PYTHON FOR MATLAB USERS

Welcome to Python!

Justin Kiggins
Product Manager

Prerequisite: Manipulating matrices in MATLAB

```
% Example of manipulating matrices in MATLAB

v = [16 5 9 4 2 11 7 14];
disp(v(5:end))

A = magic(4);
disp(A(2:4,1:2))

A(A>12) = 10;

disp(A(:,1))
```

Prerequisite: Plotting data in MATLAB

```
% Example of plotting data in MATLAB

figure
plot(t,y,'b-')
xlabel('Time (s)')
ylabel('Sensor A')

figure
scatter(y1,y2,'go')
xlabel('Sensor A')
ylabel('Sensor B')

figure
histogram(y1,[0:0.01:1])
```

Prerequisite: Control flow of MATLAB scripts

```
% Example of control flow in MATLAB

fid = fopen('magic.m','r');
count = 0;
while ~feof(fid)
    line = fgetl(fid);
    if isempty(line) || strncmp(line,'%',1) || ~ischar(line)
        continue
    end
    count = count + 1;
end
count
```

If you don't know MATLAB...

Python does more than Data Science

- General purpose programming language
  - Integrate machine learning models into large-scale applications
  - Query data from public APIs
  - Build websites that can handle large traffic volumes

Getting started with data types

- Integer
- Float
- Boolean
- String

```
# Integer
x = 1

print(x)

1
```

```
type(x)

<class 'int'>
```

```
# Float
x = 1.0
print(x)

1.0
```

```
type(x)

<class 'float'>
```

Mathematical operators

Operation	Python Operator
Addition	+
Subtraction	-
Multiplication	*
Division	/
Exponentiation	**

```
a = 3 + 12
print(a)

15

b = 4 * 5.0
print(b)

20.0
```

Exponentiation in Python

$$area = \pi r^2$$

```
radius = 5
pi = 3.14

area = pi * (radius ** 2)
print(area)

78.5
```

Warning: Do NOT use the caret operator, ^

```
# This won't take 4 to the second power
print(4 ^ 2)

# This is the bitwise XOR
6
```

PYTHON FOR MATLAB USERS

Let's get started!

PYTHON FOR MATLAB USERS

Methods and Packages

Justin Kiggins
Product Manager

Methods are functions attached to variables

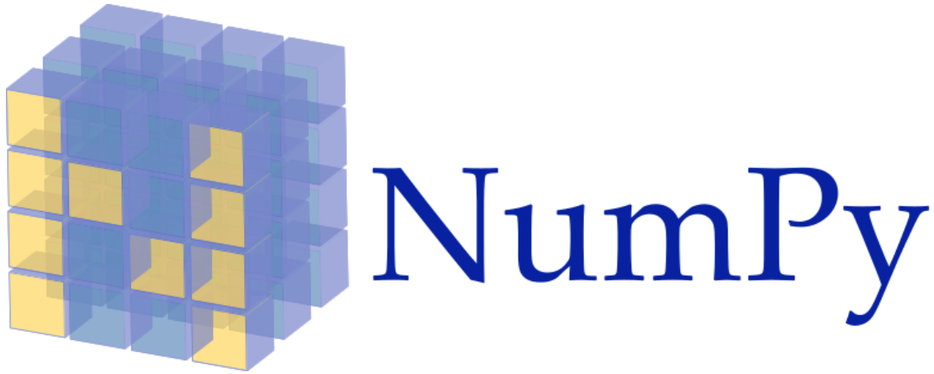
- Specific to the variable type
- Accessed with a period

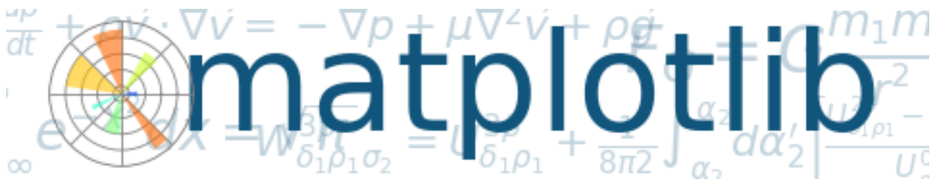
```
yelling = "WHY ARE YOU YELLING?"
whispering = yelling.lower()
print(whispering)

why are you yelling?
```

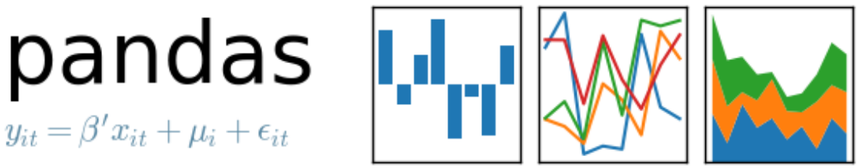
The .format() method

```
n_avocados = 1765
message = "I have {} avocados!".format(n_avocados)

print(message)

I have 1765 avocados!
```

Python packages







Packages have to be imported

```
import math

math.pi

3.141592653589793

math.log(0.9)

-0.10536051565782628
```

Importing part of a package:

```
from math import log
log(0.9)

-0.10536051565782628
```

Package aliasing

```
import datetime as dt

birth_date = dt.datetime(1961,8,4)
```

```
# Import NumPy package with common alias
import numpy as np

x = np.ndarray([1, 2, 3])

# Import pandas package with common alias
import pandas as pd

df = pd.DataFrame()

# Import Matplotlib pyplot module with common alias
import matplotlib.pyplot as plt

plt.show()

# Import Seaborn package with common alias
import seaborn as sns

sns.set()
```

