Date functions:-

(1) CURRENT_DATE :- Returns the current date in the system's time zone.

(2) current_TIMESTAMP :- Returns the current date and time in the system time zone

(3) extract :- Extracts a specific part (year, month, day, hour, etc) from a date or time stamp

(4) Date_ADD :- Adds a specified interval to a date (or) timestamp.

(5) Date_SUB :- Subtract a specific interval from a date (or) timestamp.

(6) Date_Format :- Format a date (or) timestamp into a specific string format ('yyyy_mm_dd')

(7) Date_DIFF :- Calculate the difference between two dates in term of days, months, Year.

8) date_TRUNC :-

Truncates a date (or) time to a specified unit (year, month, day, hour etc).

9) Date_PART :-

Similar to extract, it extracts a specific part from a date (or) time stamp.

10) NOW :- Returns the current date & time.

type Conversion :-

String functions :-

1) CONCAT :- concatenates two (or) more strings together

2) Length (LEN) :- Returns the length (number of characters) of a string

3) UPPER:- Converts a string to uppercase

4) LOWER:- Converts a string to lowercase

5) Substring (or) SUBSTR:-
Extract a portion of a string based on the specified start position and length.

6) REPLACE:- Replace all occurances of a substring with in a string with another substring

7) TRIM:- Removes leading and trailing spaces from a string.

8) char_length :- Returns the length (no of char) of a string

APPROX time :- How much time the processor take to 1 batch

9) Left :- no. of char from the left side of a string

Right :-

ROWS TO COLUMN :- rows to column, column to row

10) ASCII :- Code value of the first char in the string.

Reverse :-

Split :-, Regexp

---

integer function :-

1) ABS :- Return the absolute value of an integer.

ABS(-5)

2) ROUND :- a numeric value to a specified no. of decimal places.

3) CEILING (or) CEIL :- Round a numeric value up to the nearest integer. (4.2)

aggregate

4) FLOOR:- Rounds a numeric value down to the nearest integer (n-1)

5) MOD:- Returns the remainder of the division of the two integers.

6) Power (or) Pow:-
Raises a numeric value to a specified power

7) SORT:- Return square root of a numeric value

8) GREATEST:- return the largest value among the provided arguments.

9) LEAST:- Return the smallest value among the provided arguments.

10) RAND (or) RANDOM:-
generates a random float value between 0 and 1.

-) EXECUTE Function:-

numpy functions:

1. np.array :- create a numpy array from a python list or tuple

2. np.zeros :- create an array filled with zeros of a specified shape.

3. np.ones :- create an array filled with ones of a specified shape

```
arr = np.ones((2,2))
```

4. np.arange :- creates array with regularly spaced values with in a given interval.

```
np.arange(0,10,2).
```

5. np.linspace :- creates an array with specified no. of equally spaced values with in a given interval.

6) np.random.randn = generates random numbers from a standard normal distribution (mean=0, std=1).

7) np.max or np.amax:- Return max value of an array or along a specified axis.

8) np.min or np.amin:- Return the min value of an array or along a specified axis.

9) np.mean:- Cal the arithmetic mean of an array or along a specified axis

10) np.reshape:- reshape an array to specified shape without changing its data

np.sum, np.sin, np.mean, np.unique
np.exp, np.cos, np.std, np.prod.

Pandas functions:

1. pd.read_csv: → Read CSV file
   pd.concat, pd.ods, pd.isnull, pd.melge, pd.groupby
2. df.head → first n rows of a Dataframe
3. df.tail → Return the last n rows
4. df.shape → return the dimensions (row, column)
5. df.info → provide summary
6. df.describe → generates descriptive stadics of numeric columns in df
7. df.groupby → groups the Dataframe by one or more column.
8. df['column_name'] or 'df.colure_name' Access the specific column

a) df.dropna() → remove rows (or columns)

b) df.to_csv() writes the DataFrame to a CSV file.

RDD (Resilient Distributed Dataset):

is a fundamental data structure used for distributed computing.

→ Parallel across a cluster.

→ RDD API operations.

① map:- ~~rdd~~

rdd.map(lambda x: x*2)

Apply function to each element of the RDD to return a new RDD of the result.

② filter Returns a new RDD containing only the elements that satisfy a given predicate.

3) reduce :- Aggregates the elements of the RDD using a specified binary operator

4) flatMap :- similar to map, each input can be mapped to zero or more output items

5) distinct :- Return new RDD with unique elements from the original RDD

6) sortBy :- Return a new RDD sorted by the specified key

7) groupBy key :- Groups the values for each key in RDD and return new RDD of key-value pairs

8) reduceByKey :- similar to reduce. but operates on key-value pairs we apply the reduction function to the value for each key

9) join :- perform an inner join between two RDDs based on their keys

10) collect :- Returns all the elements of the RDD, as an array.

Spark session object :-

- Spark session object is the entry point for working with structured data and executing operations in spark
- include creating data frames, execute SQL queries, and manage config.

create :-

SparkSession builder :-

builder :- is used to create a spark session.builder object which is responsible for configur

and creating the 'SparkSession'.

appName :- sets the name of the Spark application.

Config :- used to set Spark configuration options, you can specify various Config options

get (or) Create :- is used to create a new 'SparkSession' or retrieve an existing one if available.

```
Read data from CSV
df = spark.read.csv(' ', header, inferSchema)
spark.sql(' ') → Execute SQL
df_filtered = df.filter(df["age"]
#transformation on data frame
df_grouped = df.groupBy('gender').agg(
)
```