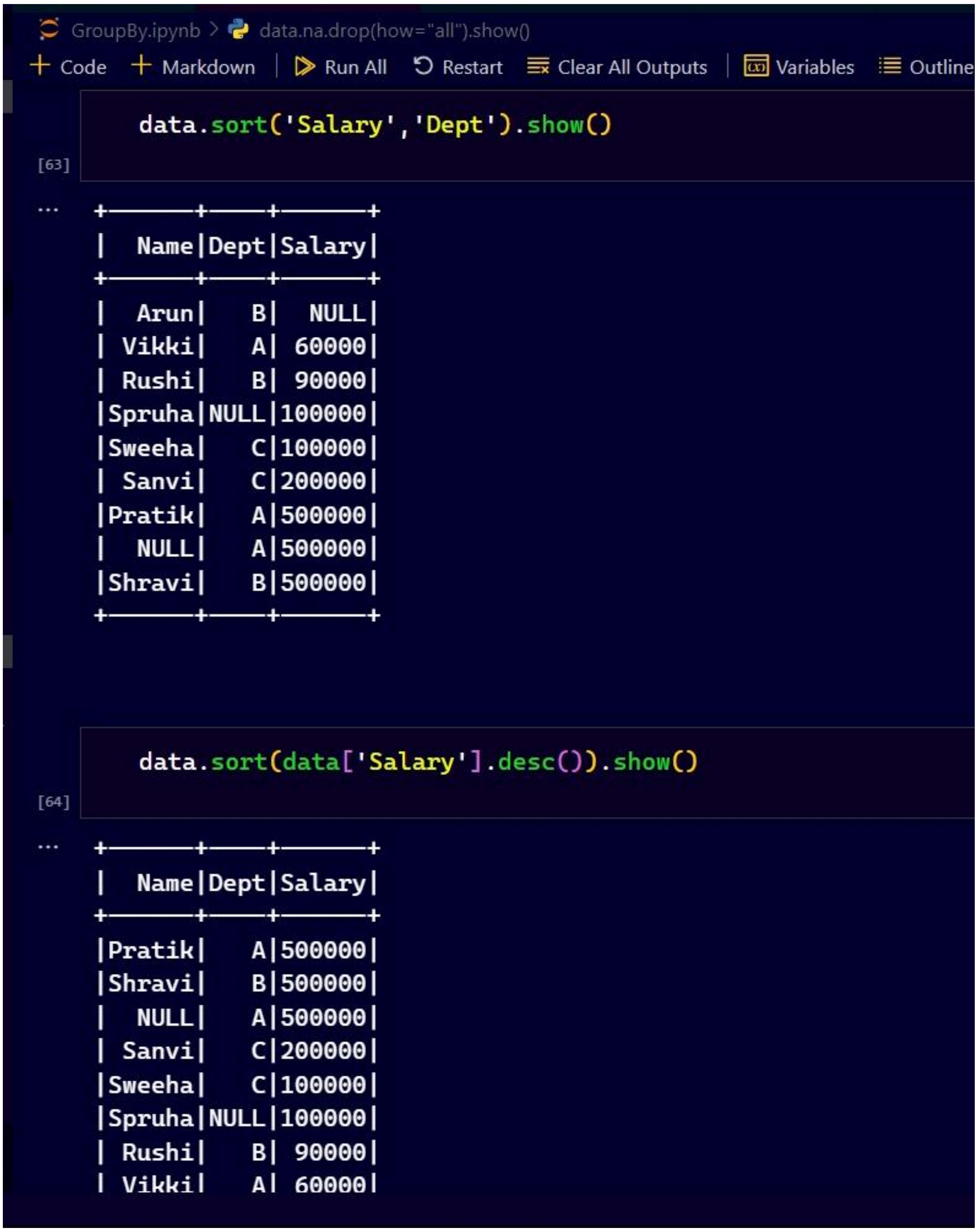
```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Shravi|   B|500000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
|Spruha|NULL|100000|
|  NULL|   A|500000|
+------+----+------+
```

```
(data.na.fill('Not known')).na.fill(0).show()
```

[60]

```
+---------+---------+------+
|     Name|     Dept|Salary|
+---------+---------+------+
|   Pratik|        A|500000|
|    Vikki|        A| 60000|
|    Rushi|        B| 90000|
|   Shravi|        B|500000|
|   Sweeha|        C|100000|
|    Sanvi|        C|200000|
|   Spruha|Not known|100000|
|     Arun|        B|     0|
|Not known|        A|500000|
```