PYTHON FOR MATLAB USERS

Let's explore some useful methods and packages

PYTHON FOR MATLAB USERS

Arrays & plotting

Justin Kiggins
Product Manager

NumPy arrays

- Arrays = matrices
- Arrays can be multidimensional
- Every element is the same type

Indexing into NumPy arrays

```
print(arr)

[[2, 4, 8, 16, 32]]
```

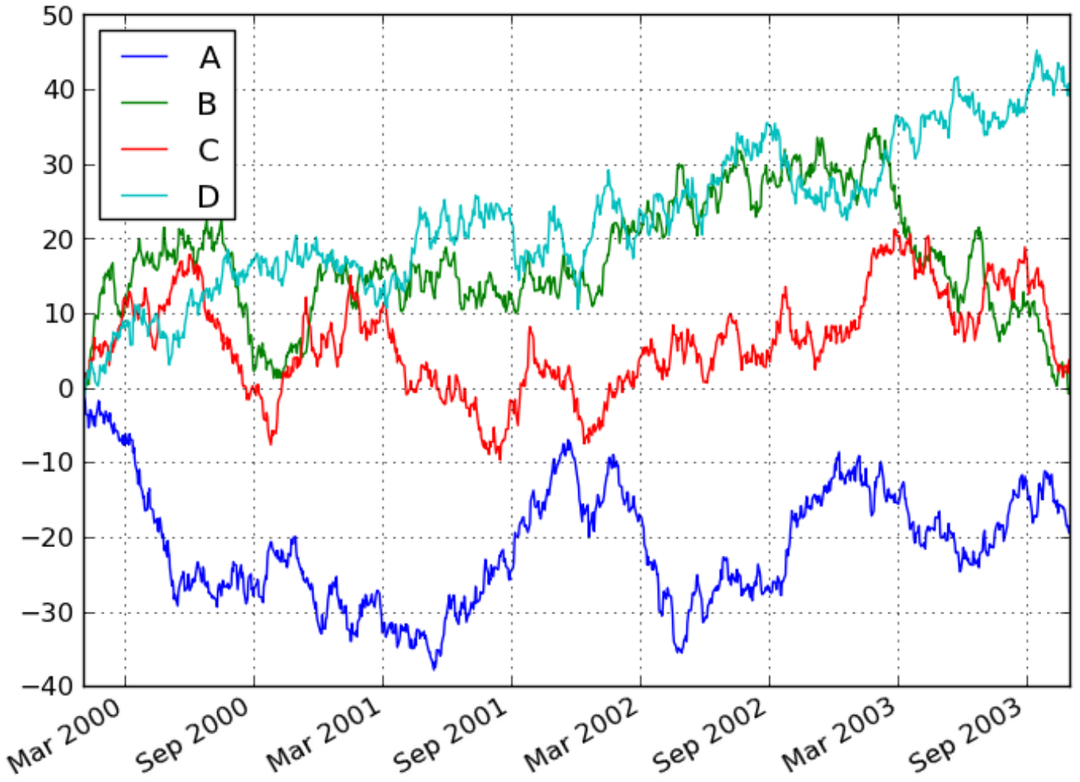
- Zero-based indexing

```
print(arr[2])

8
```

```
print(arr[0])

2
```

```
print(arr[-1])

32
```

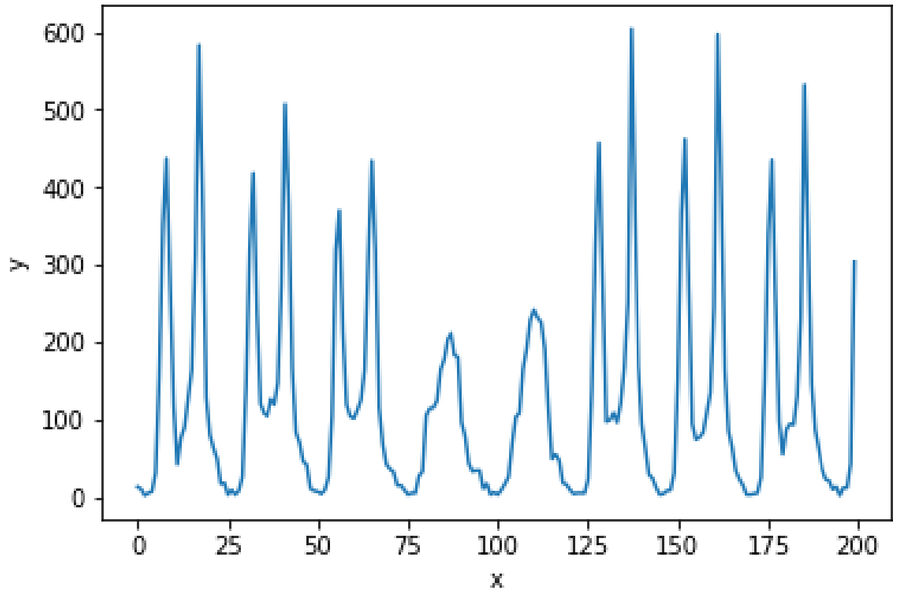
Visualizing data with Matplotlib



Using matplotlib

```
import matplotlib.pyplot as plt
```

```
# Create a new figure
plt.figure()

# Plot y as a function of x
plt.plot(x, y)

# Set the x-label and y-label
plt.xlabel('x')
plt.ylabel('y')

# Be sure to show the plot!
plt.show()
```



PYTHON FOR MATLAB USERS

Let's practice!