

algorithm

JINJA → variable

functions in SQL?

What are DATE

- i. CURRENT_DATE : Returns the Current Date
- ii. CURRENT_TIME : Returns the Current Time.
- iii. CURRENT-TIMESTAMP : Returns the Current date time.
- iv. EXTRACT : Extracts a Specific Component from a date time.
- v. DATE-PART : Extracts specific Component from a date or time.

- vi. DATEADD: Adds a specific interval to a date or time.
- vii. DATEDIFF: Calculate the difference b/w two dates or time.
- viii. DATEFORMAT: formats a date or time according to a specific format.
- ix. TO_DATE: Converts a string to a date or time format.
- What are STRING functions in SQL?
- i. CONCAT: Concatenates two or more strings together.
 - ii. SUBSTRING: Extracts a substring from a string.
 - iii. TRIM: Removes leading & trailing spaces from a string.
 - iv. LTRIM: Removes leading spaces from a string.
 - v. RTRIM: Removes trailing spaces from a string.
 - vi. LIKE: Matches a string against a specified pattern using wildcard characters.
 - vii. LENGTH: Returns the length of the string.
 - viii. UPPER: Converts a string to uppercase.
 - ix. LOWER: Converts a string to lowercase.
 - x. REPLACE: Replaces occurrence of substring within a string.

What are Functions in Python?

What are functions in variables.

- i. `Print()`: Output texts or variables.
- ii. `len()`: Returns the length of an iterable.
- iii. `type()`: Returns the datatype of an object.
- iv. `range()`: Generates a sequence of numbers in a specified range.
- v. `input()`: Prompts the user for input from console.
- vi. `abs()`: Returns the absolute value of a number.
- vii. `sorted()`: Returns a new sorted list from iterable.
- viii. `sum()`: Calculates the sum of all elements in a sequence.
- ix. `max()`, `min()`: Returns the maximum or minimum value from a sequence or set of values.
- x. `str()`, `int()`, `float()`: Converts a value to a string, integer, or float data type.

What are the functions in Numpy?

1. Array Creation:

- `array()`: Create a Numpy array from a python array.
- `arrange()`: Create any array with regularly spaced values.
- `zeros()`, `ones()`: Create arrays filled with zeros or ones, respectively.
- `empty()`: Create an array without initializing its elements.

2. Array Manipulation:

- `reshape()`: Reshape an array into a new shape.
- `resize()`: Change the shape & size of an array in-place.
- `concatenate()`: Join arrays along an existing axis.
- `split()`: Split an array into multiple sub-arrays along a specified axis.

3. Mathematical functions:

- `add()`, `subtract()`, `multiply()`, `divide()`: Perform element-wise arithmetic operations on arrays.

sum(), mean(), min(), max(): Compute the sum, mean, minimum or maximum value of an array or along a specified axis.

exp(), log(), sqrt(): Compute the exponential, logarithmic, or square root of all elements.

dot(): Perform matrix multiplication or do product of two arrays.

4. Array Indexing & slicing:-

ndarray [index]: Access elements of an array using indexing.

ndarray [start:stop:step]: Slice an array to extract portion of it.

ndarray [condition]: Use boolean indexing to select elements based on a condition.

5. Statistical functions:-

mean(); median(), std(), var(): Compute the mean, median, standard deviation, variance of an array.

argmin(); argmax(): find the indices of minimum or maximum values in an array.

`histogram()`: compute the histogram of a set of data.

What are functions in Spark API's?

1. Transformation functions (RDD, Dataframe, & Dataset API's):

`map()`: applies a transformation function to each element of an RDD, Dataframe, or Dataset & returns a new RDD, DF & DS.

`filter()`: filters elements based on a specified condition & returns a new RDD, DF & DS, containing the filtered elements.

`select()`: selects specific columns from a DF or DS.
`groupby()`: `join()`; `sort()`: `agg()`:

2. Action functions (RDD, DF & DS API's):

`collect()`: retrieves all the elements.