- Sorting



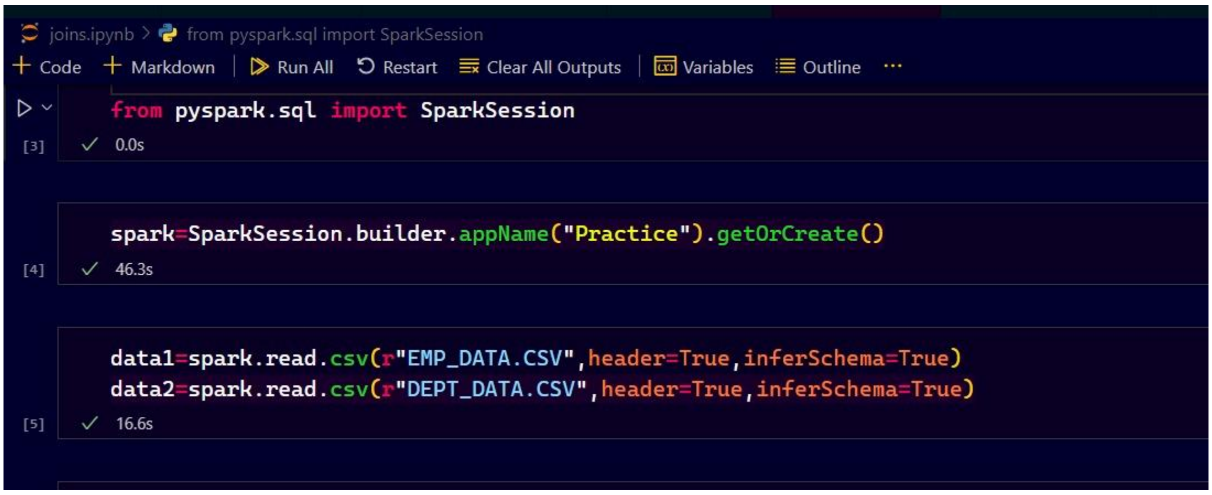
```
[63]
data.sort('Salary','Dept').show()
```

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|  Arun|   B|  NULL|
| Vikki|   A| 60000|
| Rushi|   B| 90000|
|Spruha|NULL|100000|
|Sweeha|   C|100000|
| Sanvi|   C|200000|
|Pratik|   A|500000|
|  NULL|   A|500000|
|Shravi|   B|500000|
+------+----+------+
```

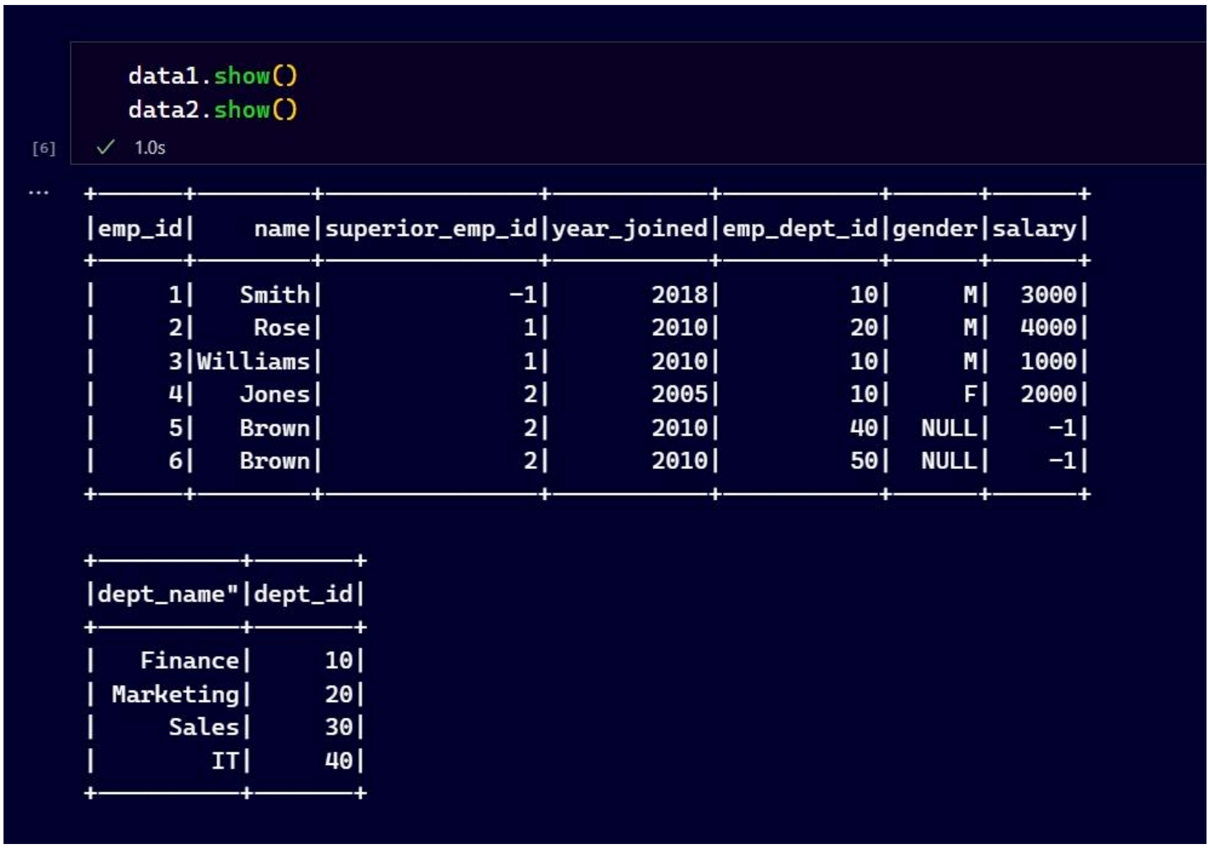
```
[64]
data.sort(data['Salary'].desc()).show()
```

