Week1 Review Big Data

**1) What is Big Data?**

A collection of data that is huge in volume yet growing exponentially with time. It is data with so large size and complexity that none of the traditional data management tools can store it or process it efficiently. Big data is also data but with a huge size.

**Challenges of traditional decision-making**

* Takes long time to arrive at a decision.
* Planning, execution and reporting is not linked
* Provides only birds eye view
* Unable to make fully informed decisions

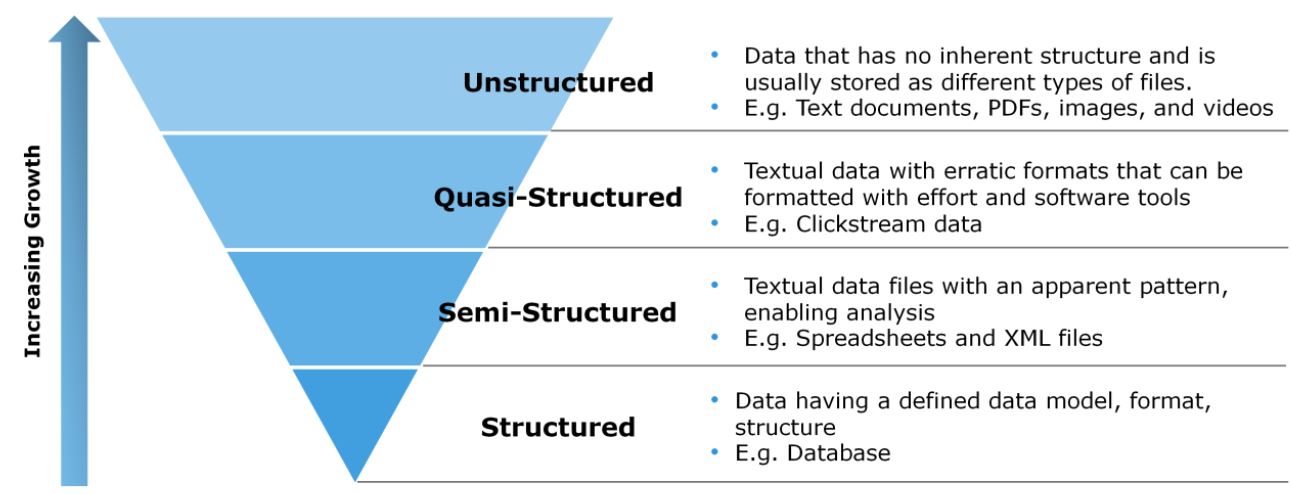
**2) What are the different types of data?**

a) Structured

b) Unstructured

c) Quasi - Structured

d) Semi-Structured



**3) Difference between Structured data vs Unstructured data?**

**Structured** data is comprised of clearly defined data types with patterns that make them easily searchable; while

**Unstructured** data – “everything else” – is comprised of data that is usually not as easily searchable, including formats like audio, video, and social media postings.

**4) What are the stages in the data lifecycle?**

Data Sources

Data Ingestion

Data Visualization

Data Query

Data Processing

Data Storage

Data Security

Data Monitoring

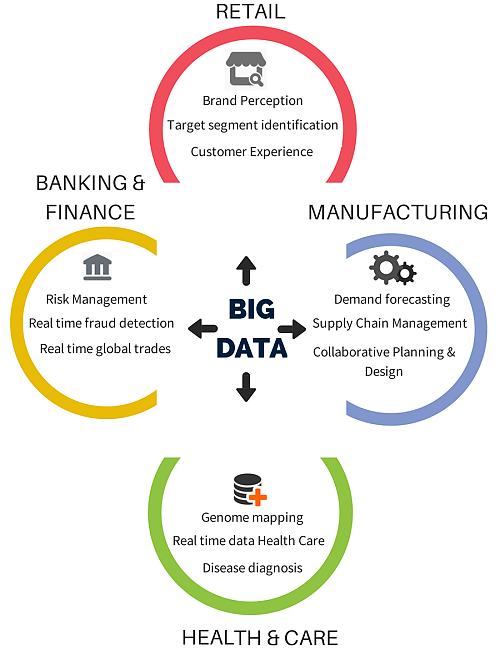
**5) What are the 5 V's of Big Data? What for each of them refer to?**

