

Map Reduce  
java

HIVE Facebook  
hql : hive query labguage  
  
1, Praveen , TCS, Delhi  
2, Mani, Dell , Pune  
  
hql --> map-reduce-->cluster

Impala Cloudera  
  
hql -->cluster

SPARK --processing engine

MAP  
REDUCE

disk

--m1--disk--rd1--disk

DISK I/O

java , python

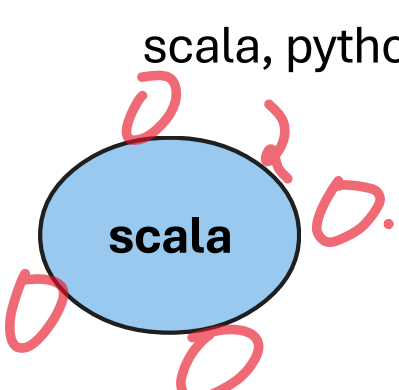
No shell

not interactive , batch  
processing language

SPARK

disk -->m1-->m2--rd1-->disk  
in-memory processing engine

scala, python , java8, sql, R



SCALA : Java , SQL ,  
Python

scala : spark-shell  
python: pyspark

HADOOP

storage: hdfs  
processing:MapReduce  
analysis:hive  
ml:mahout

SPARK

SPARK RDD : unstructured  
data  
SPARK SQL: sql  
SPARK STREAMING  
SPARK ML  
SPARK without YARN :  
Standalone

```
int a ;
a= 10;
```

Scala :spark-shell

```
var a =10;
```

```
val b=78;
```

```
var file_1 = sc.textFile("/user/sample.txt")
```

RDD: building blocks  
Resilient Distributed  
Dataset

Python

```
a = 0;
```

```
file_1 =sc.textFile("/user/sample.txt")
```

```
(file:///C:/Users/Documents)
```

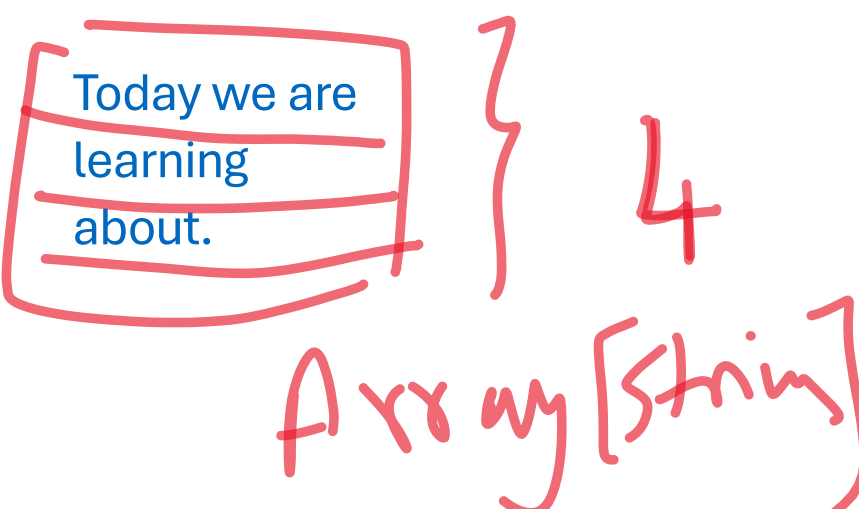
Spark(TT)

DN1

Driver

Spark Context

sc : entry point



RDD are immutable

```
rdd1=sc.textFile("")
rdd2=rdd1.map( lambda x:x.upper())
```



```
var rdd1 = sc.textFile("")
var rdd2=rdd1.map( s=>s.toUpperCase())
```



