Experiment No. 10: PySpark - RDDs

(i) What is RDD?

RDD (Resilient Distributed Dataset) is the fundamental data structure of Spark. It is an immutable distributed collection of objects that can be processed in parallel.

(ii) Ways to Create RDD

1. From parallelized collection
2. From external datasets

(iii) Parallelized Collections

```
data = [1, 2, 3, 4, 5]
rdd = spark.sparkContext.parallelize(data)
```

(iv) External Dataset

```
rdd = spark.sparkContext.textFile("hdfs://path/to/data.txt")
```

(v) Existing RDDs

Transform one RDD into another using transformations like `map`, `filter`, etc.

(vi) RDD Operations

```
rdd.count()
rdd.foreach(lambda x: print(x))
rdd.collect()
rdd1.join(rdd2)
rdd.cache()
```

Experiment No. 11: Perform PySpark Transformations

(i) map and flatMap

```
rdd = spark.sparkContext.parallelize(["hello world", "hi"])
rdd_map = rdd.map(lambda x: x.split(" "))
```

```
rdd1 = spark.sparkContext.parallelize([("a", 1), ("b", 2)])
rdd2 = spark.sparkContext.parallelize([("a", 3)])
rdd1.join(rdd2).collect()
```

Experiment No. 12: PySpark SparkConf - Attributes and Applications

(i) What is SparkConf?

SparkConf is the configuration object in PySpark that sets various Spark parameters programmatically.

(ii) Creating Spark Session using SparkConf and CSV operations

```
from pyspark import SparkConf
from pyspark.sql import SparkSession

conf = SparkConf().setAppName("CSVExample").setMaster("local")
spark = SparkSession.builder.config(conf=conf).getOrCreate()

# Reading CSV
df = spark.read.csv("data.csv", header=True, inferSchema=True)

# Moving CSV data to another location
df.write.csv("output/data_copy", header=True)
```