```
+------+----+------+
|  Name|Dept|Salary|
+------+----+------+
|Pratik|   A|500000|
|Shravi|   B|500000|
|  NULL|   A|500000|
| Sanvi|   C|200000|
|Sweeha|   C|100000|
|Spruha|NULL|100000|
| Rushi|   B| 90000|
| Vikki|   A| 60000|
```

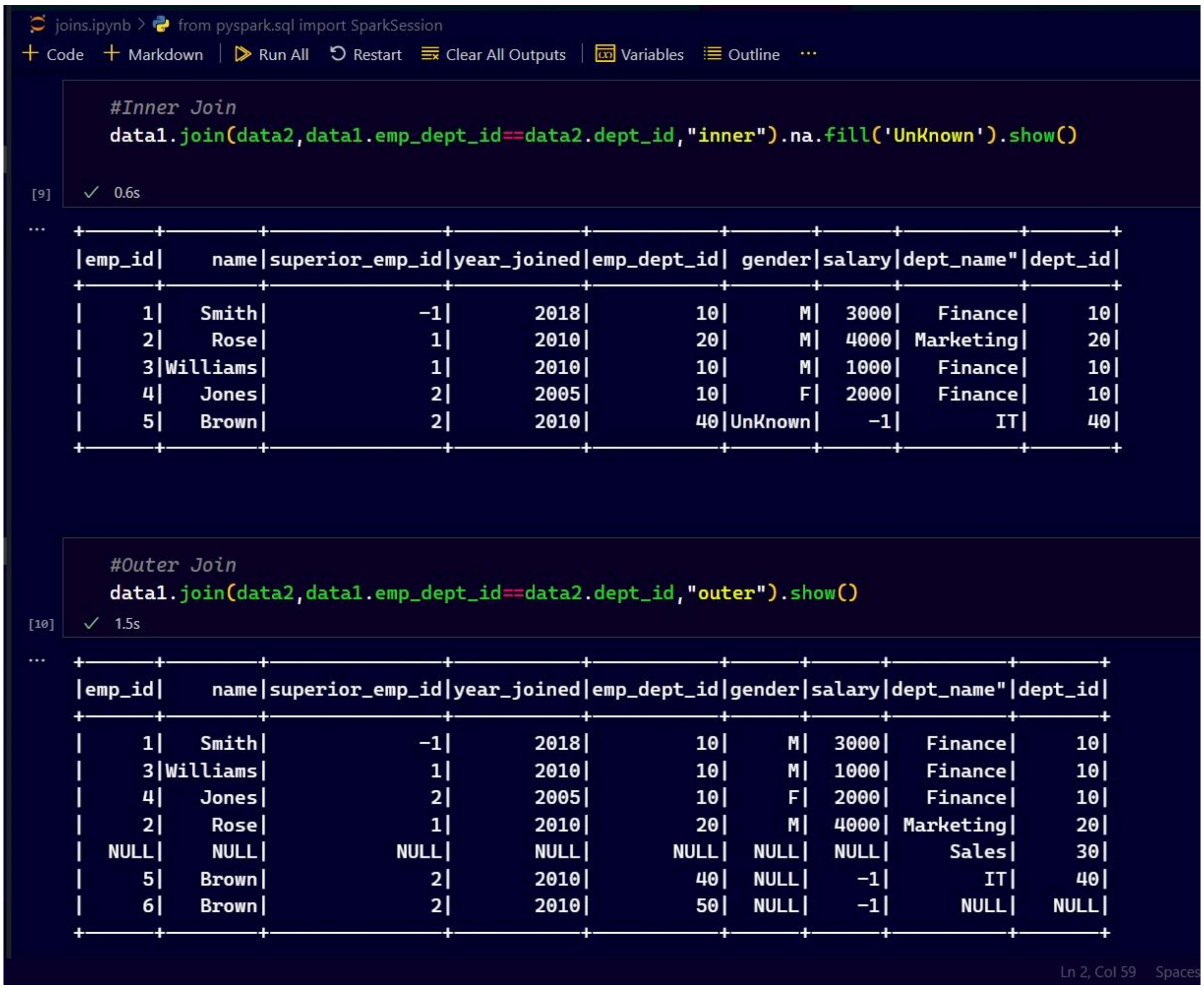
- Joins