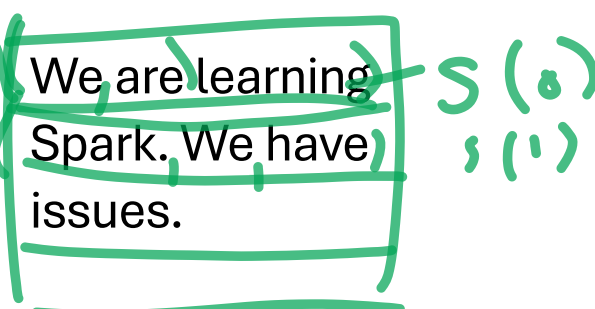
OPERATIONS

LAzy Evaluation

Transformations: map, filter , flatMap , reduceBy

Actions: count(), collect(), take(n), saveAs

```
rdd2=rdd1.map( lambda s:s.split(" "))
rdd3 =rdd1.flatMap(lambda s:s.split(" "))
```



Multiple RDD

subtract  
zip  
union  
intersect  
distinct

RDD1

Boston  
new york  
Delhi  
Shanghai

RDD2

London  
Boston  
Delhi  
San Diego

ques: remove the  
stopwords

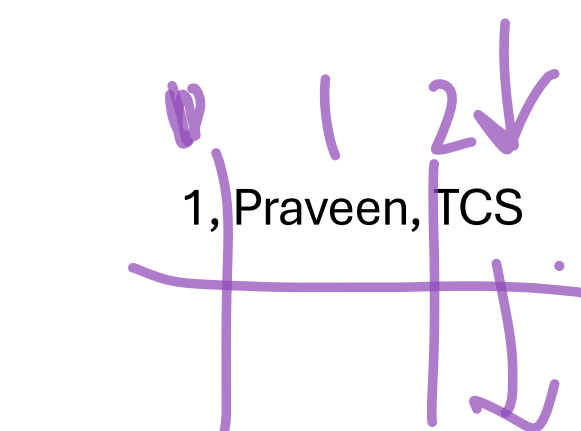
```
rdd3=rdd1.union(rdd2)
```

Mapper Phase

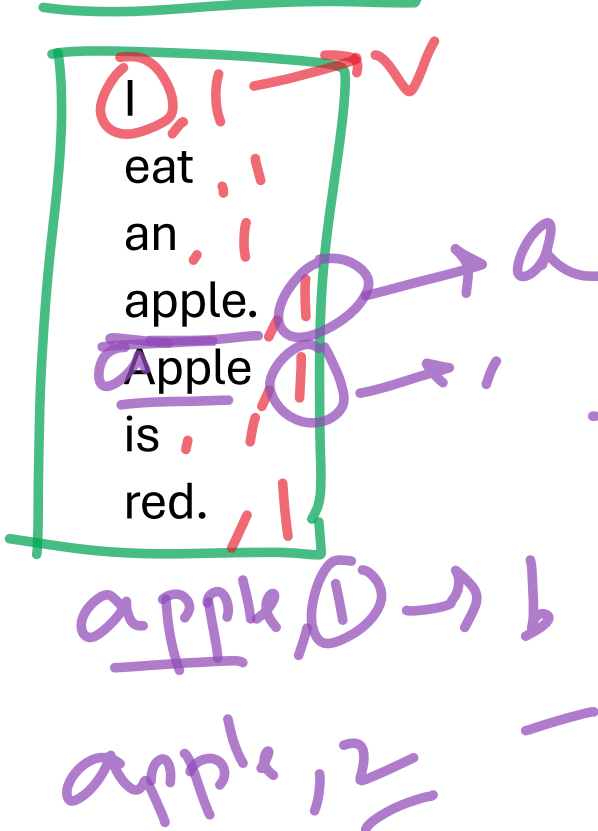
Word Count

```
rdd1=sc.textFile("...")
rdd2=rdd1.flatMap(lambda l:l.split(" "))
rdd3 =rdd2.map( lambda w :(w,1))

rdd4=rdd3.reduceByKey( lambda a,b :(a+b))
rdd4.collect()
```



I eat an apple.  
Apple is red.  
sweet apple.



1. get a file and perform some actions like count, collect, take
2. do basic map and filter transformations on rdd.
3. differenece btwn flatMap and map
4. get the first word from every record
5. multiple rdds, union, zip..
6. remove the english stopwords
7. WordCount