* Volume - the amount of data
* Velocity- speed of data generation
* Variety - different types of data
  + Structured
  + Unstructured
  + Quasi – Structured
  + Semi-Structured
* Veracity - reliability of data
* Value - ability to turn data into actionable insight

**6) Why Big Data?**

Companies use big data in their systems to improve operations, provide better customer service, create personalized marketing campaigns and take other actions that, ultimately, can increase revenue and profits.

**Industries using Big Data**

****

**7) What is SDLC?**

Software Development Life Cycle (SDLC) refers to a methodology with clearly defined processes for creating high-quality software.

**8) What are the phases of SDLC?**

a) Planning

b) Defining

c) Designing

d) Building

e) Testing

f) Deployment

**9) What is the Waterfall model of SDLC?**

Illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

**10) What is Agile?**

Agile is the process of breaking down the project into multiple smaller builds to focus on customer satisfaction and adaptability

**11) What is Scrum?**

A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

**12) What is the role of the Product Owner in Scrum? Scrum Master? Dev team?**

\*. The Product Owner

\*. The Development Team

\*. Scrum Master

-The **Product Owner** is the person responsible for the product backlog and is the person that everyone has to listen too.

-The **Scrum Master** is like support, they help ensure that the development team and everyone else makes sure the product stays on its appropraite path.

-The **Dev Team** is a group of developers, they all vote on decisions and there is no sole person responsible.

13) Know these terms:

- **Product backlog**- A list of new changes and features

- Sp**rint** - A short period of time usually a month of development

- **Sprint backlog** - the same as product backlog but for sprint

**- Feature** - stories that fulfill stakeholder needs

- **Sprint ceremonies** - the time period where the sprint is evaluated

- **Daily Scrum/Standup** - the work happens during a sprint

- **Sprint Retrospective** - a characteristic look to see if people themselves has endangered the Sprint and how we can fix that to improve the Sprint.

- **Sprint Review** - This looks over the Sprint to see if there was any issues and how well it has done. This is purely a meeting based on the Product Backlog

- **Sprint Planning -** the process of planning the next sprint

**14) What are some differences between hard disk and RAM?**

HDD is a permanent data storage

RAM is a fast but temporary data storage

**15) What is a VM?**

A Virtual Machine (VM) is a compute resource that uses software instead of a physical computer to run programs and deploy apps.

Virtual Machine

- Like running an OS on hardware, but on virtual "hardware"

- Resources are allocated from the actual hardware to run a VM

**16) Kernal?**

The OS controls all tasks of the system

**17) Terminal/Shell?**

The GUI that allows us to communicate with the kernel

18)

**19) What is Unix? Linux?**

Open to all

**20) What is Open Source?**

the code within the program or OS is free to use.

**21) Know the following Linux commands and options associated:**

- ls -list folders

- cd - change directory

- pwd - present working directory

- mkdir - make directory

- nano - text editor

- vim - text editor

- man - manual

- cat - concatinate

- mv - move

- cp - copy

- rm - remove

- rmdir -remove directory

- history - command history

- chmod - change permission

- head - reads the first 10 lines

- tail - reads the last 10 lines

- sort - sorts out a file beginning from special char to numbers then alphabet

**22) Difference between root and home in Linux?**

Root is the topmost level of the Linux system allowing modification of everything. Home is the base level for the user interaction with limitations set from the root.

**23) What are the different kind of permissions in Linux?**

r-read w-write x-execute

u-user g-group a-all

**24) What does yy do in Vi? dd? dw? yw? p? P?**

**yy** is used to yank/copy lines

**dd** is used to cut/delete lines

**dw** is used to delete a full word

yw is used to copy a full word

**p** is used to paste after the cursor

**P** is used to paste before/above the cursor

**25) What is XML?**

It is an extended markup language used to store data in a format easily readable by humans and machines.

