Create a DataFrame in PySpark by reading data from a CSV file and explore its structure and contents.

In [1]:
```
sc
```

Out[1]: **SparkContext**

Spark UI

Version	`v4.0.0`
Master	`local[*]`
AppName	`PySparkShell`

In [2]:
```
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, avg, round, max

# Initialize Spark Session
spark = SparkSession.builder.appName("StudentsDataFrameExample").getOrCreate()
```

In [4]:
```
# Step 1: Read CSV file into DataFrame
df = spark.read.csv("students.csv", header=True, inferSchema=True)
```

In [5]:
```
# Step 2: Explore dataset
print("=== First 10 rows ===")
df.show(10)
```

```
=== First 10 rows ===
+---+-------+---+------+----+-------+-------+
| id|   name|age|gender|math|science|english|
+---+-------+---+------+----+-------+-------+
|  1|  Alice| 20|     F|  66|     92|     44|
|  2|    Bob| 20|     M|  82|     52|     77|
|  3|Charlie| 22|     F|  43|     57|     76|
|  4|  David| 19|     M|  95|     69|     46|
|  5|    Eva| 19|     F|  62|     44|     96|
|  6|  Frank| 22|     F|  70|     78|     94|
|  7|  Grace| 24|     F|  67|     66|     93|
|  8|  Henry| 21|     F|  53|     82|     60|
|  9|    Ivy| 19|     M|  64|     52|     46|
| 10|   Jack| 19|     F|  44|     59|     60|
+---+-------+---+------+----+-------+-------+
only showing top 10 rows
```

In [6]:
```
print("=== Schema ===")
df.printSchema()
```

```
=== Schema ===
root
 |-- id: integer (nullable = true)
 |-- name: string (nullable = true)
 |-- age: integer (nullable = true)
 |-- gender: string (nullable = true)
 |-- math: integer (nullable = true)
 |-- science: integer (nullable = true)
 |-- english: integer (nullable = true)
```

In [7]:
```
print("=== Datatypes ===")
print(df.dtypes)
```

```
=== Datatypes ===
[('id', 'int'), ('name', 'string'), ('age', 'int'), ('gender', 'string'), ('math', 'int'), ('science', 'int'), ('english', 'in
t')]
```

In [8]:
```
print("=== Summary statistics ===")
df.describe().show()
```

```
=== Summary statistics ===
+-------+------------------+-----+------------------+------+------------------+------------------+------------------+
|summary|                id| name|               age|gender|              math|           science|           english|
+-------+------------------+-----+------------------+------+------------------+------------------+------------------+
|  count|                50|   50|                50|    50|                50|                50|                50|
|   mean|              25.5| NULL|              21.5|  NULL|             68.94|             70.16|             69.36|
| stddev|14.577379737113251| NULL|2.2337851101588404|  NULL|17.609610085034216|14.636214521186957|18.74507826560544|
|    min|                 1|Aaron|                18|     F|                40|                44|                42|
|    max|                50| Zoey|                25|     M|               100|                99|               100|
+-------+------------------+-----+------------------+------+------------------+------------------+------------------+
```

In [9]:
```
print("Total rows:", df.count())
print("Columns:", df.columns)
```

```
Total rows: 50
Columns: ['id', 'name', 'age', 'gender', 'math', 'science', 'english']
```

In [10]:
```
# Step 3: Select specific columns
print("\n=== Select name, age, and math columns ===")
df.select("name", "age", "math").show(10)
```

```
=== Select name, age, and math columns ===
+-------+---+----+
|   name|age|math|
+-------+---+----+
|  Alice| 20|  66|
|    Bob| 20|  82|
|Charlie| 22|  43|
|  David| 19|  95|
|    Eva| 19|  62|
|  Frank| 22|  70|
|  Grace| 24|  67|
|  Henry| 21|  53|
|    Ivy| 19|  64|
|   Jack| 19|  44|
+-------+---+----+
only showing top 10 rows
```

