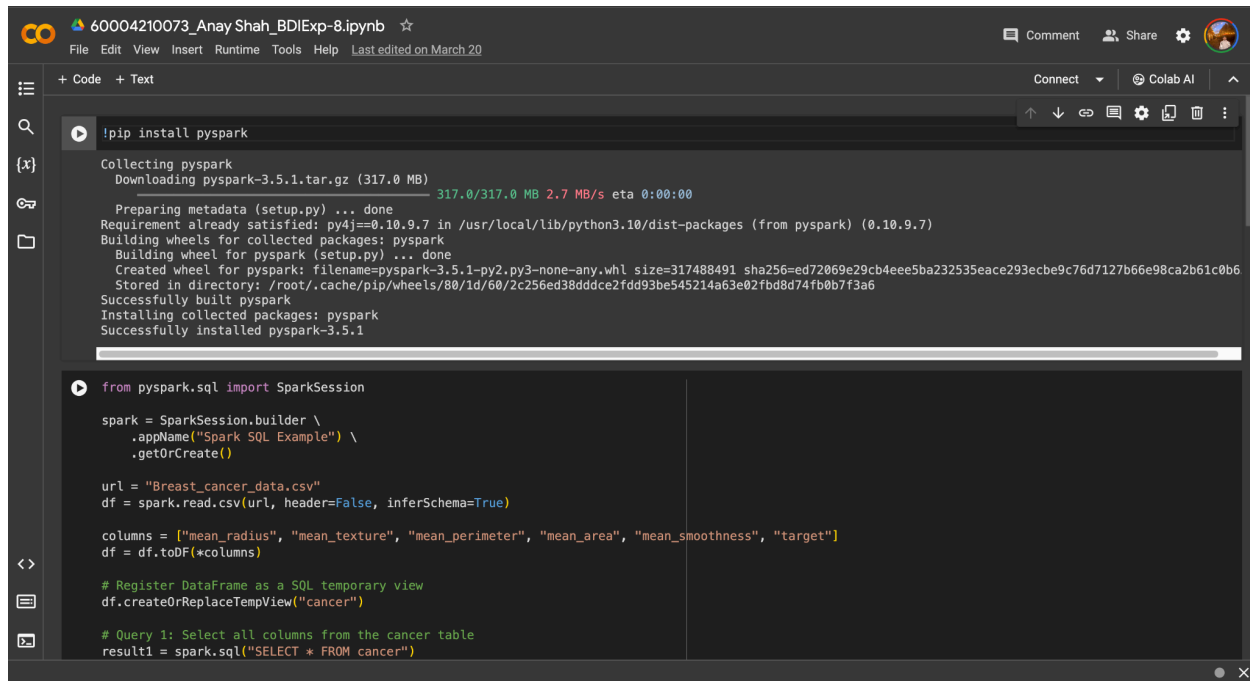


## Output:

Colab Link:

