Function 1: pandas.Series.add

Series.add(other, level=None, fill_value=None, axis=0)

Return Addition of series and other, element-wise (binary operator add).
Equivalent to series + other, but with support to substitute a fill_value for missing data in either one of the inputs.

Parameters

other: Series or scalar value

fill_value: None or float value, default None (NaN)

Fill existing missing (NaN) values, and any new element needed for successful Series alignment, with this value before computation. If data in both corresponding Series locations is missing the result of filling (at that location) will be missing.

level: int or name

Broadcast across a level, matching Index values on the passed MultiIndex level.

Returns:

Series

The result of the operation.

The input can be either scalar or series

Test Cases

test_add_series()

test_add_series_fill()
test_add_scalar()
test_add_NaN()
test_add_level()
test_add_axis()

Function 2: pandas.Series.mod

Series.mod(other, level=None, fill_value=None, axis=0)

Return Modulo of series and other, element-wise (binary operator mod).
Equivalent to series % other, but with support to substitute a fill_value for missing data in either one of the inputs.

Parameters

other Series or scalar value

fill_value None or float value, default None (NaN)

Fill existing missing (NaN) values, and any new element needed for successful Series alignment, with this value before computation. If data in both corresponding Series locations is missing the result of filling (at that location) will be missing.

level int or name

Broadcast across a level, matching Index values on the passed MultiIndex level.

Returns:

Series

The result of the operation.

test_mod_series()
test_mod_series_fill()
test_mod_scalar()
test_mod_NaN()
test_mod_level()
test_mod_axis()

Function 3:pandas.Series.combine

Series.combine(other, func, fill_value=None)

Combine the Series with a Series or scalar according to func.
Combine the Series and other using func to perform elementwise selection for combined Series. fill_value is assumed when value is missing at some index from one of the two objects being combined.

Parameters

other Series or scalar
The value(s) to be combined with the Series.

func function
Function that takes two scalars as inputs and returns an element.

fill_value scalar, optional
The value to assume when an index is missing from one Series or the other. The default specifies to use the appropriate NaN value for the underlying dtype of the Series.

Returns

Series
The result of combining the Series with the other object.

Test Cases

test_combine_series()
test_combine_series_fill()
test_combine_scalar()
test_combine_NaN()
test_combine_func()

Function 4:pandas.Series.round

Series.round(decimals=0, *args, **kwargs)

Round each value in a Series to the given number of decimals.

Parameters

decimals int, default 0

Number of decimal places to round to. If decimals is negative, it specifies the number of positions to the left of the decimal point.

***args, **kwargs**

Additional arguments and keywords have no effect but might be accepted for compatibility with NumPy.

Returns

Series

Rounded values of the Series.

Test Cases

test_round():

test_round_neg():