Applying Operations Over pandas Dataframes

- **Author:** Chris Albon (http://www.chrisalbon.com/), @ChrisAlbon (https://twitter.com/chrisalbon)
- **Date:** -
- **Repo:** Python 3 code snippets for data science (https://github.com/chrisalbon/code_py)
- **Note:**

Import Modules

In [72]:

```
import pandas as pd
import numpy as np
```

Create a dataframe

In [73]:

```
data = {'name': ['Jason', 'Molly', 'Tina', 'Jake', 'Amy'],
        'year': [2012, 2012, 2013, 2014, 2014],
        'reports': [4, 24, 31, 2, 3],
        'coverage': [25, 94, 57, 62, 70]}
df = pd.DataFrame(data, index = ['Cochice', 'Pima', 'Santa Cruz', 'Maricopa', 'Yuma'])
df
```

Out[73]:

	coverage	name	reports	year
Cochice	25	Jason	4	2012
Pima	94	Molly	24	2012
Santa Cruz	57	Tina	31	2013
Maricopa	62	Jake	2	2014
Yuma	70	Amy	3	2014

5 rows × 4 columns

Create a capitalization lambda function

In [74]:

```
capitalizer = lambda x: x.upper()
```

Apply the capitalizer function over the column 'name'

apply() can apply a function along any axis of the dataframe

In [75]:

```
df['name'].apply(capitalizer)
```

Out[75]:

```
Cochice       JASON
Pima          MOLLY
Santa Cruz     TINA
Maricopa       JAKE
Yuma            AMY
Name: name, dtype: object
```

Map the capitalizer lambda function over each element in the series 'name'

map() applies an operation over each element of a series

In [76]:

```
df['name'].map(capitalizer)
```

Out[76]:

```
Cochice       JASON
Pima          MOLLY
Santa Cruz     TINA
Maricopa       JAKE
Yuma            AMY
Name: name, dtype: object
```

Apply a square root function to every single cell in the whole data frame

applymap() applies a function to every single element in the entire dataframe.

In [77]:

```
# Drop the string variable so that applymap() can run
df = df.drop('name', axis=1)

# Return the square root of every cell in the dataframe
df.applymap(np.sqrt)
```

Out[77]:

	coverage	reports	year
Cochice	5.000000	2.000000	44.855323
Pima	9.695360	4.898979	44.855323
Santa Cruz	7.549834	5.567764	44.866469
Maricopa	7.874008	1.414214	44.877611
Yuma	8.366600	1.732051	44.877611

5 rows × 3 columns

Applying A Function Over A Dataframe

Create a function that multiplies all non-strings by 100

In [80]:

```
# create a function called times100
def times100(x):
    # that, if x is a string,
    if type(x) is str:
        # just returns it untouched
        return x
    # but, if not, return it multiplied by 100
    elif x:
        return 100 * x
    # and leave everything else
    else:
        return
```

Apply the times100 over every cell in the dataframe

In [79]:

```
df.applymap(times100)
```

Out[79]:

	coverage	reports	year
Cochice	2500	400	201200
Pima	9400	2400	201200
Santa Cruz	5700	3100	201300
Maricopa	6200	200	201400
Yuma	7000	300	201400

5 rows × 3 columns