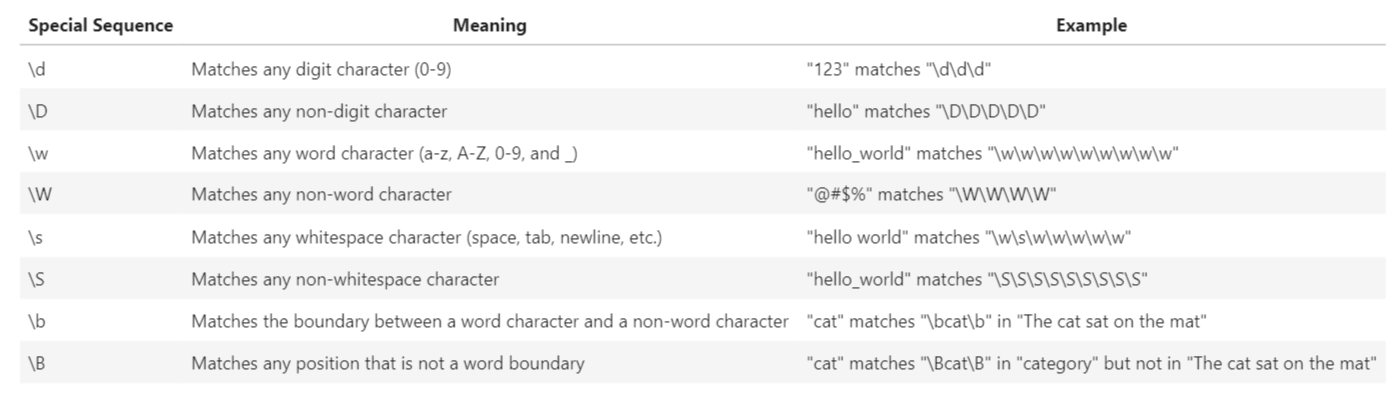
**Week 1:**

Commands:

Sys.version - gives python version you have

Sys.float - info gives system setting about float data type

For RegE, import re

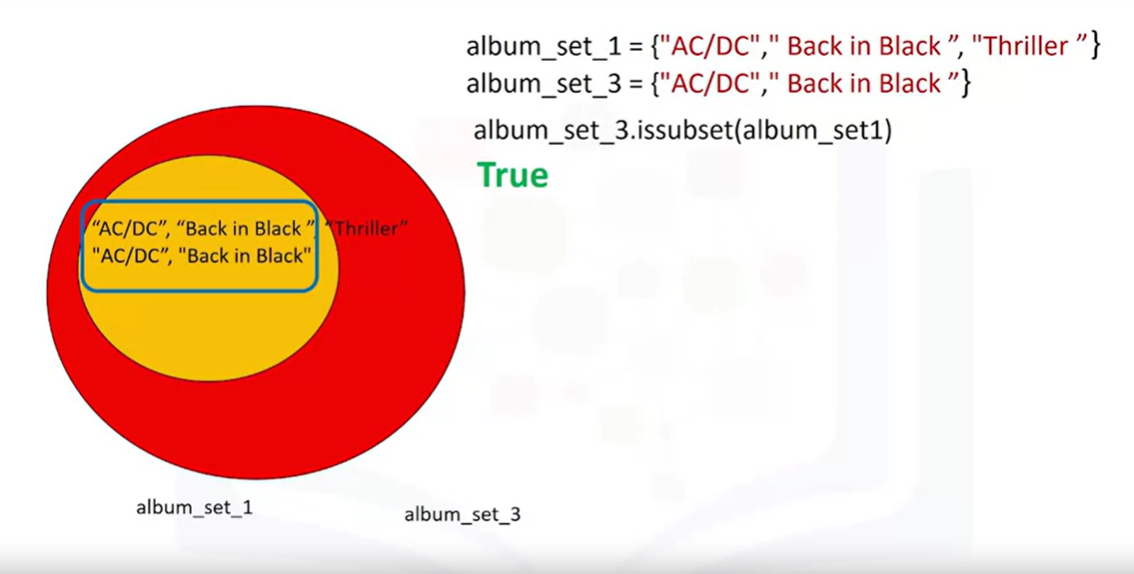


**Week 2: Lists and Tuples**

Tuples are immutable while lists are mutable

B=A[:] means variable B references a new copy or clone of list A

Unlike tuples and lists which are ordered, sets are unordered



**Week 3: Conditions and Branching**

Comparing characters: 'BA' > 'AB' returns True as in multiple letters, the first letter takes precedence in ordering

**Help()** on a function returns the documentation string (mentioned in ‘’’ in the function)

Functions – two types: pre-defined and user defined

When the number of arguments are unknown for a function, They can all be packed into a tuple as shown: def printAll(\*args):

The arguments can also be packed into a dictionary as shown: def printDictionary(\*\*args):

***Note:*** how the changes made to the list are not limited to the functions scope. This occurs as it is the lists **reference** that is passed to the function - Any changes made are on the orignal instance of the list. Therefore, one should be cautious when passing mutable objects into functions.

