Week 4: Daily Morning Challenge

Day 1: Tuesday 14th January 2020

Adeyeye Ayobami (A'why)

Question 1:

Use the table below to identify what scale of measurement (Nominal, Ordinal, Interval or Ratio) best represents the data.

	Data Set	Scale of Measurement
1	Genotype	Nominal
2	Temperature	Interval
3	Socio-economic status	Ordinal
4	Gender	Nominal
5	Income level	Interval
6	Credit score	Interval
7	Race	Nominal
8	Satisfaction rating	Ordinal
9	Weight	Ratio
10	Political party	Nominal

Question 2:

What is the difference between Array, List and Dictionary data structures?

A list is a sequential (abstract) data type that represents a countable number of ordered values, where the same value may occur more than once. Lists are mutable. It is a non-primitive type of data structure in the classification of data structure. Every element on a list contains at least two fields, one is used to store data and the other one is used for storing the address of next element.

Arrays are the set of homogeneous data elements stored in RAM. So, they can hold only one type of data. The data may be all integers, all floating numbers or all characters. Values in an array are identified using array name with subscripts. Single sub-scripted variables are known as a one-dimensional array or linear array; two sub-scripted variables are referred as a two-dimensional array

Arrays and dictionaries both store collections of data, but differ by how they are accessed. Items in an array are accessed by position (often a number) and hence have an order. Items in a dictionary are accessed by key and are unordered.

Question 3:

Give a short overview of how database technology has evolved in the 21st century (more specifically as regards SQL vs NoSQL)

The SQL programming language was first developed in the 1970s by IBM researchers Raymond Boyce and Donald Chamberlin. The programming language, known then as SEQUEL, was created following the publishing of Edgar Frank Todd's paper, "A Relational Model of Data for Large Shared Data Banks," in 1970. In his paper, Todd proposed that that all data in a database be represented in the form of relations. It was based on this theory that Boyce and Chamberlin came up with SQL. In the book "Oracle Quick Guides (Cornelio Books 2013)," author Malcolm Coxall writes that the original SQL version was designed to manipulate and retrieve data stored in IBM's original relational database management systems known as "System R." It wasn't until several years later, however, that the SQL language was made available publicly. In 1979, a company called Relational Software, which later became Oracle, commercially released its own version of the SQL language called Oracle V2. Since then, the American National Standards Institute (ANSI) and the International Standards Organization have deemed the SQL language the standard language in relational database communication. While major SQL vendors do modify the language to their desires, most base their SQL programs off of the ANSI approved version.

The acronym NoSQL was first used in 1998 by Carlo Strozzi while naming his lightweight, open-source “relational” database that did not use SQL. The name came up again in 2009 when Eric Evans and Johan Oskarsson used it to describe non-relational databases. Relational databases are often referred to as SQL systems. The term NoSQL can mean either “No SQL systems” or the more commonly accepted translation of “Not only SQL,” to emphasize the fact some systems might support SQL-like query languages.

NoSQL developed at least in the beginning as a response to web data, the need for processing unstructured data, and the need for faster processing. The NoSQL model uses a distributed database system, meaning a system with multiple computers. The non-relational system is quicker, uses an ad-hoc approach for organizing data, and processes large amounts of differing kinds of data. For general research, NoSQL databases are the better choice for large, unstructured data sets compared with relational databases due to their speed and flexibility.

Non-relational databases have their own strengths and weakness, as do relational databases. As the NoSQL revolution continues, it is important to remember “the right tool for the right job” is a useful philosophy. Relational databases support accuracy and redundancy, while non-relational databases support research. Currently, efforts are being made to merge the two database systems. Hybrid systems are adding SQL-type features like transactional support, joins, and customizable consistency. Additionally, SQL databases (for example SQL Server) are adding NoSQL features which allow more transparent tactics when using horizontal scaling. It seems reasonable to predict non-relational and relational databases will continue to merge eclectically, adding strengths and minimizing weaknesses.