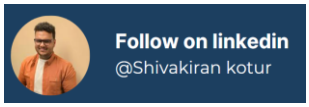
Master PySpark: From Zero to Big Data Hero!!

Windows Function in PySpark Part 2

```
from pyspark.sql import SparkSession
from pyspark.sql.window import Window
import pyspark.sql.functions as F

# Sample data
data = [
    ("Alice", 100),
    ("Bob", 200),
    ("Charlie", 200),
    ("David", 300),
    ("Eve", 400),
    ("Frank", 500),
    ("Grace", 500),
    ("Hank", 600),
    ("Ivy", 700),
    ("Jack", 800)
]
# Create a DataFrame
columns = ["Name", "Score"]
df = spark.createDataFrame(data, columns)
```

```
+-------+-----+
|   Name|Score|
+-------+-----+
|  Alice|  100|
|    Bob|  200|
|Charlie|  200|
|  David|  300|
|    Eve|  400|
|  Frank|  500|
|  Grace|  500|
|   Hank|  600|
|    Ivy|  700|
|   Jack|  800|
+-------+-----+
```



Follow me on LinkedIn – Shivakiran kotur

```python
# Define a window specification
window_spec = Window.orderBy("Score")

# Using rank() to calculate rank
df1 = df.withColumn("Rank", F.rank().over(window_spec))
```

rank:

Name	Score	Rank
Alice	100	1
Bob	200	2
Charlie	200	2
David	300	4
Eve	400	5
Frank	500	6
Grace	500	6
Hank	600	8
Ivy	700	9
Jack	800	10

```python
# Using dense_rank() to calculate dense rank
df2 = df.withColumn("DenseRank", F.dense_rank().over(window_spec))
```

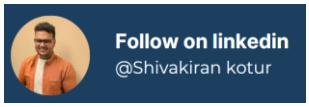
dense_rank:

Name	Score	DenseRank
Alice	100	1
Bob	200	2
Charlie	200	2
David	300	3
Eve	400	4
Frank	500	5
Grace	500	5
Hank	600	6
Ivy	700	7
Jack	800	8

```python
# Using row_number() to calculate row number
df3 = df.withColumn("RowNumber", F.row_number().over(window_spec))
```

Rownumber:

Name	Score	RowNumber
Alice	100	1
Bob	200	2
Charlie	200	3
David	300	4
Eve	400	5
Frank	500	6
Grace	500	7
Hank	600	8
Ivy	700	9
Jack	800	10



Follow me on LinkedIn – Shivakiran kotur

```
# Using lead() to calculate the difference with the next row
df4 = df.withColumn("ScoreDifferenceWithNext",
F.lead("Score").over(window_spec) - df["Score"])
```

lead:

```
+-------+-----+----------------------+
|   Name|Score|ScoreDifferenceWithNext|
+-------+-----+----------------------+
|  Alice|  100|                   100|
|    Bob|  200|                     0|
|Charlie|  200|                   100|
|  David|  300|                   100|
|    Eve|  400|                   100|
|  Frank|  500|                     0|
|  Grace|  500|                   100|
|   Hank|  600|                   100|
|    Ivy|  700|                   100|
|   Jack|  800|                  null|
+-------+-----+----------------------+
```

```
# Using lag() to calculate the difference with the previous row
df5 = df.withColumn("ScoreDifferenceWithPrevious", df["Score"] -
F.lag("Score").over(window_spec))
```

lag:

```
+-------+-----+--------------------------+
|   Name|Score|ScoreDifferenceWithPrevious|
+-------+-----+--------------------------+
|  Alice|  100|                      null|
|    Bob|  200|                       100|
|Charlie|  200|                         0|
|  David|  300|                       100|
|    Eve|  400|                       100|
|  Frank|  500|                       100|
|  Grace|  500|                         0|
|   Hank|  600|                       100|
|    Ivy|  700|                       100|
|   Jack|  800|                       100|
+-------+-----+--------------------------+
```