For example, def addItems(list):

list.append("Three")

list.append("Four")

myList = ["One","Two"]

addItems(myList)

myList

This will return ['One', 'Two', 'Three', 'Four']

**Exception handling:**

Use **try**, **except**, **else** and **finally** statement. Exception will be thrown when execution failed and in case execution succeeds, the message in else statement will be printed. Once we open a file, we have to close it also, so we use finally statement at the end.

You can also have an empty <code>except</code> at the end to catch an unexpected exception

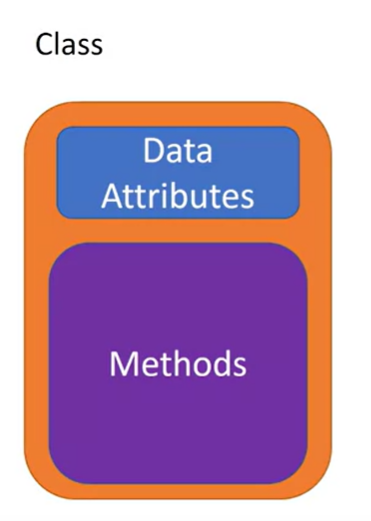
**Objects and classes:**

An object is an instance of a particular type. For example, int object or an instance of type int.

Type() command is used to get the type of an object.

A class or type’s methods are functions the instance of the type or type provides. For example, sort() method of type list.

A class has data attributes and methods. For example, for class Circle, the data attributes will be radius and color.



class Circle(object):

def \_\_init\_\_(self, radius, color): //class contructor used to initialize data attributes

self.radius = radius;

self.color = color;

redcirlce = Circle(10, ‘red’) // creating an instance of the class

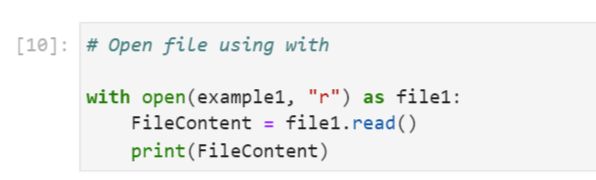
***Note:***

dir(Nameofobject): function is useful to obtain list of data attributes and methods of that object

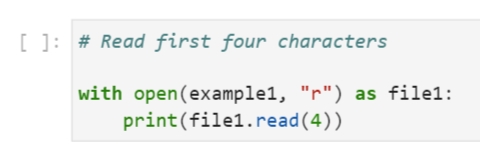
**Week 4:**

**Reading and Writing files:**

Use ‘with open’ statement as it automatically closes the file object at the end.



To Read first four characters:



We use 3 modes in open function: ‘r’, ‘w’ and ‘a’. We can write to files without losing any of the existing data by setting the mode argument to append ‘a’. Additional modes include:

* **r+** : Reading and writing. Cannot truncate the file.
* **w+** : Writing and reading. Truncates the file.
* **a+** : Appending and Reading. Creates a new file, if none exists.

**To know cursor location:**

* .tell() - returns the current position in bytes
* .seek(offset,from) - changes the position by 'offset' bytes with respect to 'from'. From can take the value of 0,1,2 corresponding to beginning, relative to current position and end

**Note:**

* The difference between **w+** and **r+**. Both of these modes allow access to read and write methods, however, opening a file in **w+** overwrites it and deletes all pre-existing data.  
  To work with a file on existing data, use **r+** and **a+**.
* While using **r+**, it can be useful to add a .truncate() method at the end of your data. This will reduce the file to your data and delete everything that follows.
* After reading files, we can also write data into files and save them in different file formats like **.txt, .csv, .xls (for excel files) etc**. You will come across these in further examples

**Loading data into Pandas:**

* Use Pandas Library to create DataFrame and Series
* Locate data in the DataFrame using loc() and iloc() functions
* Use slicing

**Numpy:**

NumPy is a Python library used for working with arrays, linear algebra, fourier transform, and matrices.A numpy array is similar to a list. NumPy stands for Numerical Python and it is an open source project.The array object in NumPy is called **ndarray**, it provides a lot of supporting functions that make working with ndarray very easy.

