Python Programming for FRC Scouting and Data Analysis

Curriculum

# Session 1: Overview and System Setup

1. Topics
   1. Brief overview of Python and why we use it
   2. Ensure all students have a Windows 10 laptop with the Anaconda Python Stack installed.
   3. Create Python project folder and configure Windows command line
   4. Configure environment variables (PATH and PYTHONPATH) and verify Python installation functions correctly.
   5. Review online Python tutorial and reference material
   6. Python command line tool
   7. Python mathematical operators
2. Training objectives: 1, 2, 3, 4, 14
3. Reading Assignment
   1. Python Tutorial: <https://docs.python.org/3/tutorial/index.html>, Sections 1 and 2.
   2. Windows Command Prompt: <https://www.computerhope.com/issues/chusedos.htm>
   3. Windows Environment Variables: <https://windowsreport.com/edit-windows-path-environment-variable/>
4. Exercises
   1. Calculate the average finals score for the winners of each division at the Houston world championships.
   2. Calculate 13 to the 18th power.
   3. What is the remainder of 13 to the 18th power divided by 1318?
   4. Print the value of the PATH environment variable from the command line
   5. Using only the command line, navigate from you Python project file to the root folder (C:/) and list the contents.

# Session 2: Data Types and Control Statements

1. Topics
   1. Jupyter Notebooks
   2. Assigning values to variables
   3. Data types
   4. Calling built-in functions
   5. Python Standard Library packages
   6. For loops
   7. Logical operators
   8. If statements
   9. While loops
   10. Lists
2. Training objectives: 5, 6, 7, 8, 9, 13, 18, 19, 21, 22, 23, 26
3. Reading Assignment
   1. Python Tutorial: <https://docs.python.org/3/tutorial/index.html>, Sections 3, 4.1 – 4.5.
   2. Jupyter Notebooks: <https://jupyter-notebook.readthedocs.io/en/stable/examples/Notebook/Notebook%20Basics.html>, *Notebook Basics* section.
4. Exercises
   1. Use a for loop to calculate the factorial of 1318
   2. Use a while loop to calculate the factorial of 1318
   3. Create a list containing all integers between 0 and 1318 that are multiples of 13 or 18.
   4. Analyze FIRST API schedule to find all matches with a specific team
   5. Analyze FIRST API schedule to find a match where two given teams play each other

# Session 3: Functions, Modules, and Scope

1. Topics
   1. Defining functions
   2. Function arguments
   3. Function return values
   4. Default Argument values
   5. Named and positional arguments
   6. Running Python code from command line
   7. Variable Scope
   8. Comments
   9. Function and module documentation strings
2. Training objectives: 11, 12, 15, 16, 17, 33,
3. Reading Assignment
   1. Python Tutorial: <https://docs.python.org/3/tutorial/index.html>, Sections 4.7, 4.7.1, 4.7.2, 4.7.6, 4.8, 6.1 – 6.3
4. Exercises
   1. Write a function that calculates the area of a circle given the radius.
   2. Write a function that takes the playoff alliance numbers as arguments (e.g., alliance #1, alliance #5, etc) and returns a string indicating the level at which those alliances would meet in the playoffs (e.g., finals, quarterfinals, etc.). Include comments in the function.
   3. Convert functions for exercises a and b into code that can be run from command line and accepts command line arguments.
   4. Write a function that calculates the probability that in a group of N people, no two people will have the same birthday. (Ignore leap year and assume that all days of the year have equal probability for being someone’s birthday.)
   5. Import a module with functions into a second module. Call the function from the first module in the second module.
   6. Write documentation strings for all of your functions
   7. Write a documentation string for a module.

# Session 4: Modules, Importing, and Packages

1. Topics
   1. Modules
   2. Import statements
   3. Creating Packages
   4. Coding Style
2. Reading Assignment
   1. Python Tutorial: <https://docs.python.org/3/tutorial/index.html>, Section 6
   2. Google Python Style Guide: <https://google.github.io/styleguide/pyguide.html>, Python Style Rules section
   3. Atom Basics: <https://flight-manual.atom.io/getting-started/sections/atom-basics/>

# Session 5: Introduction to Classes

1. Topics
   1. Instance Attributes and Methods
   2. Static Attributes and Methods
   3. Constructing and Using Classes
   4. Frequently Used Classes in Python Standard Library

# Session 6: Testing Your Code: Pytest, Pylint, and Debugging

1. Topics
   1. Evaluating code with Pylint
   2. Coupling and Cohesion
   3. Assert statements
   4. Writing and running unit tests with Pytest
   5. Using breakpoints to step through code

# Session 7: Data Structures: Lists, Dictionaries, and JSON

1. Topics
   1. Creating and using lists
   2. Creating and using dictionaries
   3. Looping through lists and dictionaries
   4. Dictionary and list comprehensions
   5. JSON syntax
   6. Converting between lists, dictionaries, and JSON
   7. Transforming lists and dictionaries with map() and filter()

# Session 8: Manipulating Text: Python String Functions and Regular Expressions

1. Topics
   1. Basic string manipulation in Python
   2. Python string formatting tools
   3. Regular expressions

# Session 9: Downloading FRC Data via HTTP

1. Topics
   1. Overview of Hypertext Transfer Protocol (HTTP)
   2. HTTP tools in Python
   3. Using HTTP to download FRC data from *The Blue Alliance*
   4. Using HTTP to download FRC data from *FIRST API*

# Session 10: Essential Programming Tools: Git, Github, and Managing Packages

1. Topics
   1. Saving backups of code in Git repositories
   2. Sharing code via Github
   3. Git Branches
   4. Merging and Rebasing
   5. Installing and updating Python Packages with Anaconda

# Session 11: Numpy Arrays and Pandas Dataframes

1. Topics
   1. Numpy Arrays
   2. Creating dataframes
   3. Querying dataframes
   4. Manipulating (e.g., stacking, unstacking, filtering, etc.) dataframes

# Session 12: Plotting

1. Topics
   1. Pandas Plotting Tools
   2. Matplotlib Plotting Package
   3. Holoviews Plotting Package
   4. Types of plots: bar charts, histograms, scatter plots, line plots, etc.

# Session 13: Useful Things: Python Tools for Descriptive Statistics, Set Theory, and Other Stuff

1. Topics
   1. Python tools for means, trimmed means, and medians
   2. Python tools for variance, covariance and correlation
   3. Understanding point estimation and confidence intervals
   4. Intersections, unions, complements, partitions, and the Python Set datatype
   5. Other useful Scipy tools: Sympy for symbolic mathematics, Scipy scientific library

# Session 14: Advanced Python

1. Topics
   1. Creating classes
   2. Generators and the YIELD statement
   3. Exceptions and Error Handling

# Session 15: Relational Databases and Structured Query Language

1. Topics
   1. Overview of relational databases
   2. Postgresql database server
   3. SQLalchemy package
   4. SQL SELECT statements
   5. Joining tables
   6. Using INSERT, UPDATE and DELETE statements to modify data in databases
   7. Transferring data between relational databases and Pandas dataframes

# Session 16: Creating User-friendly Interfaces in Jupyter

1. Topics
   1. Using custom Jupyter packages
   2. Jupyter notebook initialization
   3. iPython widgets for Jupyter notebooks
   4. Markdown text
   5. Embedding HTML in Jupyter notebooks

# Session 17: HTTP Servers

1. Topics
   1. Overview of CherryPy HTTP Server
   2. Introduction to HTML
   3. Introduction to Cascading Style Sheets
   4. Introduction to Javascript