<https://colab.research.google.com/drive/1JUoC3sojDwkXAEMss5KSHLO6-lj9p7NF?usp=sharing>



```
!pip install pyspark

Collecting pyspark
  Downloading pyspark-3.5.1.tar.gz (317.0 MB)
    317.0/317.0 MB 2.7 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: py4j==0.10.9.7 in /usr/local/lib/python3.10/dist-packages (from pyspark) (0.10.9.7)
Building wheels for collected packages: pyspark
  Building wheel for pyspark (setup.py) ... done
  Created wheel for pyspark: filename=pyspark-3.5.1-py2.py3-none-any.whl size=317488491 sha256=ed72069e29cb4eee5ba232535eace293ecbe9c76d7127b66e98ca2b61c0b6
  Stored in directory: /root/.cache/pip/wheels/80/1d/60/2c256ed38ddce2fdd93be545214a63e02fbd8d74fb0b7f3a6
Successfully built pyspark
Installing collected packages: pyspark
Successfully installed pyspark-3.5.1

from pyspark.sql import SparkSession

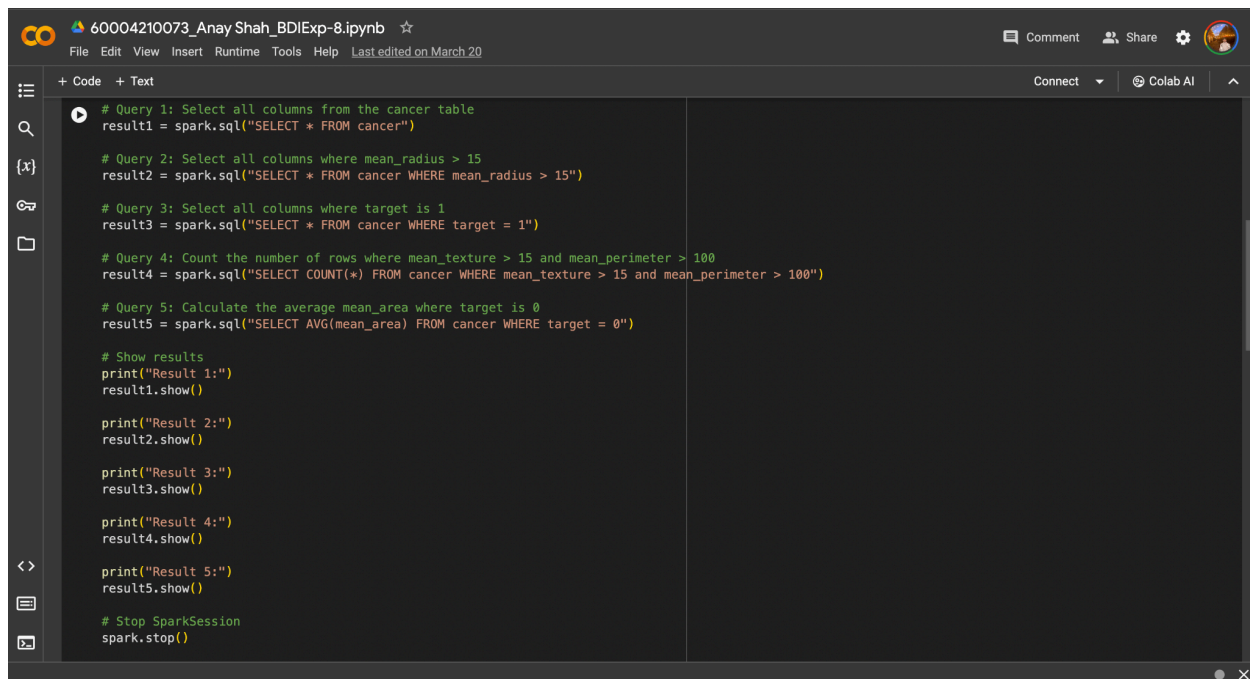
spark = SparkSession.builder \
    .appName("Spark SQL Example") \
    .getOrCreate()

url = "Breast_cancer_data.csv"
df = spark.read.csv(url, header=False, inferSchema=True)

columns = ["mean_radius", "mean_texture", "mean_perimeter", "mean_area", "mean_smoothness", "target"]
df = df.toDF(*columns)

# Register DataFrame as a SQL temporary view
df.createOrReplaceTempView("cancer")

# Query 1: Select all columns from the cancer table
result1 = spark.sql("SELECT * FROM cancer")
```



```
# Query 1: Select all columns from the cancer table
result1 = spark.sql("SELECT * FROM cancer")

# Query 2: Select all columns where mean_radius > 15
result2 = spark.sql("SELECT * FROM cancer WHERE mean_radius > 15")

# Query 3: Select all columns where target is 1
result3 = spark.sql("SELECT * FROM cancer WHERE target = 1")

# Query 4: Count the number of rows where mean_texture > 15 and mean_perimeter > 100
result4 = spark.sql("SELECT COUNT(*) FROM cancer WHERE mean_texture > 15 and mean_perimeter > 100")

# Query 5: Calculate the average mean_area where target is 0
result5 = spark.sql("SELECT AVG(mean_area) FROM cancer WHERE target = 0")

# Show results
print("Result 1:")
result1.show()

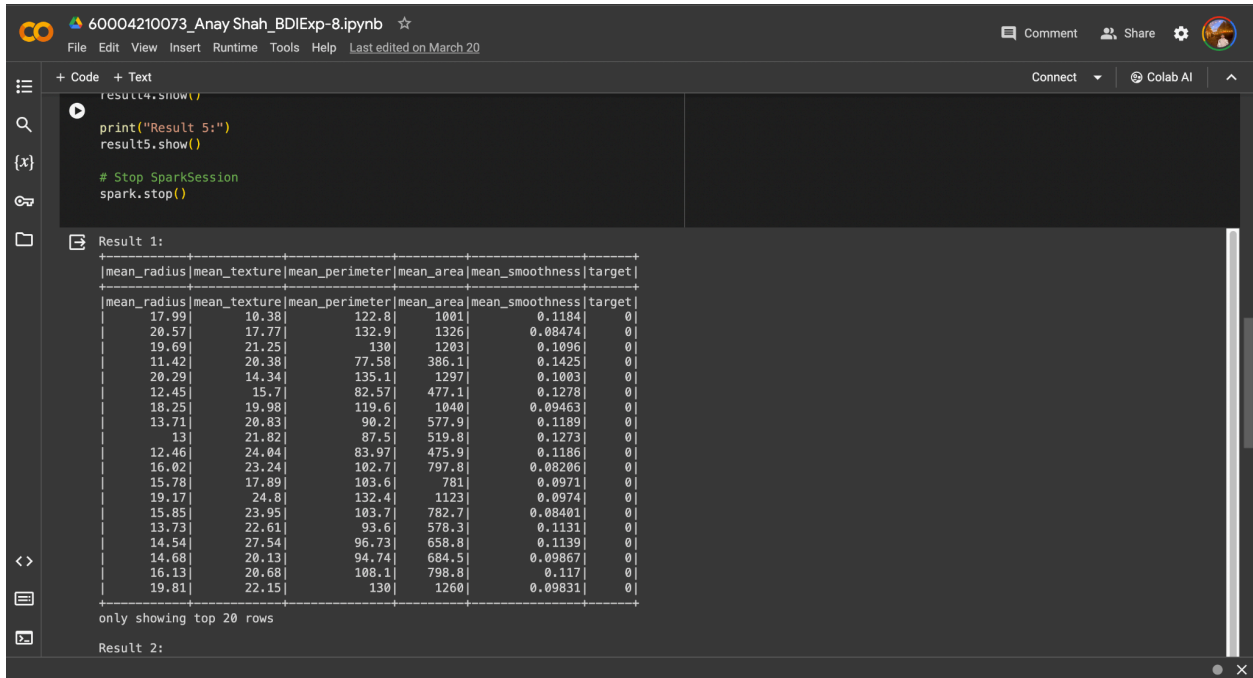
print("Result 2:")
result2.show()

print("Result 3:")
result3.show()

print("Result 4:")
result4.show()

print("Result 5:")
result5.show()

# Stop SparkSession
spark.stop()
```