```
from pyspark.sql import SparkSession
[3]  0.0s

spark=SparkSession.builder.appName("Practice").getOrCreate()
[4]  46.3s

data1=spark.read.csv(r"EMP_DATA.CSV",header=True,inferSchema=True)
data2=spark.read.csv(r"DEPT_DATA.CSV",header=True,inferSchema=True)
[5]  16.6s
```



```
data1.show()
data2.show()
[6]  1.0s

+------+--------+---------------+-----------+-----------+------+------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id|gender|salary|
+------+--------+---------------+-----------+-----------+------+------+
|     1|   Smith|             -1|       2018|         10|     M|  3000|
|     2|    Rose|              1|       2010|         20|     M|  4000|
|     3|Williams|              1|       2010|         10|     M|  1000|
|     4|   Jones|              2|       2005|         10|     F|  2000|
|     5|   Brown|              2|       2010|         40|  NULL|    -1|
|     6|   Brown|              2|       2010|         50|  NULL|    -1|
+------+--------+---------------+-----------+-----------+------+------+

+----------+-------+
|dept_name"|dept_id|
+----------+-------+
|   Finance|     10|
| Marketing|     20|
|     Sales|     30|
|        IT|     40|
+----------+-------+
```



```
#Inner Join
data1.join(data2,data1.emp_dept_id==data2.dept_id,"inner").na.fill('Unknown').show()
```

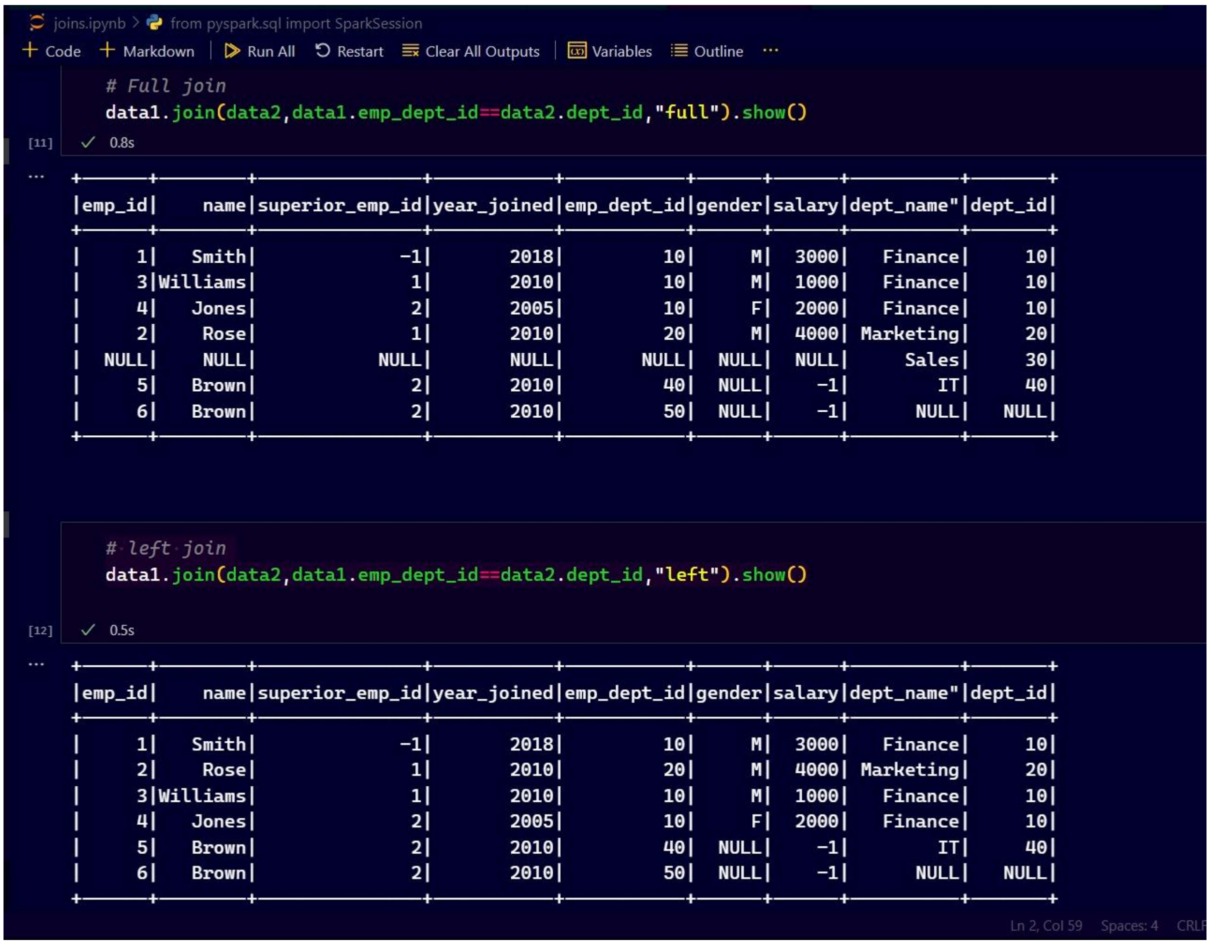
[9] ✓ 0.6s

```
+------+--------+---------------+-----------+-----------+-------+------+----------+-------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id| gender|salary|dept_name"|dept_id|
