# Transforming Data With Pandas: Takeaways



by Dataquest Labs, Inc. - All rights reserved © 2019

## **Syntax**

#### APPLYING FUNCTIONS ELEMENT-WISE

• Apply a function element-wise to a series:

```
df[col_name].apply(function_name)

df[col_name].map(function_name)
```

• Apply a function element-wise to a dataframe:

```
df.applymap(function_name)
```

#### APPLYING FUNCTIONS ALONG AN AXIS

• Apply a function along an axis, column-wise:

```
df.apply(function_name)
```

#### **RESHAPING DATAFRAMES**

• Reshape a dataframe:

```
pd.melt(df, id_vars=[col1, col2], value_vars=[col3, col4])
```

## Concepts

• The Series.apply() and Series.map() methods can be used to apply a function element—wise to a series. The DataFrame.applymap() method can be used to apply a function element—wise to a dataframe.

- The DataFrame.apply() method has different capabilities than the Series.apply() method.

  Instead of applying functions element—wise, the df.apply() method applies functions along an axis, either column—wise or row—wise. When we create a function to use with df.apply() , we set it up to accept a Series, most commonly a column.
- Use the <a href="https://example.com/apply">apply()</a> method when a vectorized function does not exist because a vectorized function can perform an equivalent task faster than the <a href="https://example.com/apply()">apply()</a> method. Sometimes, it may be necessary to reshape a dataframe to use a vectorized method.

### Resources

• Tidy Data



Takeaways by Dataquest Labs, Inc. - All rights reserved © 2019