The screenshot shows a Jupyter Notebook interface with a code cell and its output. The code cell contains the following code:

```
result4.show()

print("Result 5:")
result5.show()

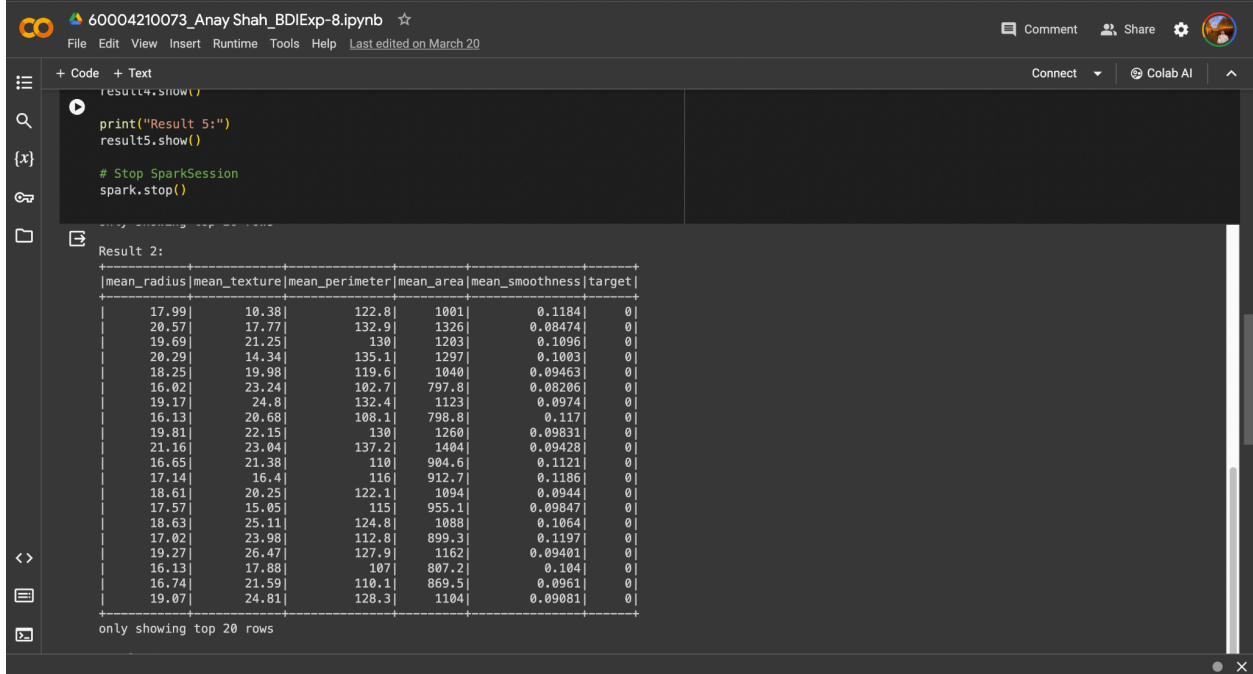
# Stop SparkSession
spark.stop()
```