+------+--------+---------------+-----------+-----------+-------+------+----------+-------+
|     1|   Smith|             -1|       2018|         10|      M|  3000|   Finance|     10|
|     2|    Rose|              1|       2010|         20|      M|  4000| Marketing|     20|
|     3|Williams|              1|       2010|         10|      M|  1000|   Finance|     10|
|     4|   Jones|              2|       2005|         10|      F|  2000|   Finance|     10|
|     5|   Brown|              2|       2010|         40|Unknown|    -1|        IT|     40|
+------+--------+---------------+-----------+-----------+-------+------+----------+-------+
```

```
#Outer Join
data1.join(data2,data1.emp_dept_id==data2.dept_id,"outer").show()
```

[10] ✓ 1.5s

```
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id|gender|salary|dept_name"|dept_id|
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
|     1|   Smith|             -1|       2018|         10|     M|  3000|   Finance|     10|
|     3|Williams|              1|       2010|         10|     M|  1000|   Finance|     10|
|     4|   Jones|              2|       2005|         10|     F|  2000|   Finance|     10|
|     2|    Rose|              1|       2010|         20|     M|  4000| Marketing|     20|
|  NULL|    NULL|           NULL|       NULL|       NULL|  NULL|  NULL|     Sales|     30|
|     5|   Brown|              2|       2010|         40|  NULL|    -1|        IT|     40|
|     6|   Brown|              2|       2010|         50|  NULL|    -1|      NULL|   NULL|
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
```