**26) What is a prolog in XML? Is it necessary?**

A prolog appears prior to the root element and includes the version and text format.

**27) What is a root element? Is it necessary?**

A root element encloses all other elements as the sole-parent element. It is necessary.

**28) Does every element tag need to be closed?**

Yes

Week 2 SQL

1. **What is SQL?**
   1. Structured Query Language
2. **What is a relational database management system?**
   1. RDBMS. It is a database management system in which the database is organized and accessed according to the relationships between data items. Expressed with tables
3. **What is a database?**
   1. An organized collection of structured information or data, stored and accessed from a computer system
4. **What are the sublanguages of SQL?**
   1. Data Definition Language (DDL)
   2. Data Manipulation Language (DML)
   3. Transaction Control Language (TCL)
   4. Data Control Language (DCL)
   5. Data Query Language (DQL)
5. **What is cardinality?**
   1. > <https://en.wikipedia.org/wiki/Cardinality_(data_modeling)>
   2. Numerical relationship between rows of one table and rows in the other
   3. One-to-one, one-to-many, many-to-many
6. **What is a candidate key?**
   1. Unique key to identify a record uniquely in a table
7. **What is referential integrity?**
   1. The logical dependency of a foreign key on a primary key
8. **What are primary keys? Foreign keys?**
   1. **Primary keys** are candidate keys that are used to relate one column in a table to the rest.
   2. **Foreign keys** are used to relate rows of one table to rows of another table
9. **What are some of the different constraints on columns?**
   1. NOT NULL
   2. UNIQUE
   3. PRIMARY KEY
   4. FOREIGN KEY
   5. DEFAULT
   6. CREATE INDEX
10. **What is an entity relation diagram?**
    1. Shows the relationships of entity sets stored in a database
11. **What are the differences between WHERE vs HAVING?**
    1. HAVING can work on aggregated data.
    2. Good practice to use WHERE before GROUP BY and HAVING after GROUP BY
12. **What are the differences between GROUP BY and ORDER BY?**
    1. GROUP BY groups rows that have the same value
    2. while ORDER BY sorts
13. **What does LIKE do?**
    1. LIKE is used to search for a specific pattern in a column
14. **How do I use sub queries?**
    1. Use it as a nested query in a WHERE clause, SELECT clause, or FROM clause. Use parentheses
15. **How does BETWEEN work?** 
    1. Selects the values in a given range, inclusive on both ends
16. **What is the order of operations in an SQL statement?**
    1. The order is SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY
17. **What is the difference between an aggregate function and a scalar function?**
    1. Aggregate functions return a single summarizing value while scalar functions return a value based on scalar inputs
18. **What are examples of aggregate and scalar functions?**
    1. **Aggregate**: AVG, SUM, COUNT, MAX, MIN…
    2. **Scalar**: ROUND, FORMAT, UCASE, LCASE
19. **What are the different joins in SQL?**
    1. > We have CROSS, INNER, OUTER LEFT, OUTER RIGHT, and OUTER FULL joins.
    2. INNER joins only include records with a match in the output (so records where the join condition is true).
    3. OUTER joins includes records with a match \*and\* all unmatched records from the left, right, or both tables.
    4. > The part of the JOIN ```ON album.artist\_id = artist.artist\_id``` is the join condition. When the join condition is true for a pair of records, those records are matched together in the output. 90+% of the time, the join condition will be equality based on a foreign key relationship, but you can have various strange join conditions.
    5. A join condition of just TRUE will include all pairs of records in the output and is called a CROSS JOIN.
20. **What are the different set operations in SQL? Which set operations support duplicates?**
    1. > UNION, INTERSECT, UNION ALL are good to know. UNION combines two resultsets removing duplicates, INTERSECT produces results that appear in both of two result sets, and UNION ALL combines two resultsets including duplicates.
21. **What is the difference between joins and set operations?**
    1. Joins are used to combine columns from different tables
    2. while set operations are used to combine rows
22. **How can I create an alias in SQL?** 
    1. Keyword AS
23. **What does the AS keyword do in a query?** 
    1. Creates an alias
24. **What is a transaction?**
    1. A single unit of work (a single statement)
25. **What are the properties of a transaction?**
    1. ACID (Atomicity, Consistency, Isolation, Durability)
26. **What are the transaction isolation levels and what do they prevent?**
    1. Transaction Isolation Levels are a measure of the extent to which transaction isolation succeeds. Levels are defined as the presence or absence of these:
       1. **Dirty Reads** are similar to non-repeatable and phantom reads, but relate to reading UNCOMMITTED data, and occur when an UPDATE, INSERT or DELETE from another transaction is read, and the other transaction has NOT yet committed the data. It is reading "in progress" data, which may not be complete, and may never actually be committed.
       2. **Nonrepeatable Reads** are when your transaction reads committed UPDATES from another transaction. The same row now has different values than it did when your transaction began.
       3. Ph**antoms** are similar but when reading from committed INSERTS and/or DELETES from another transaction. There are new rows or rows that have disappeared since you began the transaction.
    2. **The levels are :**
       1. Read uncommitted
       2. Read committed
       3. Repeatable read
       4. Serializable
27. **What are dirty reads, non repeatable reads, and phantom reads?**