The output of the code cell is displayed as a table with 6 columns: mean\_radius, mean\_texture, mean\_perimeter, mean\_area, mean\_smoothness, and target. The table shows 20 rows of data, with the first row being the header and the subsequent rows containing numerical values. The output is truncated with "only showing top 20 rows".

mean_radius	mean_texture	mean_perimeter	mean_area	mean_smoothness	target
17.99	10.38	122.8	1001	0.1184	0
20.57	17.77	132.9	1326	0.08474	0
19.69	21.25	130	1203	0.1096	0
11.42	20.38	77.58	386.1	0.1425	0
20.29	14.34	135.1	1297	0.1003	0
12.45	15.7	82.57	477.1	0.1278	0
18.25	19.98	119.6	1040	0.09463	0
13.71	20.83	90.2	577.9	0.1189	0
13	21.82	87.5	519.8	0.1273	0
12.46	24.04	83.97	475.9	0.1186	0
16.02	23.24	102.7	797.8	0.08206	0
15.78	17.89	103.6	781	0.0971	0
19.17	24.8	132.4	1123	0.0974	0
15.85	23.95	103.7	782.7	0.08401	0
13.73	22.61	93.6	578.3	0.1131	0
14.54	27.54	96.73	658.8	0.1139	0
14.68	20.13	94.74	684.5	0.09867	0
16.13	20.68	108.1	798.8	0.117	0
19.81	22.15	130	1260	0.09831	0

only showing top 20 rows

Result 2:



The screenshot shows a Jupyter Notebook interface with a code cell and its output. The code cell contains the following code:

```
result4.show()

print("Result 5:")
result5.show()

# Stop SparkSession
spark.stop()
```

The output of the code cell is displayed as a table with 6 columns: mean\_radius, mean\_texture, mean\_perimeter, mean\_area, mean\_smoothness, and target. The table shows 20 rows of data, with the first row being the header and the subsequent rows containing numerical values. The output is truncated with "only showing top 20 rows".

