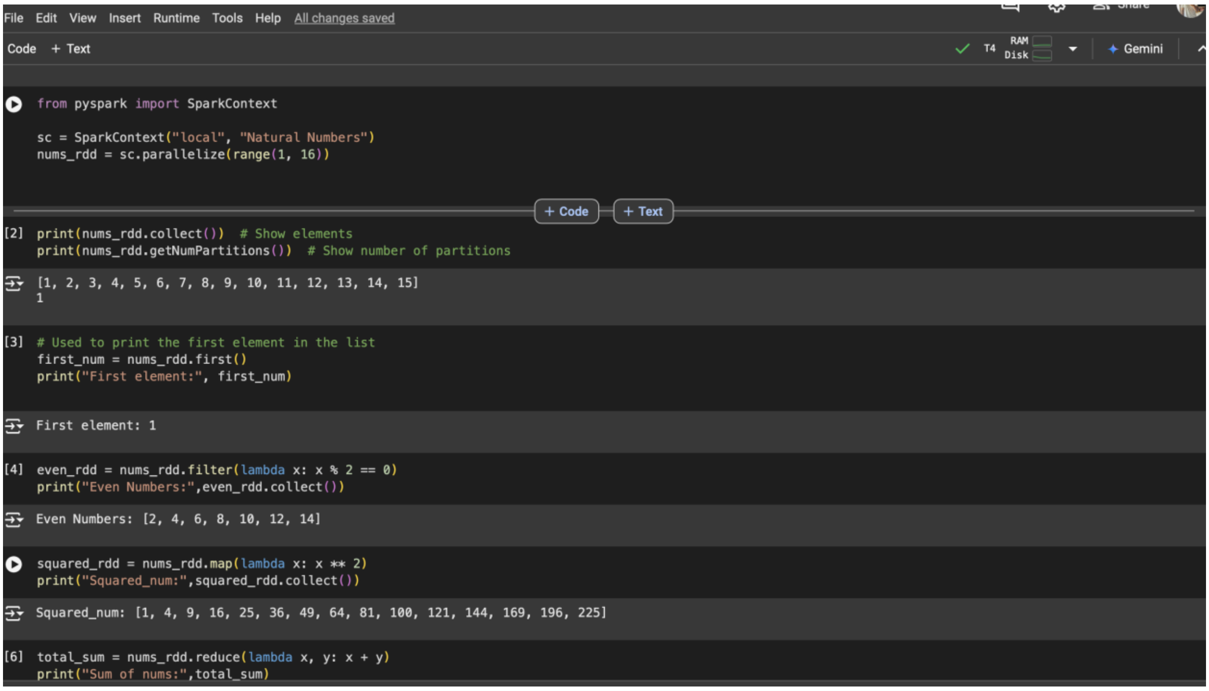


```
from pyspark import SparkContext

sc = SparkContext("local", "Natural Numbers")
nums_rdd = sc.parallelize(range(1, 16))
```

```
[2] print(nums_rdd.collect())  # Show elements
    print(nums_rdd.getNumPartitions())  # Show number of partitions
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]
1
```

```
[3] # Used to print the first element in the list
    first_num = nums_rdd.first()
    print("First element:", first_num)
```

```
First element: 1
```

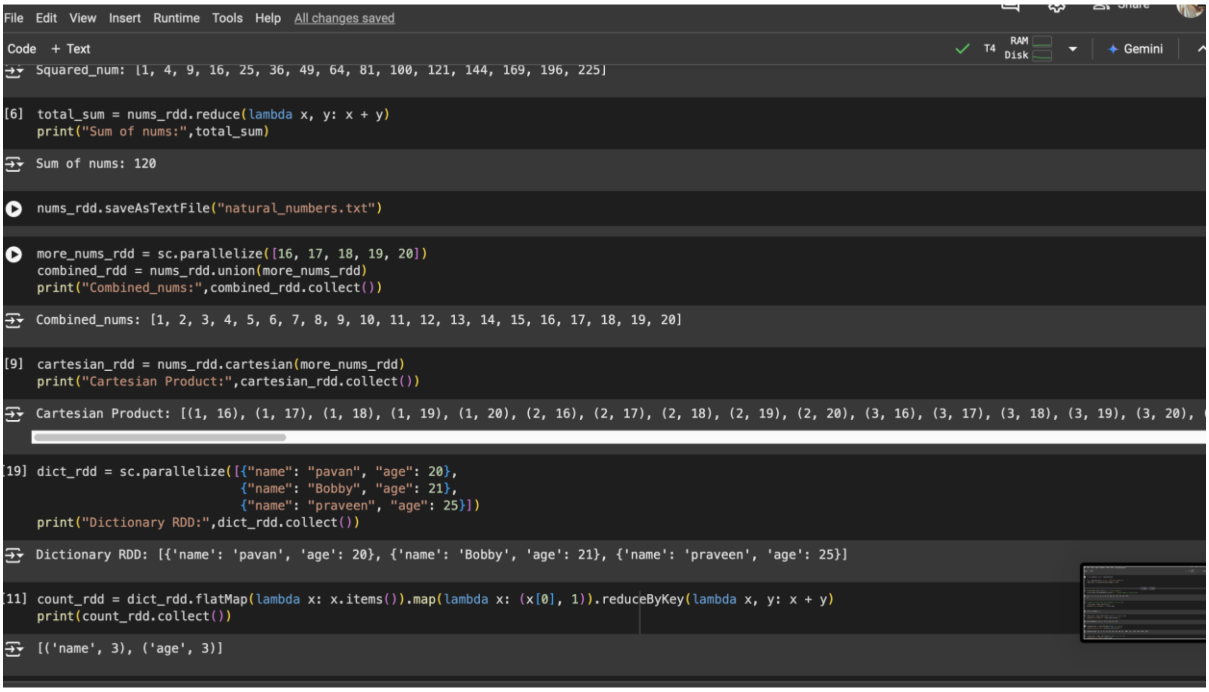
```
[4] even_rdd = nums_rdd.filter(lambda x: x % 2 == 0)
    print("Even Numbers:",even_rdd.collect())
```

```
Even Numbers: [2, 4, 6, 8, 10, 12, 14]
```

```
squared_rdd = nums_rdd.map(lambda x: x ** 2)
print("Squared_num:",squared_rdd.collect())
```

```
Squared_num: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225]
```

```
[6] total_sum = nums_rdd.reduce(lambda x, y: x + y)
    print("Sum of nums:",total_sum)
```



```
Squared_num: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225]
```

```
[6] total_sum = nums_rdd.reduce(lambda x, y: x + y)
    print("Sum of nums:",total_sum)
```

```
Sum of nums: 120
```

```
nums_rdd.saveAsTextFile("natural_numbers.txt")
```

```
