A picture containing drawing

Description automatically generated

**FAQ:**

1. Introduction to Python data structures
2. **Lists.**

**Why do We Need Lists?**  
Let us talk about why we need a data structure like a list or when to use it. We will borrow an example from the world of Wall Street for this discussion.

Companies listed on the NASDAQ exchange have ticker symbols or abbreviations for each company name. For e.g., the ticker symbol for Alphabet, Inc. is GOOGL.

Imagine now that you own stocks for one company, say Microsoft, and want to be able to print out the ticker symbol of your stock. Since it is one value, you can store it in the variable microsoft, and assign it the value of MSFT. Like this:

microsoft = MSFT

Well, that is convenient! So, now when you want to print the ticker symbol for the company you hold stocks for, you use the print command.

print(microsoft)

>>> MSFT

Let's now consider that you are an investment fund manager, and you want to print out the stocks (or holdings) you own in an index fund (e.g., Vanguard Institutional Index Fund). An index fund includes stocks (also called holdings) for many companies. Turns out Vanguard Institutional Index Fund has 506 holdings!

Printing the tickets for all 506 holdings using individual strings would require 506 strings. Not ideal! Because we'll need to remember the name of each string to print it.  
You also have to think about how to group the 506 strings under the same index fund. Not convenient at all!

**This is where the beauty of data structures comes into play! You can use a list.**

Since index funds have ticker symbols too, you use that as the name for the list, here VINIX, and add the ticker symbols for all the holdings into that list. Let's populate the list with the top holdings listed for Vanguard Institutional Index Fund .

**VINIX** = ['C', 'MA', 'BA', 'PG', 'CSCO', 'VZ', 'PFE', 'HD', 'INTC', 'T', 'V', 'UNH', 'WFC', 'CVX', 'BAC', 'JNJ', 'GOOGL', 'GOOG', 'BRK.B', 'XOM', 'JPM', 'FB', 'AMZN', 'MSFT', 'AAPL']

Now, printing the tickers becomes slightly easier. And you don't have to remember the names of the strings!

print(VINIX[0])

>>> C

print(VINIX[1])

>>> MA

Later you will learn about more efficient ways to print the elements in a list.

You can even use the list to see if a particular stock **is in** the index fund VINIX or not.  
Like this:

'GE' **in** VINIX

>>> False

'GOOGL' **in** VINIX

>>> True

**1. How do I index a nested list? For example if I want to grab 2 from [1,1,[1,2]]?**

You would just add another set of brackets for indexing the nested list, for example: my\_list[2][1] .

1. **Dictionaries.**

**1. Do dictionaries keep an order? How do I print the values of the dictionary in order?**

Dictionaries are mappings and do not retain order! If you do want the capabilities of a dictionary but you would like ordering as well, check out the **ordereddict** object lecture later in the course!

1. Tuples.
2. Sets.

**---X---**