Note: List is casted to numpy array which is of fixed size and each element is of same type. And the type of the array is numpy.ndarray

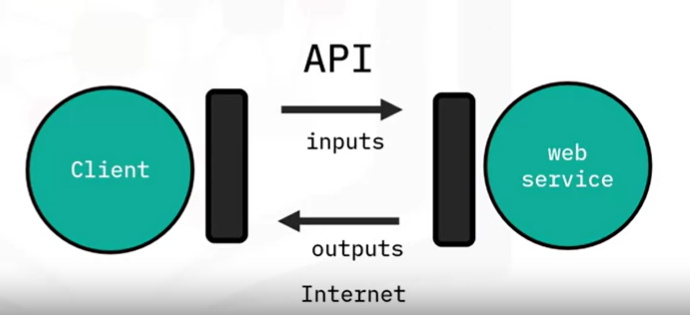
Python library used for scientific computing and is a basis for Pandas is Numpy

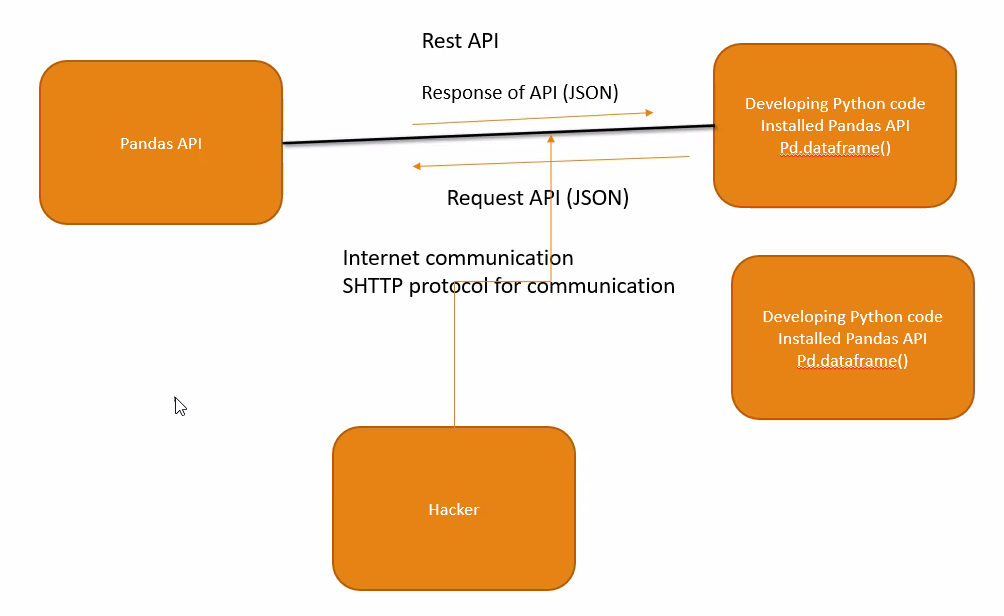
**Week 5:**

**Simple API (Application Program Interface):**

Pandas is an API library within which there are many APIs such as dataframe.

REST APIs – Representational state transfer APIs



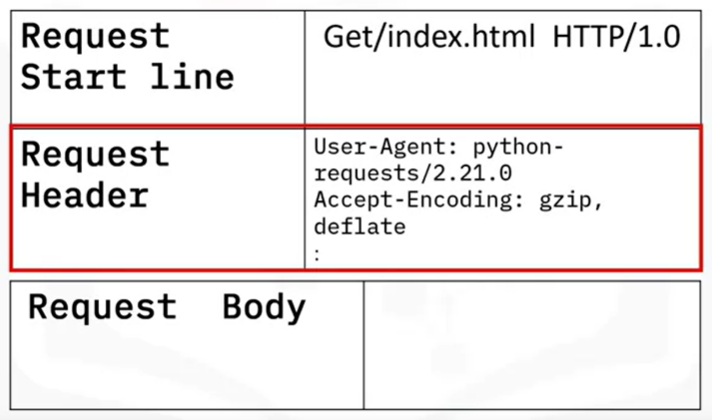


**Rest APIs:**

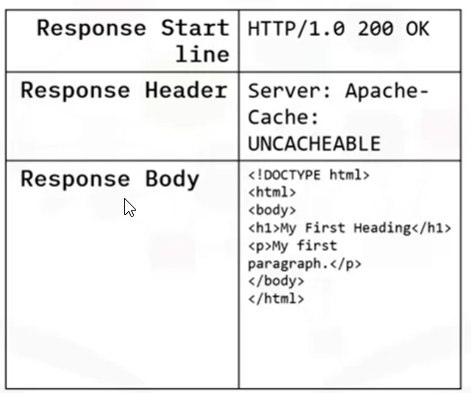
**Overview of HTTP:**

* Scheme: http://
* Internet address or base URL (Uniform resource locator): for example, [www.ibm.com](http://www.ibm.com)
* Route: location on web server. For example, /images/IDSNlogo.png

