Data Quest

https://www.dataquest.io/m/1/python-basics

DataCamp

1. Intro to Python for Data Science
2. Intermediate Python for Data Science
3. Python Data Science Toolbox (Part 1)
4. Python Data Science Toolbox (Part 2)
5. Cleaning Data in Python
6. pandas Foundations
7. Manipulating DataFrames with pandas
8. Merging DataFrames with pandas

Intro to SQL for Data Science Data Types for Data Science

Intro to Python for Data Science

Chapter 4 - NumPy

Slides

https://campus.datacamp.com/courses/intro-to-python-for-data-science/chapter-4-numpy?ex=3

Your First NumPy Array

In [11]:
```
# Create list baseball
baseball = [180, 215, 210, 210, 188, 176, 209, 200]

# Import the numpy package as np
import numpy as np

# Create a numpy array from baseball: np_baseball
np_baseball = np.array(baseball)


# Print out type of np_baseball
print(type(np_baseball))
```

```
<class 'numpy.ndarray'>
```

Baseball players' height

In [12]:
```
# height is available as a regular list
height = [67, 70, 72, 78, 63]

# Import numpy
import numpy as np

# Create a numpy array from height: np_height
np_height = np.array(height)

# Print out np_height
print(np.array(np_height))

# Convert np_height to m: np_height_m
np_height_m = np_height * 0.0254

# Print np_height_m
print(np_height_m)
```

```
[67 70 72 78 63]
[ 1.7018  1.778   1.8288  1.9812  1.6002]
```

Baseball player's BMI

In [13]:
```
# height and weight are available as a regular lists
height = [67, 70, 72, 78, 63]
weight = [120, 190, 220, 250, 200]

# Import numpy
import numpy as np

# Create array from height with correct units: np_height_m
np_height_m = np.array(height) * 0.0254

# Create array from weight with correct units: np_weight_kg
np_weight_kg = np.array(weight) * 0.453592

# Calculate the BMI: bmi
bmi = np_weight_kg / np_height_m ** 2

# Print out bmi
print(bmi)
```

```
[ 18.79444882  27.26185942  29.83703344  28.89008079  35.42801744]
```

Lightweight baseball players

In [14]:
```
# height and weight are available as a regular lists
height = [67, 70, 72, 78, 63]
weight = [120, 190, 220, 250, 200]

# Import numpy
import numpy as np

# Calculate the BMI: bmi
np_height_m = np.array(height) * 0.0254
np_weight_kg = np.array(weight) * 0.453592
bmi = np_weight_kg / np_height_m ** 2
```

```
# Create the light array
light = bmi < 21
bmi[light]

# Print out light
print(light)

# Print out BMIs of all baseball players whose BMI is below 21
print(bmi[light])
```

```
[ True False False False False]
[ 18.79444882]
```

NumPy Side Effects

- Numpy array can not contain elements with different types
- Type Coercion: string, numeric, boolean

Subsetting NumPy Arrays

In [21]:
```
# height and weight are available as a regular lists

# Import numpy
import numpy as np

# Store weight and height lists as numpy arrays
np_weight = np.array(weight)
np_height = np.array(height)

# Print out the weight at index 50
print(np_weight[3])

# Print out sub-array of np_height: index 100 up to and including index 110
print(np_height[1:3])
```

```
250
[70 72]
```

2-Dimensional Arrays

In [31]:
```
np_2d = np.array([[1, 2, 3, 4, 5], [6, 7, 8, 9, 10]])
```

In [32]:
```
# ndarray is N-dimensional Array
type(np_2d)
```

Out[32]:
```
numpy.ndarray
```

In [33]:
```
# 2 rows, 5 columns
print(np_2d.shape)
```

```
(2, 5)
```