* **Dirty reads** are similar to non-repeatable and phantom reads, but relate to reading UNCOMMITTED data, and occur when an UPDATE, INSERT, or DELETE from another transaction is read, and the other transaction has NOT yet committed the data. It is reading "in progress" data, which may not be complete, and may never actually be committed.
* **Non-repeatable** reads are when your transaction reads committed UPDATES from another transaction. The same row now has different values than it did when your transaction began.
* Ph**antom** reads are similar but when reading from committed INSERTS and/or DELETES from another transaction. There are new rows or rows that have disappeared since you began the transaction.

1. **What is normalization?**
   1. The process of organizing the columns and tables of a database to ensure dependencies are properly enforced by database integrity constraints
      1. 1NF
         1. Each table cell should contain a single value
         2. Each record is unique
      2. 2NF
         1. Be in 1NF
         2. Single column primary key that does not functionally depend on any subset of candidate key relation
      3. 3NF
         1. Be in 2NF
         2. Have no transitive functional dependencies
2. **What are the requirements for different levels of normalization?**
   1. ^^^
3. **What is a view?**
   1. A view is like a virtual table that contains data from one or more tables. It does not physically exist in the database
4. **What is a DAO?**
   1. Native programming object model that lets you get at the heart of Access and SQL Server to create, delete, modify, and list objects, tables, fields, indexes, relations, queries, properties, and external databases
5. **What is the danger of putting values directly into our queries?**
   1. Makes database prone to injection
6. **What is multiplicity? Examples of 1-to-1, 1-to-N, N-to-N?**
   1. specifies the cardinality or number of instances of an Entity Type that can be associated with the instances of another Entity Type
7. **What is an Index?** 
   1. A quick lookup table that is used for finding records quickly in a database
8. **What advantages does creating an Index give us? Disadvantages?**
   1. Allows for quicker searching
   2. Decreases performance on inserts updates and deletes
9. **What is CRUD?**
   1. Create, Read, Update, Delete. The most basic operations that can be performed in SQL
10. **What does it mean that an operation or transaction on a data store is atomic?**
    1. It means that the entire transaction happens, or it doesn’t happen at all
11. **What does ACID stand for?**
    1. **Atomicity** – All or nothing when it comes to completion of a transaction
    2. **Consistency** – Guarantees that a transaction never leaves your database in a half-finished state
    3. **Isolation** – Keeps transactions separated until they are finished (one transaction can’t interact with another)
    4. **Durability** – Database will keep track of pending changed in such a way that the server can recover from an abnormal termination

Week 3 Python

**1) What is the difference between an interpreter and a compiler? Which does Python use? interpreter**

- https://docs.python.org/3/tutorial/interpreter.html

-----Translates just one statement of the program at a time into machine code (line by line).

