Functions of Hive:

String:

length() – returns the length of string

reverse() – reverse input string

concat() – returns string or bytes after concatenating all strings

space() – returns a string with spaces

upper() | lower() – converts to uppercase | lowercase

Date:

current_date() – current date is returned

current_timestamp() – current sys time & date in format

date_diff() – diff b/w specified dates

last_day()

year()

Quarter()

month()

day()

functions of list :

append() - add ele at end of list
pop() - remove at specified position
sort() - sorts the list
reverse() - reverse the list
remove() - first item with specified value is removed
insert() - adds ele at specified position
index() - returns index of first ele
extend() - add ele of list to end of current list
count() - returns no. of ele with value
copy() - returns a copy of list
clear() - removes all ele from list

Pandas:

read_csv → We can read data in pandas dataframe

2. df.head() → Head returns the first rows, if no i/p is given it will always show above 5 rows.

3. df.tail() →

4. df.shape() → gives total row & column

5. df.size() → no. of rows times no. of col in dataframe

6. df.describe → It will give you count, mean, s.d, & also 5 no summary

7. df.columns → To know the names of all the variables in a dataframe, we can use df.colum

8. df.sample → It allows to ~~create~~ sel values randomly from a series or DF

9. where → use to replace values or column based on cond.

10. Merge() → combines DF based on values in shared col.

11. drop_duplicates()

Numpy :

1. np.array() - used to create an array in Numpy
2. np.arange() - used to create an array with a range of values
3. np.random.rand() - used to create an array with random values b/w 0 & 1.
4. np.max() → used to find maximum value in an array
5. np.mean() → used to find mean value of an array
6. np.repeat() → repeats the ele of an array
7. np.count_nonzero() → Count all the non-zero ele & return their count.
8. np.sort() — sort the array & return
9. np.abs() — return the absolute value of ele inside an array.
10. np.put() — replaces specified elements of an array with given values

Functions in Spark APIs:

1. Data loading & I/P:

→ Spark.read.csv() → reads csv files into DF
→ Spark.read.parquet() →
→ " " .json() →
.text()
.jdbc()

2. Dataframe Operations:

→ Dataframe.select() → sel specific col from a DF
.filter() → filters rows based on a con
.withcolumn() → Adds or replaces a col in any
.groupBy() → group data based on once of more col
.join() → join two DF based on col &exp
.sort() → sorts the DF by one or more col

3. Aggregation & window functions

. Dataframe.agg() → Applies aggregation fun to grouped data

Dataframe.groupBy().agg() → Aggregates db on groups

pyspark.sql.functions.sum() → computes sum of col
" " " .count() → counts no of rows or non-null values in col
" " " .avg()
.rank()

4) Data Transformation & cleaning:-

- DataFrame.withColumnRenamed() → Renames col
  - .drop() → Drops specified col
  - .fillna() → fills missing values in DF with a specified value
  - .na.drop() → Drops rows with missing values from DF
  - sql.functions.when() → Applies conditional transformations to

5) SQL Queries

spark.sql() → executes SQL queries on registed tables or DF

DataFrame.CreateOrReplaceTempView() → registers a DF as a temporary

spark.catalog.listTables() → lists the tables availa

spark.catalog.refreshTable() → refre meladata of a table

6) Output & storage

Dataframe.write.csv() → write a DF to csv

- .parquet()
- .jdbc()
- .json()

5 fun in RDD API.

Transformation fun:

map() - applies a transformation fun to each ele of the RDD & returns a new RDD

filter() - filters ele based on a given cond & returns a new RDD containing the filtered ele

flatMap() - applies a transf fun that returns an iterator for each ele & flattens the results into a new RDD

distinct() - returns a new RDD containing distinct ele from orig RDD

sortBy - sorts the ele of the RDD based on a specified criterion & returns a new RDD.

Action fun:

collect() - returns all ele of RDD as an array to the driver pgm

count() - returns the no. of ele in RDD.

reduce() - Aggregates the ele of the RDD using a specified fun.

take() - returns the first N ele from RDD as an array

foreach() - Applies a fun to each ele of the RDD

Pair RDD fun:

reduceByKey(): Performs a reduction operation on values of a pair RDD based on the key.

groupByKey(): Groups the values of a pair RDD bas on key

sortBykey() - Sorts the ele of a pair RDD based on key.

join() - performs an innerjoin b/w two pair RDD based on key.

cogroup() - Groups the values of multiple pair RDD's sharing the same key.

Persistence functions:

cache() - persists the RDD in memory for faster future accers

persist() - allows specifying diff storage levels for persisting the RDD

unpersist() - removes RDD from mem/disk storage

Spark session obj:

sparksession.

1. builder() — It is used to create a new spark session, this return sparksession.Builder.

2. Read() — returns an instance of DataFrameReader. this is used to read records from CSV

3. Create DataFrame() — This creates a df from a collection & an RDD

4. Create Dataset() — This creates a dataset from the collection, DF & RDD

5. Empty Data frame — creates an empty DF

6. Empty Data set — creates an empty DS

7. sparkcontext() — returns a spark context

8. stop()

~~9. Spark session.~~

9. newsession()

10. table() → load data from table