

≡ 13. Glossary

Glossary

Below is the summary of all the functions and methods that you learned in this lesson:

Category: Initialization and Utility

Function/Method	Description
<code>pandas.read_csv(relative_path_to_file)</code>	Reads a comma-separated values (csv) file present at <code>relative_path_to_file</code> and loads it as a DataFrame
<code>pandas.DataFrame(data)</code>	Returns a 2-D heterogeneous tabular data. Note: There are other optional arguments as well that you can use to create a dataframe.
<code>pandas.Series(data, index)</code>	Returns 1-D ndarray with axis labels
<code>pandas.Series.shape</code> <code>pandas.DataFrame.shape</code>	Returns a tuple representing the dimensions
<code>pandas.Series.ndim</code> <code>pandas.DataFrame.ndim</code>	Returns the number of the dimensions (rank). It will return 1 in case of a Series
<code>pandas.Series.size</code> <code>pandas.DataFrame.size</code>	Returns the number of elements
<code>pandas.Series.values</code>	Returns the data available in the Series
<code>pandas.Series.index</code>	Returns the indexes available in the Series
<code>pandas.DataFrame.isnull()</code>	Returns a same sized object having True for NaN elements and False otherwise.
<code>pandas.DataFrame.count(axis)</code>	Returns the count of non-NaN values along the given axis. If axis=0, it will count down the dataframe, meaning column-wise count of non-NaN values.
<code>pandas.DataFrame.head([n])</code>	Return the first <i>n</i> rows from the dataframe. By default, n=5.
<code>pandas.DataFrame.tail([n])</code>	Return the last <i>n</i> rows from the dataframe. By default, n=5. Supports negative indexing as well.
<code>pandas.DataFrame.describe()</code>	Generate the descriptive statistics, such as, count, mean, std deviation, min, and max.
<code>pandas.DataFrame.min()</code>	Returns the minimum of the values along the given axis.
<code>pandas.DataFrame.max()</code>	Returns the maximum of the values along the given axis.
<code>pandas.DataFrame.mean()</code>	Returns the mean of the values along the given axis.
<code>pandas.DataFrame.corr()</code>	Compute pairwise correlation of columns, excluding NA/null values.
<code>pandas.DataFrame.rolling(window).rolling(window).mean()</code>	Provide rolling window calculation, such as <code>pandas.DataFrame.rolling(15).mean()</code> for rolling mean over window size of 15.
<code>pandas.DataFrame.loc[label]</code>	Access a group of rows and columns by label(s)
<code>pandas.DataFrame.groupby(mapping_function)</code>	Groups the dataframe using a given mapper function or or by a Series of columns.

Category: Manipulation

Function/Method	Description
<code>pandas.Series.drop(index)</code>	Drops the element positioned at the given index(es)
<code>pandas.DataFrame.drop(labels)</code>	Drop specified labels (entire columns or rows) from the dataframe.
<code>pandas.DataFrame.pop(item)</code>	Return the item and drop it from the frame. If not found, then raise a KeyError.
<code>pandas.DataFrame.insert(location, column, values)</code>	Insert column having given values into DataFrame at specified location.
<code>pandas.DataFrame.rename(dictionary-like)</code>	Rename label(s) (columns or row-indexes) as mentioned in the <code>dictionary-like</code>
<code>pandas.DataFrame.set_index(keys)</code>	Set the DataFrame's row-indexes using existing column-values.
<code>pandas.DataFrame.dropna(axis)</code>	Remove rows (if axis=0) or columns (if axis=1) that contain missing values.
<code>pandas.DataFrame.fillna(value, method, axis)</code>	Replace NaN values with the specified value along the given axis, and using the given method ('backfill', 'bfill', 'pad', 'ffill', None)
<code>pandas.DataFrame.interpolate(method, axis)</code>	Replace the NaN values with the estimated value calculated using the given method along the given axis.

For any other requirement, refer to the complete list of function definitions for [DataFrame](#) and [Series](#) classes.

【udacity2.0】 If you need more courses, please add wechat : udacity6