-----Compiler scans the entire program and translates the whole of it into machine code at once

**2) What is a REPL?**

A Read-Eval-Print Loop, or REPL, is a computer environment where user inputs are read and evaluated, and then the results are returned to the user.

**3) What is a compound statement?**

Compound statements contain (groups of) other statements; they affect or control the execution of those other statements in some way. In general, compound statements span multiple lines, although in simple incarnations a whole compound statement may be contained in one line.

**4) What data types do we have by default in Python?**

**Numeric**

* Int
* Float
* complex

**Sequence Type**

* string
* list
* tuple

**Boolean**

* True/False

**Set**

* as in set theory

**Dictionary**

Unordered collection of data values, used to store data values like a map, which unlike other Data Types that hold only single value as an element, Dictionary holds key:value pair

**5) What is a namespace?**

a collection of currently defined symbolic names along with information about the object that each name references.

**Built**

**Global**

**Local inside the global**

**6) What is the scope of a variable?**

A variable is only available from inside the region it is created. This is called scope.

**7) What are the different operators we have in Python?**

**Arithmetic operators**

+ addition

- subtraction

\* multiplication

/ division

% modulus

\*\* explonentiation

// floor division (no remainders)

**Assignment operators**

x = 5 (assigns 5 to x)

x += 5 (assigs x+5 to x)

x -= 5 (assigs x-5 to x)

x \*= 5 (assigs x\*5 to x)

x /= 5 (assigs x/5 to x)

x %= 5 (assigs x%5 to x)

x //= 5 (assigs x//5 to x)

x \*\*= 5 (assigs x^5 to x)

x ^= 5 (assigs x^5 to x)

(???)

x &= 5 (bitwise operator for subtraction)

x |= 5 (bitwise representation of x+5)

x >>= 5 (bitwise right shift by 5 {divide by [2^5]})

x <<= 5 (bitwise left shift by 5 {multiply by [2^5]})

**Comparison operators**

== equals

!= not equal

> greater than

< less than

>= greater than OR equal to

<= less than OR equal to

**Logical operators**

AND

OR

NOT

**Identity operators**

is

is not

**Membership operators**

in

not in

**Bitwise operators**

& AND (Sets each bit to 1 if both bits are 1)

| OR (Sets each bit to 1 if one of two bits is 1)

^ XOR (Sets each bit to 1 if only one of two bits is 1)

~ NOT (Inverts all the bits)

<< Zero fill left shift (Shift left by pushing zeros in from the right and let the leftmost bits fall off)

>> Signed right shift (Shift right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off)

**8) What are functions?**

Blocks of code that execute a certain action to return something

**9) What is a lambda function? What is the syntax for creating a lambda?**

anonymous function or a function having no name. It is a small and restricted function having no more than one line.

lambda p1, p2: expression

i.e.

adder = lambda x, y: x + y

print (adder (1, 2))

**10) What is a list? a list is created by placing elements inside square brackets [] , separated by commas.**

list.append(x)

Add an item to the end of the list. Equivalent to a[len(a):] = [x].

list.extend(iterable)

Extend the list by appending all the items from the iterable. Equivalent to a[len(a):] = iterable.

list.insert(i, x)

Insert an item at a given position. The first argument is the index of the element before which to insert, so a.insert(0, x) inserts at the front of the list, and a.insert(len(a), x) is equivalent to a.append(x).

list.remove(x)

Remove the first item from the list whose value is equal to x. It raises a ValueError if there is no such item.

list.pop([i])

Remove the item at the given position in the list, and return it. If no index is specified, a.pop() removes and returns the last item in the list. (The square brackets around the i in the method signature denote that the parameter is optional, not that you should type square brackets at that position. You will see this notation frequently in the Python Library Reference.)

list.clear()

Remove all items from the list. Equivalent to del a[:].

list.index(x[, start[, end]])

Return zero-based index in the list of the first item whose value is equal to x. Raises a ValueError if there is no such item.

