Dataframe.aggregate() function is used to apply some aggregation across one or more column. Aggregate using callable, string, dict, or list of string/callables. Most frequently used aggregations are:

sum: Return the sum of the values for the requested axis
min: Return the minimum of the values for the requested axis
max: Return the maximum of the values for the requested axis

Syntax: DataFrame.aggregate(func, axis=0, *args, **kwargs)
Parameters:
func : callable, string, dictionary, or list of string/callables. Function to use for aggregating the data. If a function, must either work when passed a DataFrame or when passed to DataFrame.apply. For a DataFrame, can pass a dict, if the keys are DataFrame column names.
axis : (default 0) {0 or 'index', 1 or 'columns'} 0 or 'index': apply function to each column. 1 or 'columns': apply function to each row.
Returns: Aggregated DataFrame

Example :

Aggregate 'sum' and 'min' function across all the columns in data frame.

```
# importing pandas package
import pandas as pd

# making data frame from csv file
df = pd.read_csv("nba.csv")

# printing the first 10 rows of the dataframe
df[:10]
```

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0
5	Amir Johnson	Boston Celtics	90.0	PF	29.0	6-9	240.0	NaN	12000000.0
6	Jordan Mickey	Boston Celtics	55.0	PF	21.0	6-8	235.0	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41.0	C	25.0	7-0	238.0	Gonzaga	2165160.0
8	Terry Rozier	Boston Celtics	12.0	PG	22.0	6-2	190.0	Louisville	1824360.0
9	Marcus Smart	Boston Celtics	36.0	PG	22.0	6-4	220.0	Oklahoma State	3431040.0

Aggregation works with only numeric type columns.

```
# Applying aggregation across all the columns
# sum and min will be found for each
# numeric type column in df dataframe
```

df.aggregate(['sum', 'min'])

Output:
For each column which are having numeric values, minimum and sum of all values has been found. For dataframe df , we have four such columns Number, Age, Weight, Salary.

	Number	Age	Weight	Salary
sum	8079.0	12311.0	101236.0	2.159837e+09
min	0.0	19.0	161.0	3.088800e+04