`count()`: Returns the number of elements.

`first()`, `take()`: Returns the first N elements.

`show()`: Displays the content of a DF in tabular form.

`foreach()`: applies a function to each element.

3. Dataframe & Dataset functions:

`withColumn()`: Adds a new column or replaces an existing column in Df.

`groupBy(), agg()`: Performs group-by & aggregate operation on columns.

`pivot()`: Perform pivot operations on a Df or creating a pivot table.

`explode()`: Expands a column with arrays maps into multiple rows in Df.

`na (Null Handling)`: Provides functions to handle missing or null values in D.F or D.S.

4. SQL functions (DF & D.S API'S)

`select Expr()`: Allows executing SQL-like expression on DF or D.S.

`where(), filter()`; filters row based on a SQL condition.

`groupby(); agg()`,

`join()`,

5. Machine Learning functions (MLlib API):

`fit()`, `transform()`: functions used for model training & transformation in the MLlib API.

`evaluate()`, `Predict()`: functions used for model evaluation & prediction in MLlib API.

What are the functions in RDD API's?

Transformation functions:

`map()`: Applies a transformation function to each element of RDD & returns a new RDD.

`filter()`: filters elements based on a given condition & return a new RDD containing filtered elements.

`flatMap()`: Applies a transformation function that return an iterator for each element & flattens the results into a new RDD.

`distinct()`: Returns a new RDD containing distinct elements from the original RDD.

`sortBy()`: sorts the elements of the RDD based on a specific criterion & returns new RDD.

Union(); intersection(); subtract():
set operations b/w two RDD's

2. Action functions:-

collect(): Returns all the elements of the RDD as an array to drives program.

count(): Returns the number of elements in RDD.

reduce(): Aggregates the elements of RDD using specified function.

take(): Returns the first N elements from RDD as an array.

foreach(): Applies a function to each element of the RDD.

3. Pair RDD functions:-

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

groupByKey(): Groups values of a pair RDD.

sortByKey(): Sorts the elements of a pair RDD.

`join()`

`cogroup()`: Groups the values of multipliers
pairs RDDs sharing same key.

4. Persistence functions:

`cache()`: Persist the RDD in memory for
faster future access.

`persist()`: Allows specifying different storage
levels for persisting RDD.

`unpersist()`: Removes the RDD from memory
disk storage.

5. Input & Output functions:

`textFile()`: Reads a textfile & converts it
into RDD of strings.

`saveAsTextfile()`: writes the contents of an
RDD to a text file.

In what are the functions in Pandas?

1. Data Reading & Writing:

`read_csv()`, `read_excel()`, `read_sql()`: Read data from various file formats or databases.

`to_csv()`, `to_excel()`, `to_sql()`: Write data from a data frame to various file formats.

2. Data Inspection & Exploration:-

`head()`, `tail()`: Display the first, or last n rows of a data frame.

`info()`: Provide a summary of the data frame including columns names, data types & missing values.

`describe()`: Compute descriptive statistics of numerical columns in data frame.

`shape()`: Return the dimensions (rows, columns) of the data frame.

`columns()`: Returns the column labels of the data frame.

3. Data Selection & Filtering:

`loc[]; iloc[];` Select rows & columns using label-based or integer-based indexing respectively.

`[];` Select columns based on a specific boolean condition.

`query();` Select rows based on specific boolean condition.

`isin()` Check if values are present in column.

4. Data Manipulation & Transformation:-

`drop()`: Remove specified row or columns from the data frame.

`sort_value()`: Sort the data frame by one or more columns.

`groupby()`: Group the data frame by one or more columns for aggregation.

5. Data Aggregation & Summarization:

`sum(); mean(); min(); max();` Compute the sum, maximum, minimum value of columns.

`mean(); max(); min();` Mean, maximum, minimum value in each column.

`count();` Count non-null values in each column.

value_counts(): Count the occurrences of unique values in column.

agg(); Perform multiple aggregation functions on columns.