

# Pandas Basics Cheatsheet 1.0

Import Dataframes	Stack/Unstack	Setting/Resetting Index	Merge Data
<pre># import pandas module: <b>import</b> pandas as pd # making dataframe: df = pd.read_csv("https://bit.ly/2XFqnQz") # it was print the first 5-rows: print(df.head())</pre> <p><b>USING STACK METHOD</b></p> <pre># import pandas module: <b>import</b> pandas as pd # making dataframe: df = pd.read_csv("nba.csv") # reshape the dataframe using stack() method: df_stacked = df.stack() print(df_stacked.head(26))</pre> <p><b>USING UNSTACK METHOD</b></p> <pre># import pandas module: <b>import</b> pandas as pd # making dataframe: df = pd.read_csv("nba.csv") # unstack() method: df_unstacked = df_stacked.unstack() print(df_unstacked.head(10))</pre> <p><b>USING MELT METHOD</b></p> <pre># import pandas module: <b>import</b> pandas as pd # making dataframe df = pd.read_csv("nba.csv") # it takes two columns "Name" and "Team" df_melt = df.melt(id_vars =['Name', 'Team']) print(df_melt.head(10))</pre>	<p><b>Pivot a level of column labels</b></p> <pre>&gt;&gt;&gt; stacked = df5.stack()</pre> <p><b>Pivot a level of index labels</b></p> <pre>&gt;&gt;&gt; stacked.unstack()</pre> <p><b>Melt</b></p> <p><b>Gather columns into rows</b></p> <pre>&gt;&gt;&gt; pd.melt(df2,               id_vars=["Date"],               value_vars=["Type", "Value"],               value_name="Observations")</pre>	<p><b>Set the index</b></p> <pre>&gt;&gt;&gt; df.set_index('Country')</pre> <p><b>Reset the index</b></p> <pre>&gt;&gt;&gt; df4 = df.reset_index()</pre> <p><b>Rename DataFrame</b></p> <pre>&gt;&gt;&gt; df = df.rename(index=str,                     columns={"Country": "cntry",                               "Capital": "cptl",                               "Population": "ppltn"})</pre>	<pre>&gt;&gt;&gt; pd.merge(data1,              data2,              how='left',              on='Col1')</pre> <pre>&gt;&gt;&gt; pd.merge(data1,              data2,              how='right',              on='Col1')</pre> <pre>&gt;&gt;&gt; pd.merge(data1,              data2,              how='inner',              on='Col1')</pre> <pre>&gt;&gt;&gt; pd.merge(data1,              data2,              how='outer',              on='Col1')</pre>
Reshaping Data- Pivot	Iteration	MultiIndexing	Join
<p><b>Spread rows into columns</b></p> <pre>&gt;&gt;&gt; df3= df2.pivot(index='Date',                      columns='Type',                      values='Value')</pre> <p><b>Spread rows into columns</b></p> <pre>&gt;&gt;&gt; df4 = pd.pivot_table(df2,                           values='Value',                           index='Date',                           columns='Type')</pre>	<p><b>(Column-index, Series) pairs</b></p> <pre>&gt;&gt;&gt; df.iteritems()</pre> <p><b>(Row-index, Series) pairs</b></p> <pre>&gt;&gt;&gt; df.iterrows()</pre>	<pre>&gt;&gt;&gt; arrays = [np.array([1,2,3]),                np.array([5,4,3])] &gt;&gt;&gt; df5 = pd.DataFrame(np.random.rand(3, 2), index=arrays) &gt;&gt;&gt; tuples = list(zip(*arrays)) &gt;&gt;&gt; index = pd.MultiIndex.from_tuples(tuples,                                       names=['first', 'second']) &gt;&gt;&gt; df6 = pd.DataFrame(np.random.rand(3, 2), index=index) &gt;&gt;&gt; df2.set_index(["Date", "Type"])</pre>	<pre>&gt;&gt;&gt; data1.join(data2,                how='right')</pre>
	Advanced Indexing	Duplicate Data	Concatenate
	<p><b>Selecting</b></p> <p>Select cols with any vals &gt;1</p> <pre>&gt;&gt;&gt; df3.loc[:,(df3&gt;1).any()]</pre> <p>Select cols with vals &gt; 1</p> <pre>&gt;&gt;&gt; df3.loc[:,(df3&gt;1).all()]</pre> <p>Select cols with NaN</p> <pre>&gt;&gt;&gt; df3.loc[:,df3.isnull().any()]</pre> <p>Select cols without NaN</p> <pre>&gt;&gt;&gt; df3.loc[:,df3.notnull().all()]</pre> <p><b>Indexing With isin</b></p> <p>Find same elements</p> <pre>&gt;&gt;&gt; df[(df.Country.isin(df2.Type))]</pre> <p>Filter on values</p> <pre>&gt;&gt;&gt; df3.filter(items="a","b")</pre> <p>Select specific elements</p> <pre>&gt;&gt;&gt; df.select(lambda x: not x%5)</pre> <p><b>Where</b></p> <p>Subset the data</p> <pre>&gt;&gt;&gt; s.where(s &gt; 0)</pre> <p><b>Query</b></p> <p>Query DataFrame</p> <pre>&gt;&gt;&gt; df6.query('second &gt; first')</pre>	<p>Return unique values</p> <pre>&gt;&gt;&gt; s3.unique()</pre> <p>Check duplicates</p> <pre>&gt;&gt;&gt; df2.duplicated('Type')</pre> <p>Drop duplicates</p> <pre>&gt;&gt;&gt; df2.drop_duplicates('Type', keep='last')</pre> <p>Check index duplicates</p> <pre>&gt;&gt;&gt; df.index.duplicated()</pre>	<p><b>Vertical</b></p> <pre>&gt;&gt;&gt; s.append(s2)</pre> <p><b>Horizontal/Vertical</b></p> <pre>&gt;&gt;&gt; pd.concat([s,s2],axis=1,                keys=['One','Two']) &gt;&gt;&gt; pd.concat([data1,                data2], axis=1, join='inner')</pre>
	Missing Data		
	<p>Drop NaN values</p> <pre>&gt;&gt;&gt; df.dropna()</pre> <p>Fill NaN values with predetermined value</p> <pre>&gt;&gt;&gt; df3.fillna(df3.mean())</pre> <p>Replace values with others</p> <pre>&gt;&gt;&gt; df2.replace("a", "f")</pre>		