```
# Full join
data1.join(data2,data1.emp_dept_id==data2.dept_id,"full").show()
```

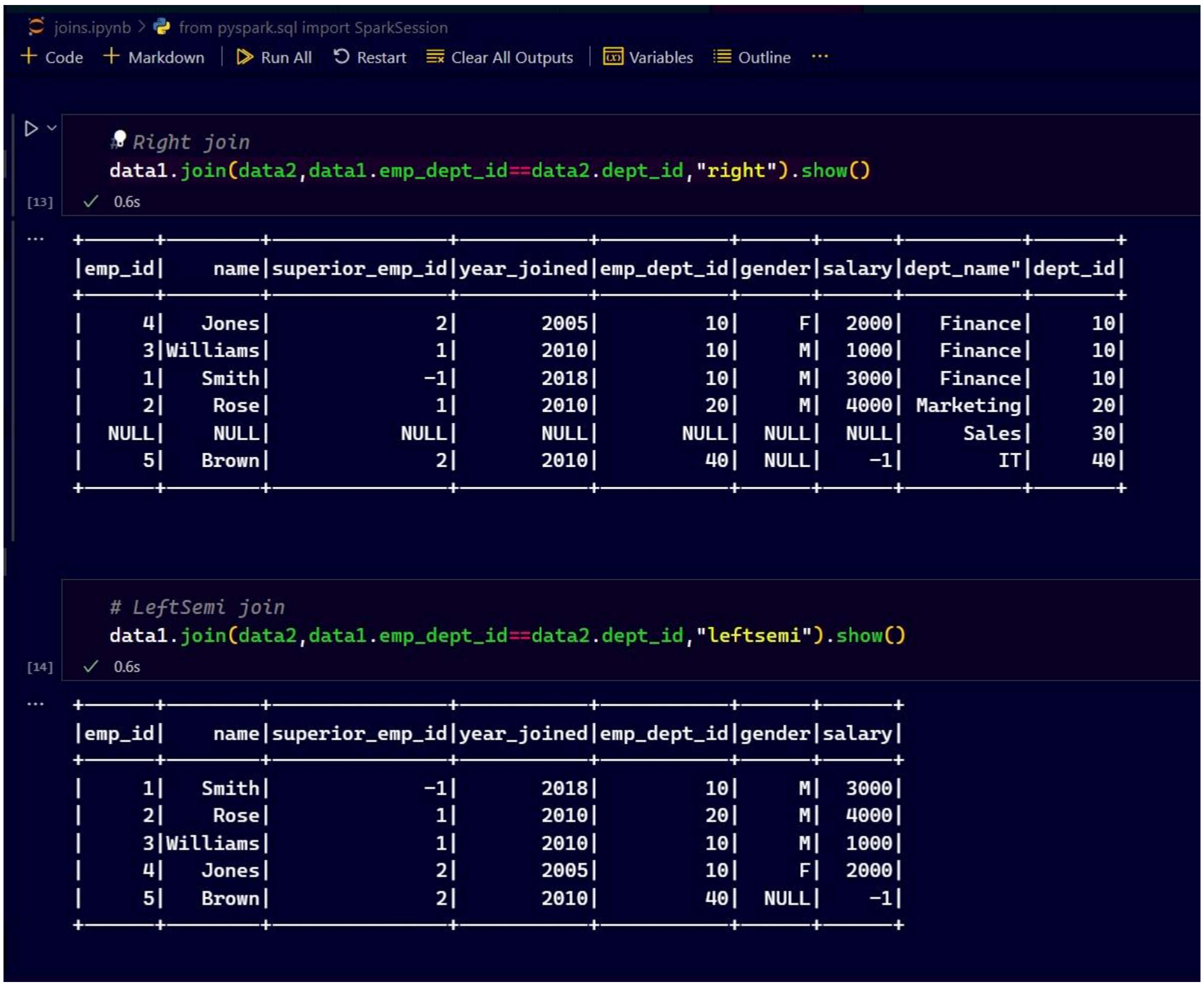
[11] ✓ 0.8s

```
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id|gender|salary|dept_name"|dept_id|
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
|     1|   Smith|             -1|       2018|         10|     M|  3000|   Finance|     10|
|     3|Williams|              1|       2010|         10|     M|  1000|   Finance|     10|
|     4|   Jones|              2|       2005|         10|     F|  2000|   Finance|     10|
|     2|    Rose|              1|       2010|         20|     M|  4000| Marketing|     20|
|  NULL|    NULL|           NULL|       NULL|       NULL|  NULL|  NULL|     Sales|     30|
|     5|   Brown|              2|       2010|         40|  NULL|    -1|        IT|     40|
|     6|   Brown|              2|       2010|         50|  NULL|    -1|      NULL|   NULL|
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
```

```
# left join
data1.join(data2,data1.emp_dept_id==data2.dept_id,"left").show()
```

[12] ✓ 0.5s