more_nums_rdd = sc.parallelize([16, 17, 18, 19, 20])
combined_rdd = nums_rdd.union(more_nums_rdd)
print("Combined_nums:",combined_rdd.collect())
```

```
Combined_nums: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
```

```
[9] cartesian_rdd = nums_rdd.cartesian(more_nums_rdd)
    print("Cartesian Product:",cartesian_rdd.collect())
```

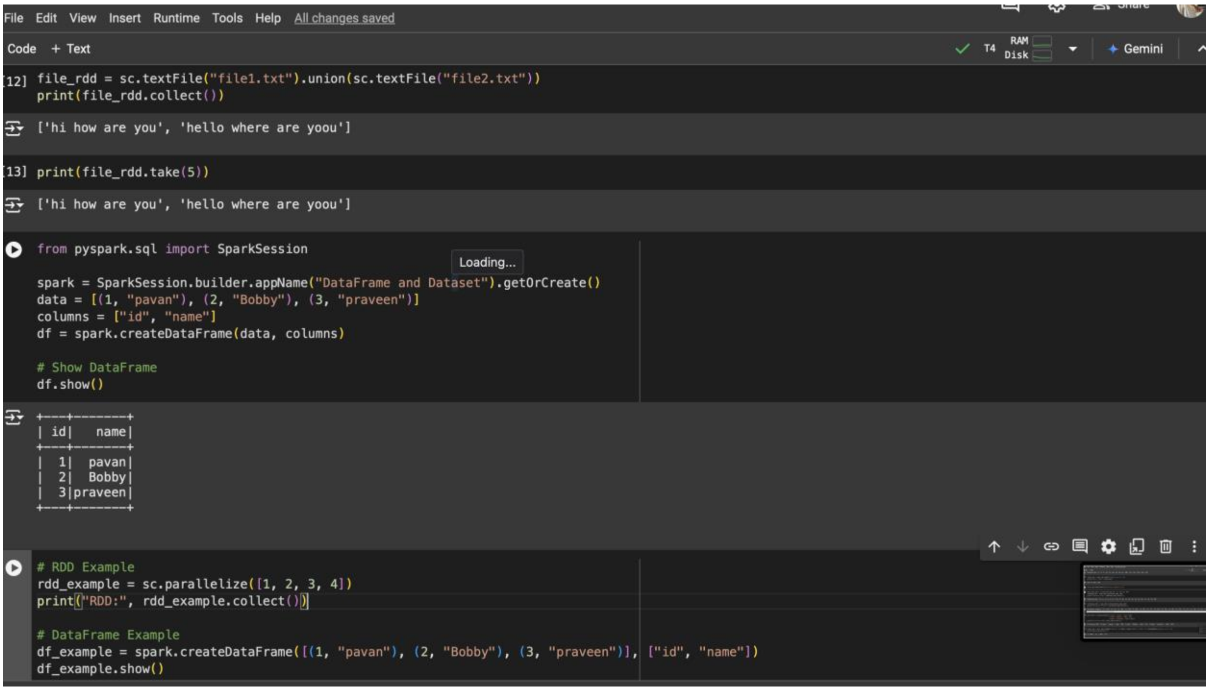
```
Cartesian Product: [(1, 16), (1, 17), (1, 18), (1, 19), (1, 20), (2, 16), (2, 17), (2, 18), (2, 19), (2, 20), (3, 16), (3, 17), (3, 18), (3, 19), (3, 20),
```

```
[19] dict_rdd = sc.parallelize([{"name": "pavan", "age": 20},
                               {"name": "Bobby", "age": 21},
                               {"name": "praveen", "age": 25}])
     print("Dictionary RDD:",dict_rdd.collect())
```

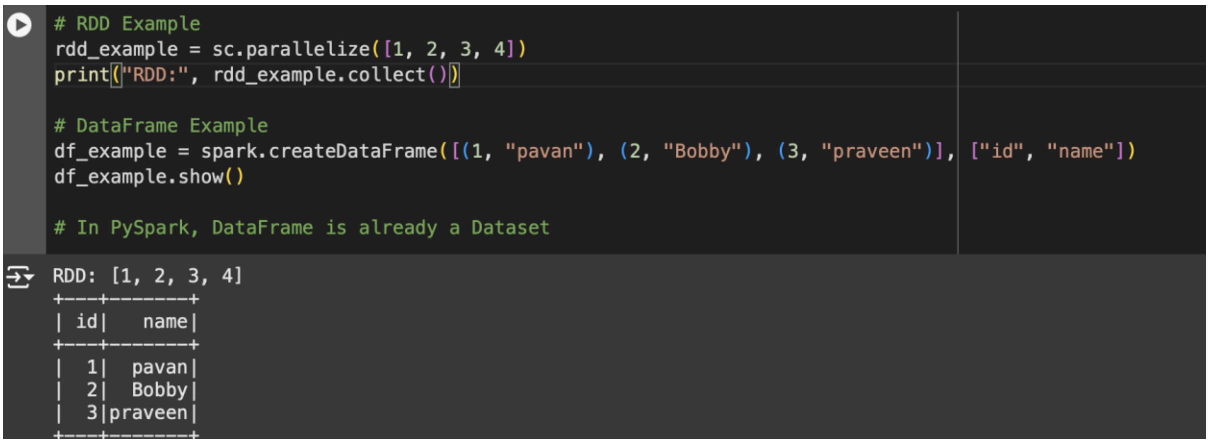
```
Dictionary RDD: [{'name': 'pavan', 'age': 20}, {'name': 'Bobby', 'age': 21}, {'name': 'praveen', 'age': 25}]
```

```
[11] count_rdd = dict_rdd.flatMap(lambda x: x.items()).map(lambda x: (x[0], 1)).reduceByKey(lambda x, y: x + y)
     print(count_rdd.collect())
```

```
[('name', 3), ('age', 3)]
```



```
[12] file_rdd = sc.textFile("file1.txt").union(sc.textFile("file2.txt"))
     print(file_rdd.collect())

