Master PySpark: From Zero to Big Data Hero!!

Key Notes on when and otherwise

The when and otherwise functions in PySpark provide a way to create conditional expressions within a DataFrame, allowing you to specify different values for new or existing columns based on specific conditions.

when: The when function in PySpark is used to define a condition. If the condition is met, it returns the specified value. You can chain multiple when conditions to handle various cases.

otherwise: The otherwise function specifies a default value to return if none of the conditions in the when statements are met.

```
from pyspark.sql.functions import when

# Syntax to add a new column based on a condition
df = df.withColumn("new_column_name", when(condition1,
value1).when(condition2, value2).otherwise(default_value))
```

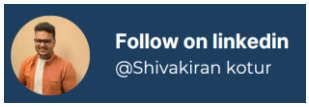
Example

Let's create a dataset and apply when and otherwise conditions.

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import when
from pyspark.sql.types import StructType, StructField, IntegerType,
StringType

# Initialize Spark session
spark =
SparkSession.builder.appName("WhenOtherwiseExample").getOrCreate()

# Define the schema for the dataset
schema = StructType([
    StructField("name", StringType(), True),
    StructField("age", IntegerType(), True),
    StructField("salary", IntegerType(), True)
])
```



```
# Create a sample dataset
data = [
    ("Alice", 25, 3000),
    ("Bob", 35, 4000),
    ("Charlie", 40, 5000),
    ("David", 28, 4500),
    ("Eve", 32, 3500)
]

# Create DataFrame
df = spark.createDataFrame(data, schema)

df.show()
```

```
+-------+---+------+
|   name|age|salary|
+-------+---+------+
|  Alice| 25|  3000|
|    Bob| 35|  4000|
|Charlie| 40|  5000|
|  David| 28|  4500|
|    Eve| 32|  3500|
+-------+---+------+
```

```
# Apply 'when' and 'otherwise' to add new columns based on
conditions
df = (
    df.withColumn("status", when(df.age < 30,
"Young").otherwise("Adult"))
      .withColumn("income_bracket", when(df.salary < 4000, "Low")
                                    .when((df.salary >= 4000) &
(df.salary <= 4500), "Medium")
                                    .otherwise("High"))
)

# Show the result
df.show()
```

```
+-------+---+------+------+--------------+
|   name|age|salary|status|income_bracket|
+-------+---+------+------+--------------+
|  Alice| 25|  3000| Young|           Low|
|    Bob| 35|  4000| Adult|        Medium|
|Charlie| 40|  5000| Adult|          High|
|  David| 28|  4500| Young|        Medium|
|    Eve| 32|  3500| Adult|           Low|
+-------+---+------+------+--------------+
```

Explanation

1. **"status" column**: Assigns "Young" if age < 30, otherwise "Adult".
2. **"income_bracket" column**:
    - Assigns "Low" if salary < 4000.
    - Assigns "Medium" if salary is between 4000 and 4500.
    - Assigns "High" for any other salary values.

This approach allows for flexible handling of multiple conditions in PySpark DataFrames using when and otherwise.