```
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id|gender|salary|dept_name"|dept_id|
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
|     1|   Smith|             -1|       2018|         10|     M|  3000|   Finance|     10|
|     2|    Rose|              1|       2010|         20|     M|  4000| Marketing|     20|
|     3|Williams|              1|       2010|         10|     M|  1000|   Finance|     10|
|     4|   Jones|              2|       2005|         10|     F|  2000|   Finance|     10|
|     5|   Brown|              2|       2010|         40|  NULL|    -1|        IT|     40|
|     6|   Brown|              2|       2010|         50|  NULL|    -1|      NULL|   NULL|
+------+--------+---------------+-----------+-----------+------+------+----------+-------+
```


joins.ipynb > from pyspark.sql import SparkSession
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline ...

```
# Right join
data1.join(data2,data1.emp_dept_id==data2.dept_id,"right").show()
```

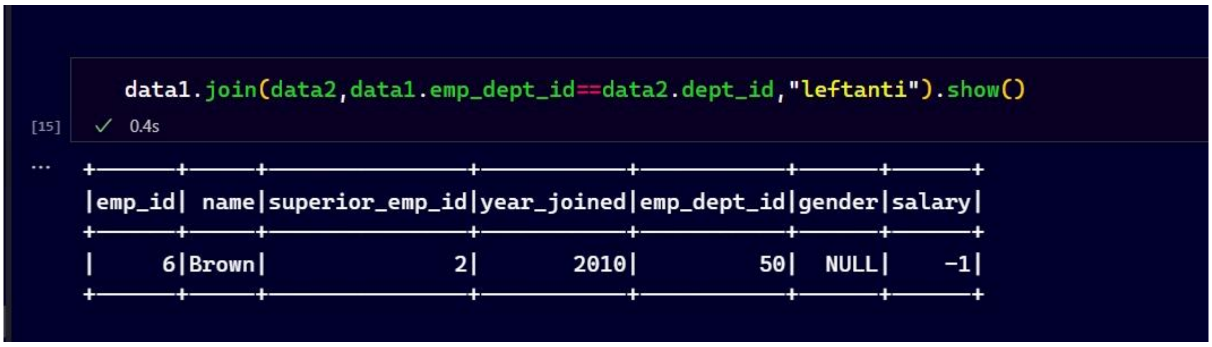
[13] ✓ 0.6s

```
+------+--------+-----------------+-----------+-----------+------+------+---------+-------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id|gender|salary|dept_name"|dept_id|
+------+--------+-----------------+-----------+-----------+------+------+---------+-------+
|     4|   Jones|                2|       2005|         10|     F|  2000|  Finance|     10|
|     3|Williams|                1|       2010|         10|     M|  1000|  Finance|     10|
|     1|   Smith|               -1|       2018|         10|     M|  3000|  Finance|     10|
|     2|    Rose|                1|       2010|         20|     M|  4000|Marketing|     20|
|  NULL|    NULL|             NULL|       NULL|       NULL|  NULL|  NULL|    Sales|     30|
|     5|   Brown|                2|       2010|         40|  NULL|    -1|       IT|     40|
+------+--------+-----------------+-----------+-----------+------+------+---------+-------+
```

```
# LeftSemi join
data1.join(data2,data1.emp_dept_id==data2.dept_id,"leftsemi").show()
```

[14] ✓ 0.6s

```
+------+--------+---------------+-----------+-----------+------+------+
|emp_id|    name|superior_emp_id|year_joined|emp_dept_id|gender|salary|
+------+--------+---------------+-----------+-----------+------+------+
|     1|   Smith|             -1|       2018|         10|     M|  3000|
|     2|    Rose|              1|       2010|         20|     M|  4000|
|     3|Williams|              1|       2010|         10|     M|  1000|
|     4|   Jones|              2|       2005|         10|     F|  2000|
|     5|   Brown|              2|       2010|         40|  NULL|    -1|
+------+--------+---------------+-----------+-----------+------+------+
```



```
data1.join(data2,data1.emp_dept_id==data2.dept_id,"leftanti").show()
```

[15] ✓ 0.4s

```
+------+-----+---------------+-----------+-----------+------+------+
|emp_id| name|superior_emp_id|year_joined|emp_dept_id|gender|salary|
+------+-----+---------------+-----------+-----------+------+------+
|     6|Brown|              2|       2010|         50|  NULL|    -1|
+------+-----+---------------+-----------+-----------+------+------+
```