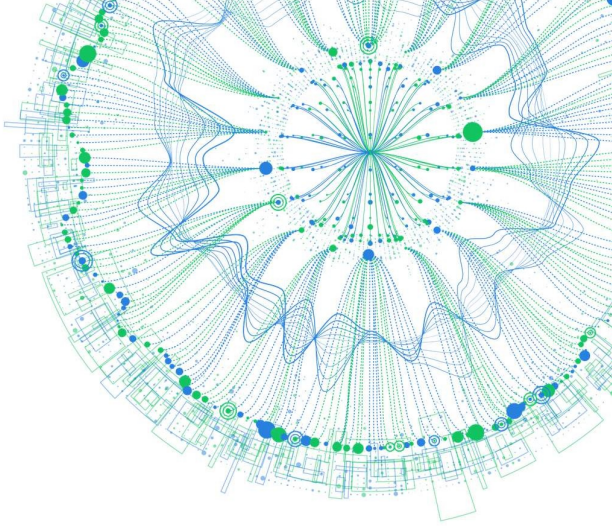


Python for Engineers

Kubrick's Python training syllabus is aimed to provide a practical foundation of Python for use in data driven applications.

Part 1 – Introductory Overview

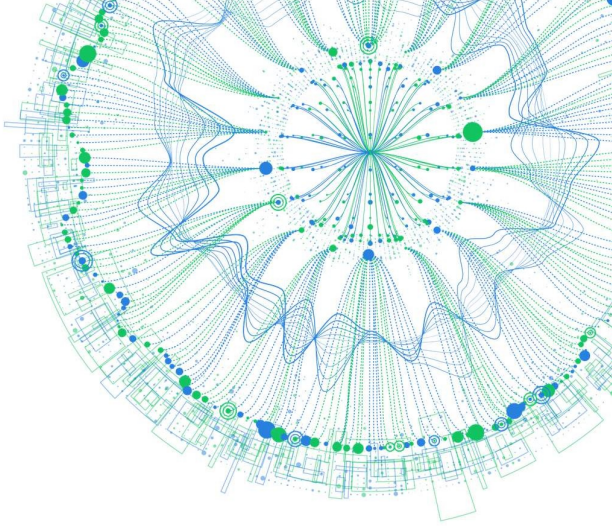
1. **Getting Started**
- Installation
- Running Python and Jupyter Notebooks

2. **Introduction to Concepts**
- Variables and Arithmetic Expressions
- Conditional Statements
- File I/O
- Lists, Tuples, Sets and Dictionaries
- Functions
- Objects and Classes
- Exceptions
- Modules and Packages

3. **Conventions and Syntax**
- Termination of statements
- Indentation
- Numeric and string literals
- Containers
- Operators
- Documentation strings

Part 2 – Types and Objects

1. **Introduction and Terminology**
- Object, Identity, Type
- Mutability and Immutability
- References and copies
- Reference counting
- First class objects



2. **Built-In Types for Representing Data**
- None
- Numbers
- Sequences
- Mapping
- Sets
- String Methods

3. **Built-In Types for Representing Programming Structure**
- Callables
- Types, Classes and Instances
- Modules

Part 3 – Program Structure

1. **Boolean Expressions and Truth Values**
- Using the and, or and not statements.

2. **Conditional Execution**
- if statements

3. **Iteration and Looping**
- while loops
- for loops

4. **Handling Errors**
- Exceptions
- Exception handling
- Exception hierarchy

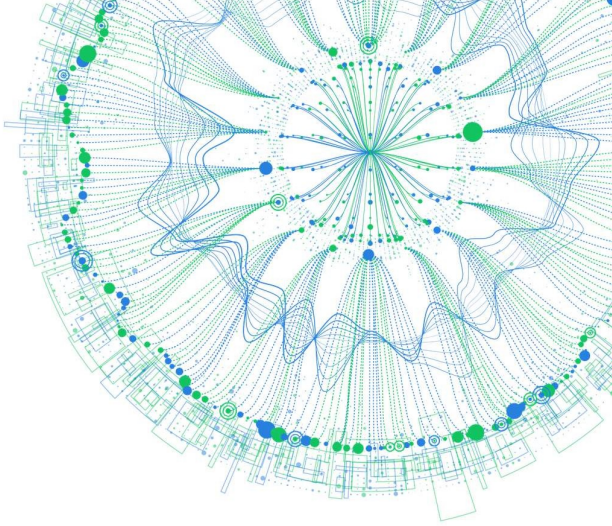
Part 4 – Functions

5. **Introduction to Functions**
- Why we need functions

6. **Parameter Passing and Return Values**
- Default arguments
- Variable number of arguments
- Passing in mutable arguments

7. **Scoping Rules**
- Local variables
- Global variables

8. **Decorators**
- Use of the @ operator



9. **Generators**
- Generators vs Lists

10. **Comprehensions**
- List comprehensions
- Generator comprehensions

11. **Lambda Functions**
- Definition and applications

12. **Other Convenience Functions**
- Map
- Filter

Part 5 – Object Oriented Programming

1. **Introduction to Classes**
- Why we need classes
- Differences between the class and the instance
- Three big ideas behind object oriented programming

2. **Scoping Rules**
- Instance variables
- Use of the self keyword
- Scoping within classes