In [11]:
```
# Step 4: Filter students (age >= 21 and math >= 70)
print("\n=== Students with age >= 21 and math >= 70 ===")
```

```
df.filter((col("age") >= 21) & (col("math") >= 70)).show(10)
```

```
=== Students with age >= 21 and math >= 70 ===
+---+-------+---+------+----+-------+-------+
| id|   name|age|gender|math|science|english|
+---+-------+---+------+----+-------+-------+
|  6|  Frank| 22|     F|  70|     78|     94|
| 11|  Kathy| 25|     M|  85|     71|     89|
| 12|    Leo| 24|     M|  97|     84|     83|
| 14| Nathan| 23|     F|  71|     66|     60|
| 22| Victor| 22|     M|  96|     75|     56|
| 25|   Yara| 21|     F| 100|     62|     54|
| 27|  Aaron| 25|     F|  81|     99|     44|
| 30|  Diana| 21|     M|  78|     89|     45|
| 35|    Ian| 21|     F|  72|     75|     70|
| 36|Jasmine| 21|     F|  90|     58|     71|
+---+-------+---+------+----+-------+-------+
only showing top 10 rows
```

In [12]:
```
# Step 5: Add a new column: average marks
df_with_avg = df.withColumn("average", round((col("math") + col("science") + col("english")) / 3, 2))
print("\n=== Dataset with new column 'average' ===")
df_with_avg.show(10)
```

```
=== Dataset with new column 'average' ===
+---+-------+---+------+----+-------+-------+-------+
| id|   name|age|gender|math|science|english|average|
+---+-------+---+------+----+-------+-------+-------+
|  1|  Alice| 20|     F|  66|     92|     44|  67.33|
|  2|    Bob| 20|     M|  82|     52|     77|  70.33|
|  3|Charlie| 22|     F|  43|     57|     76|  58.67|
|  4|  David| 19|     M|  95|     69|     46|   70.0|
|  5|    Eva| 19|     F|  62|     44|     96|  67.33|
|  6|  Frank| 22|     F|  70|     78|     94|  80.67|
|  7|  Grace| 24|     F|  67|     66|     93|  75.33|
|  8|  Henry| 21|     F|  53|     82|     60|   65.0|
|  9|    Ivy| 19|     M|  64|     52|     46|   54.0|
| 10|   Jack| 19|     F|  44|     59|     60|  54.33|
+---+-------+---+------+----+-------+-------+-------+
only showing top 10 rows
```

In [13]:
```
# Step 6: Filter students with average >= 75 and sort descending
print("\n=== Students with average >= 75 (sorted) ===")
df_with_avg.filter(col("average") >= 75).orderBy(col("average").desc()).show(10)
```

```
=== Students with average >= 75 (sorted) ===
+---+------+---+------+----+-------+-------+-------+
| id|  name|age|gender|math|science|english|average|
+---+------+---+------+----+-------+-------+-------+
| 12|   Leo| 24|     M|  97|     84|     83|   88.0|
| 15|Olivia| 18|     M|  87|     90|     87|   88.0|
| 44|  Rita| 24|     M|  90|     82|     88|  86.67|
| 11| Kathy| 25|     M|  85|     71|     89|  81.67|
| 33|George| 22|     M|  66|     95|     84|  81.67|
|  6| Frank| 22|     F|  70|     78|     94|  80.67|
| 41| Oscar| 20|     M|  87|     72|     81|   80.0|
| 21|   Uma| 19|     F|  89|     70|     76|  78.33|
| 37|  Kyle| 21|     M|  57|     86|     92|  78.33|
| 39|  Matt| 25|     M|  64|     71|    100|  78.33|
+---+------+---+------+----+-------+-------+-------+
only showing top 10 rows
```

In [14]:
```
# Step 7: Group by gender and calculate average marks
print("\n=== Average marks by gender ===")
df_with_avg.groupBy("gender").agg(
    round(avg("math"), 2).alias("avg_math"),
    round(avg("science"), 2).alias("avg_science"),
    round(avg("english"), 2).alias("avg_english"),
    round(avg("average"), 2).alias("overall_avg")
).show()
```

```
=== Average marks by gender ===
+------+--------+-----------+-----------+-----------+
|gender|avg_math|avg_science|avg_english|overall_avg|
+------+--------+-----------+-----------+-----------+
|     F|   63.86|      68.55|      70.55|      67.66|
|     M|   75.95|      72.38|      67.71|      72.02|
+------+--------+-----------+-----------+-----------+
```

In [17]:
```
# Stop Spark session
```

```
# spark.stop()
```