Python Week 1

Introduction & Review

AI Academy

Welcome AI Academy Cohort 3

Program Courses

1. Computer Programming with Python
2. Data Mining
3. Introduction to Artificial Intelligence
4. Machine Learning

AI Academy

NC STATE

Computer Programming with Python

Instructor: James E. Robinson, III

Teaching Assistant: Travis Martin

AI Academy

NC STATE

About Travis

Travis has several degrees including a master’s degree in computer science education, a bachelors in physics education, and a bachelors in computer engineering. Additionally, he has taught high school science for 15 years, mostly physics.

Travis currently lives in Texas but has lived in multiple places in the US over the course of his life.

About James

James has a Masters of Science in Computer Engineering from NC State University. The latter with a focus on computer networking and software design.

While currently involved primarily in systems architecture, James still writes integration code for projects when required (Python, SQL, cloud service APIs, Amazon Web Services).

About James

- Pascal
- C
- C++
- TCSH
- BASH
- Awk
- Perl
- Java
- TCL/TK
- Ruby
- Go
- JavaScript
- Python

Development Environment

This environment will be used by subsequent AIA courses.

Recommended

- Windows 10
- Anaconda v. 2021.11 or later
  - Interpreter: Python 3.9 - included
  - IDE: Spyder v5 - included
  - Modules: numpy, pandas, matplotlib, etc. - included

Lightning Review

- Variables
- Expressions
- Functions
- Conditional Control
- Looping
- Lists & Tuples
- Nested Lists
- Objects / Classes
- File Handling
- Dictionary
- Debugging
- Error Types
- Using Modules

Variables - labels for things

```
# good examples
full_name = "Laila Ali"
street_name = "Butterfly Street"

# bad examples
XÆA12 = "son"

# yes, Python does understand unicode characters
# no, do not use them for variable/object names
```

AI Academy

NC STATE

Expression

Some combination of values, variables, and operators.

```
# examples
y = 3
x = 2
z = y % x  # modulo/remainder operator
a = 2 * (x + z) + y # PEMDAS FTW
```

Function

```
# A block of code that runs when called.
# May take parameters, may return something.

def my_summer(param1, param2 = 0):
    '''
    Sum two values.
    Inputs:
        param1 - int or float
        param2 - int or float, optional
    Returns:
        sum - int or float
    '''
    return param1 + param2
```

Conditional Control

```
# execute some statements based on a condition
if temperature > 100.3: # per CDC guidelines
    print("Patient has a fever")
    if temperature > 102.9:
        print("Patient should seek medical attention")
elif temperature > 99.5:
    print("Patient has low-grade fever")
else:
    print("Patient is either normal or has assumed room temp.")
```

Looping

```
x = 1; y = 6
while x < y:
    x += 3
    print("while: good data")

for x in range(1, 100, 3):
    if x < y:
        print("for: good data")
    else:
        break
```

Lists

```
# a list
a = ['foo', 'bar', 'baz', 'qux']
```

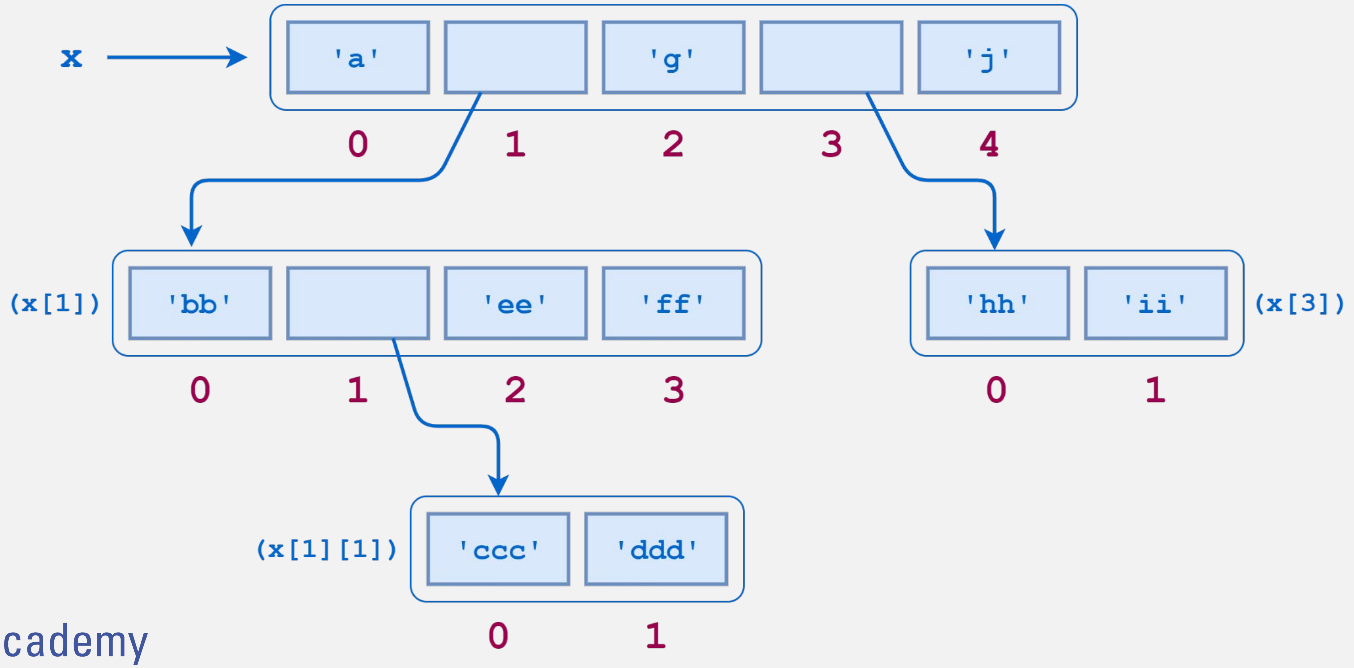
- Lists are ordered.
- Lists can contain any arbitrary objects.
- List elements can be accessed by index.
- Lists can be nested to arbitrary depth.
- Lists are mutable.
- Lists are dynamic.

Tuples

```
# a tuple is a list that you can't change - immutable

t = ('foo', 'bar', 'baz', 'qux', 'quux', 'corge')

t[1] = 'boo'

>>> TypeError: 'tuple' object does not support item assignment
```

Nested Lists

x = ['a', ['bb', ['ccc', 'ddd'], 'ee', 'ff'], 'g', ['hh', 'ii'], 'j']



AI Academy

NC STATE

Objects & Classes

```
# class definition
class Dog:
    def __init__(self, name, breed)
            self.name = name
            self.breed = breed
            return

# object created from class
mydog = Dog("Julie", "mixed")
```

File Handling

```
# opening, reading/writing, closing
# use 'with' statement to make it easier

with open('class.txt', 'r') as fp:
    contents = fp.read()

# with handles closing for us
```

Dictionary

```
# dictionaries are ordered, keys are unique, mutable
thisdict = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

print(thisdict["brand"])
```

Debugging

- Syntax Errors
- Runtime Errors
- Semantic Errors

Error Types

- NameError
- TypeError
- KeyError
- AttributeError
- IndexError

Using Modules

- using import
- import best practices
  - what do you need?
  - use a name that fits
- referencing module methods

Example

Problem: Using the NC Lottery Cash 5 data, what is the distribution of winning numbers by ball? Graph the result (5 graphs).

AI Academy

NC STATE