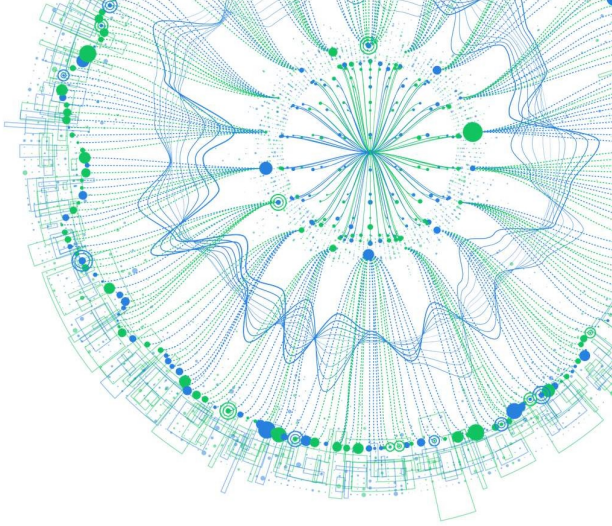
3. **Inheritance**
- Inheriting from a single class
- Multiple inheritance

4. **Polymorphism**
- Definition and why it is useful

5. **Method Types**
- Instance methods
- Class methods
- Static methods

6. **Properties**
- Definition and why we need them

7. **Data Encapsulation**
- Why we need encapsulation
- Private attributes



Part 6 – Web Scraping

8. **Introduction to Web Scraping**
- Types of http requests
- JSON and HTML

9. **Requests**
- Making requests
- status codes

10. **BeautifulSoup**
- Why we need BeautifulSoup
- Navigating through HTML using BeautifulSoup

11. **Selenium**
- Why we need Selenium
- Loading and finding elements on a webpage

Part 7 - Pandas

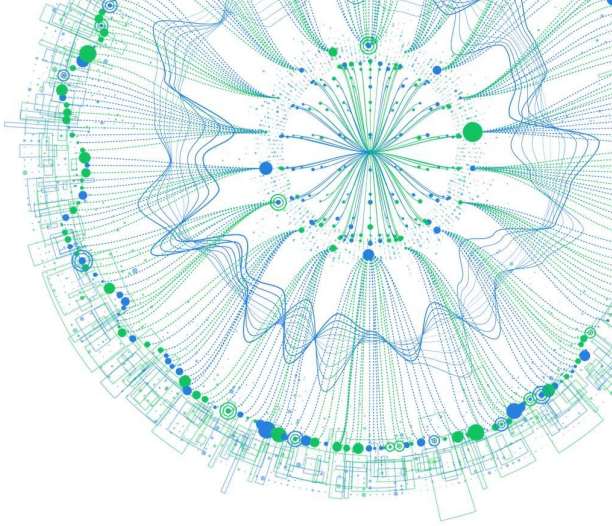
1. **Introduction**
- Why we need Pandas
- Primary data types in Pandas
- Series and DataFrame objects

2. **Construction of Series and DataFrames**
- Construction Methods
- Data types
- Shape

3. **Data Indexing and Selection**
- Pandas Index object
- Implicit indexing
- Explicit indexing
- Slicing
- Boolean Indexing

4. **Element-Wise Data Operations**
- Numpy's universal functions
- Using the apply method

5. **Handling Missing Data**
- Dropping
- Imputation



6. **Combining Datasets**
- Concatenation / Unions
- Joining

7. **Aggregation**
- Aggregation functions
- Grouping
- Pivot Tables

8. **Time Series**
- Introduction to time series
- Resampling
- Various time series methods

Part 8 - Plotting using Python

1. **Plotting using Matplotlib**
- MATLAB style Vs Object Oriented Interface
- Line plots
- Scatter plots
- Histograms
- Contour plots
- Surface plots

2. **Plotting Pandas DataFrames**
- Scatter matrices
- In-built plotting methods

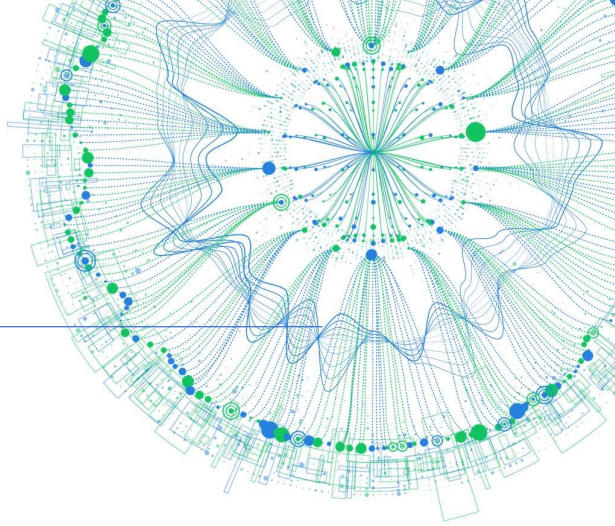
Extensions

1. **Regular Expressions**
- Introduction to regex patterns
- Matching and finding using regular expressions

2. **Interfacing with Databases**
- Connecting to databases using PyODBC

3. **Modules and Packages**
- Python Modules
- Grouping Modules into Packages

4. **Virtual Environments**
- Why we need virtual environments
- Creating / Saving / Loading virtual environments



Module Duration ~ 10 days