     ['hi how are you', 'hello where are yoou']

[13] print(file_rdd.take(5))

     ['hi how are you', 'hello where are yoou']

     from pyspark.sql import SparkSession

     spark = SparkSession.builder.appName("DataFrame and Dataset").getOrCreate()
     data = [(1, "pavan"), (2, "Bobby"), (3, "praveen")]
     columns = ["id", "name"]
     df = spark.createDataFrame(data, columns)

     # Show DataFrame
     df.show()

     +---+-------+
     | id|   name|
     +---+-------+
     |  1|  pavan|
     |  2|  Bobby|
     |  3|praveen|
     +---+-------+

     # RDD Example
     rdd_example = sc.parallelize([1, 2, 3, 4])
     print("RDD:", rdd_example.collect())

     # DataFrame Example
     df_example = spark.createDataFrame([(1, "pavan"), (2, "Bobby"), (3, "praveen")], ["id", "name"])
     df_example.show()
```



```
# RDD Example
rdd_example = sc.parallelize([1, 2, 3, 4])
print("RDD:", rdd_example.collect())

# DataFrame Example
df_example = spark.createDataFrame([(1, "pavan"), (2, "Bobby"), (3, "praveen")], ["id", "name"])
df_example.show()

# In PySpark, DataFrame is already a Dataset

RDD: [1, 2, 3, 4]
+---+-------+
| id|   name|
+---+-------+
|  1|  pavan|
|  2|  Bobby|
|  3|praveen|
+---+-------+
```

My github link:https://github.com/nithin1086/BDA