mean_radius	mean_texture	mean_perimeter	mean_area	mean_smoothness	target
17.99	10.38	122.8	1001	0.1184	0
20.57	17.77	132.9	1326	0.08474	0
19.69	21.25	130	1203	0.1096	0
20.29	14.34	135.1	1297	0.1003	0
18.25	19.98	119.6	1040	0.09463	0
16.02	23.24	102.7	797.8	0.08206	0
19.17	24.8	132.4	1123	0.0974	0
16.13	20.68	108.1	798.8	0.117	0
19.81	22.15	130	1260	0.09831	0
21.16	23.04	137.2	1404	0.09428	0
16.65	21.38	110	904.6	0.1121	0
17.14	16.4	116	912.7	0.1186	0
18.61	20.25	122.1	1094	0.0944	0
17.57	15.05	115	955.1	0.09847	0
18.63	25.11	124.8	1088	0.1064	0
17.02	23.98	112.8	899.3	0.1197	0
19.27	26.47	127.9	1162	0.09401	0
16.13	17.88	107	807.2	0.104	0
16.74	21.59	110.1	869.5	0.0961	0
19.07	24.81	128.3	1104	0.09081	0

only showing top 20 rows

```
60004210073_Anay Shah_BDIExp-8.ipynb ☆
File Edit View Insert Runtime Tools Help Last edited on March 20

+ Code + Text
Connect Colab AI

Result 3:
-----
|mean_radius|mean_texture|mean_perimeter|mean_area|mean_smoothness|target|
|-----|
|13.54|14.36|87.46|566.3|0.09779|1|
|13.08|15.71|85.63|520|0.1075|1|
|9.584|12.44|60.34|273.9|0.1024|1|
|13.03|18.42|82.61|523.8|0.08983|1|
|8.196|16.84|51.71|201.9|0.086|1|
|12.05|14.63|78.04|449.3|0.1031|1|
|13.49|22.3|86.91|561|0.08752|1|
|11.76|21.6|74.72|427.9|0.08637|1|
|13.64|16.34|87.21|571.8|0.07685|1|
|11.94|18.24|75.71|437.6|0.08261|1|
|11.52|18.75|73.34|409|0.09524|1|
|13.05|19.31|82.61|527.2|0.0806|1|
|8.618|11.79|54.34|224.5|0.09752|1|
|10.17|14.88|64.55|311.9|0.1134|1|
|8.598|20.98|54.66|221.8|0.1243|1|
|9.173|13.86|59.2|260.9|0.07721|1|
|9.465|21.01|60.11|269.4|0.1044|1|
|11.31|19.04|71.8|394.1|0.08139|1|
|9.029|17.33|58.79|250.5|0.1066|1|
|12.78|16.49|81.37|502.5|0.09831|1|
-----
only showing top 20 rows

Result 4:
-----
|count(1)|
|148|
-----

Result 5:
-----
```

```
60004210073_Anay Shah_BDIExp-8.ipynb ☆
File Edit View Insert Runtime Tools Help Last edited on March 20

+ Code + Text
Connect Colab AI

[ ] |13.04|18.24|75.71|437.6|0.08261|1|
|11.52|18.75|73.34|409|0.09524|1|
|13.05|19.31|82.61|527.2|0.0806|1|
|8.618|11.79|54.34|224.5|0.09752|1|
|10.17|14.88|64.55|311.9|0.1134|1|
|8.598|20.98|54.66|221.8|0.1243|1|
|9.173|13.86|59.2|260.9|0.07721|1|
|9.465|21.01|60.11|269.4|0.1044|1|
|11.31|19.04|71.8|394.1|0.08139|1|
|9.029|17.33|58.79|250.5|0.1066|1|
|12.78|16.49|81.37|502.5|0.09831|1|
-----
only showing top 20 rows

Result 4:
-----
|count(1)|
|148|
-----

Result 5:
-----
|avg(mean_area)|
|978.3764150943397|
-----

[ ] Start coding or generate with AI.
```