Q Search the docs ...

Spark SQL pyspark.sql.SparkSession pyspark.sql.Catalog pyspark.sql.DataFrame pyspark.sql.Column pyspark.sql.Row pyspark.sql.GroupedData pyspark.sql.PandasCogro pyspark.sql.DataFrameNa pyspark.sql.DataFrameSt pyspark.sql.Window pyspark.sql.SparkSession pyspark.sql.SparkSession <u>pyspark.sql.SparkSessior</u> pyspark.sql.SparkSession <u>pyspark.sql.SparkSession</u> pyspark.sql.SparkSession pyspark.sql.conf.Runtime pyspark.sql.DataFrameRe pyspark.sql.DataFrameRe pyspark.sql.DataFrameRe pyspark.sql.DataFrameRe pyspark.sql.DataFrameRe <u>pyspark.sql.DataFrameRe</u> pyspark.sql.DataFrameRe <u>pyspark.sql.DataFrameRe</u> <u>pyspark.sql.DataFrameRe</u> pyspark.sql.DataFrameRe pyspark.sql.DataFrameRe pyspark.sql.DataFrameWi

pyspark.sql.DataFrame.dropna

DataFrame.dropna(how='any', thresh=None, subset=None)

[source]

Returns a new <u>DataFrame</u> omitting rows with null values. <u>DataFrame.dropna()</u> and <u>DataFrameNaFunctions.drop()</u> are aliases of each other.

New in version 1.3.1.

Parameters: how: str, optional

'any' or 'all'. If 'any', drop a row if it contains any nulls. If 'all', drop a row only if all its values are null.

thresh: int, optional

default None If specified, drop rows that have less than *thresh* non-null values. This overwrites the *how* parameter.

subset : str, tuple or list, optional

optional list of column names to consider.

Examples

<<pyspark.sql.DataFrame.drop_duplicates</pre>

pyspark.sql.DataFrame.dtypes >>

© Copyright .

Created using <u>Sphinx</u> 3.0.4.