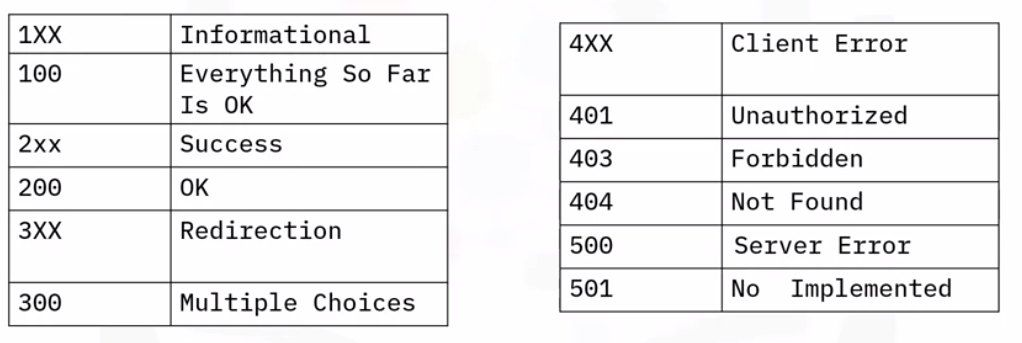
**Request Message:**



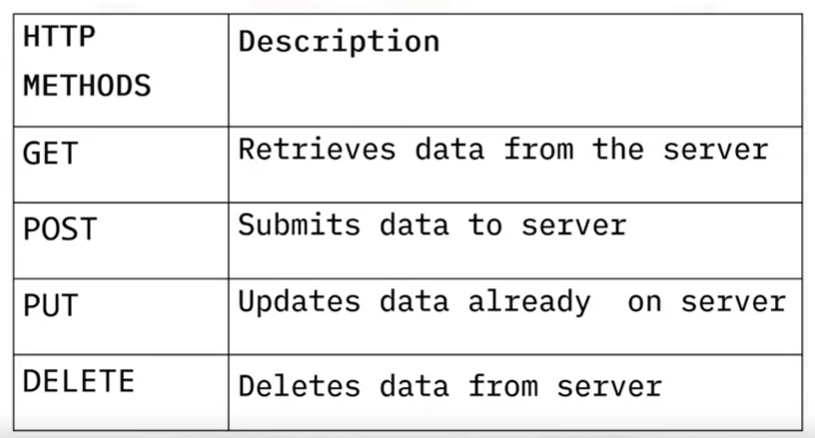
**Response Message:**



**Status Codes:**



**HTTP Methods:**

****

**Data Engineering Process:**

1. **Extract** - Data extraction is getting data from multiple sources. Ex. Data extraction from a website using Web scraping or gathering information from the data that are stored in different formats(JSON, CSV, XLSX etc.).
2. **Transform** - Transforming the data means removing the data that we don't need for further analysis and converting the data in the format that all the data from the multiple sources is in the same format.
3. **Load** - Loading the data inside a data warehouse. Data warehouse essentially contains large volumes of data that are accessed to gather insights.

**Writing JSON to a File**

This is usually called **serialization**. It is the process of converting an object into a special format which is suitable for transmitting over the network or storing in file or database.

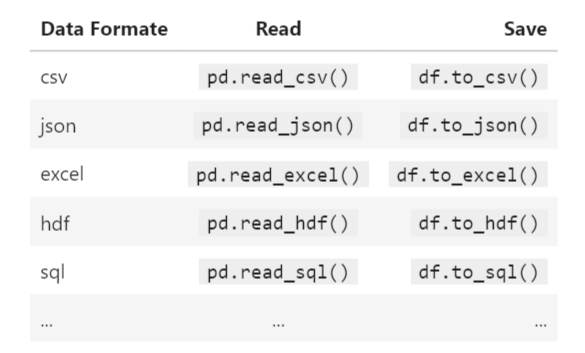
**Reading JSON to a File**

This process is usually called **Deserialization** - it is the reverse of serialization. It converts the special format returned by the serialization back into a usable object.

### **Writing with xml.etree.ElementTree**

The **xml.etree.ElementTree** module comes built-in with Python. It provides functionality for parsing and creating XML documents. **ElementTree** represents the XML document as a tree. We can move across the document using nodes which are elements and sub-elements of the XML file.

## **Read/Save Other Data Formats**



**Reading binary format:**

* 1. **Reading Images:**

**PIL** is the Python Imaging Library which provides the python interpreter with image editing capabilities.

**Visualization**

Visualization is one of the best way to get insights from the dataset. Seaborn and Matplotlib are two of Python's most powerful visualization libraries.