The optional arguments start and end are interpreted as in the slice notation and are used to limit the search to a particular subsequence of the list. The returned index is computed relative to the beginning of the full sequence rather than the start argument.

list.count(x)

Return the number of times x appears in the list.

list.sort(\*, key=None, reverse=False)

Sort the items of the list in place (the arguments can be used for sort customization, see sorted() for their explanation).

list.reverse()

Reverse the elements of the list in place.

list.copy()

Return a shallow copy of the list. Equivalent to a[:].

**11) What is a set?**

A set is an unordered collection of items.

Every set element is unique (no duplicates) and must be immutable (cannot be changed).

However, a set itself is mutable. We can add or remove items from it.

**12) What is a tuple?**

used to store multiple items in a single variable.

A tuple is a collection which is ordered and unchangeable.

Tuples are written with round brackets.

**13) What is a Dictionary?**

Dictionaries are used to store data values in key:value pairs.

A dictionary is a collection which is ordered\*, changeable and do not allow duplicates.

As of Python version 3.7, dictionaries are ordered. In Python 3.6 and earlier, dictionaries are unordered.

Dictionaries are written with curly brackets, and have keys and values:

**14) How can we open a file? What are the different modes for opening a file?**

- https://docs.python.org/3/tutorial/inputoutput.html

f = open("test.txt") # equivalent to 'r' or 'rt'

f = open("test.txt",'w') # write in text mode

f = open("img.bmp",'r+b') # read and write in binary mode

r for reading – The file pointer is placed at the beginning of the file. This is the default mode.

r+ Opens a file for both reading and writing. The file pointer will be at the beginning of the file.

w Opens a file for writing only. Overwrites the file if the file exists. If the file does not exist, creates a new file for writing.

w+ Opens a file for both writing and reading. Overwrites the existing file if the file exists. If the file does not exist, it creates a new file for reading and writing.

rb Opens a file for reading only in binary format. The file pointer is placed at the beginning of the file.

rb+ Opens a file for both reading and writing in binary format.

wb+ Opens a file for both writing and reading in binary format. Overwrites the existing file if the file exists. If the file does not exist, it creates a new file for reading and writing.

a Opens a file for appending. The file pointer is at the end of the file if the file exists. That is, the file is in the append mode. If the file does not exist, it creates a new file for writing.

ab Opens a file for appending in binary format. The file pointer is at the end of the file if the file exists. That is, the file is in the append mode. If the file does not exist, it creates a new file for writing.

a+ Opens a file for both appending and reading. The file pointer is at the end of the file if the file exists. The file opens in the append mode. If the file does not exist, it creates a new file for reading and writing.

ab+ Opens a file for both appending and reading in binary format. The file pointer is at the end of the file if the file exists. The file opens in the append mode. If the file does not exist, it creates a new file for reading and writing.

x open for exclusive creation, failing if the file already exists (Python 3)

**15) What is a module? How do we import a module?**

- https://docs.python.org/3/tutorial/modules.html

A module is a file containing Python definitions and statements. The file name is the module name with the suffix .py appended. Within a module, the module’s name (as a string) is available as the value of the global variable \_\_name\_\_.

**16) What is a datetime object?**

The datetime module supplies classes for manipulating dates and times.

Date and time objects may be categorized as “aware” or “naive” depending on whether or not they include timezone information.

**17) What are Regular Expressions? How do we use them?**

A RegEx, or Regular Expression, is a sequence of characters that forms a search pattern.

RegEx can be used to check if a string contains the specified search pattern.

import re

txt = "The rain in Spain"

x = re.search("^The.\*Spain$", txt)

**18) What are some data collections we have at our disposal in the Collections module?**

DEFAULTDICT is exactly like a dictionary in python. The only difference is that it does not give an exception/key error when you try to access the non-existent key.

COUNTER is a built-in data structure which is used to count the occurrence of each value present in an array or list.

DEQUE is an optimal version of list used for inserting and removing items. It can add/remove items from either start or the end of the list.

The NAMEDTUPLE() solves a very major problem in the field of computer science. Usual tuples need to remember the index of each field of a tuple object, however, namedtuple() solves this by simply returning with names for each position in the tuple.

CHAINMAP combines a lot of dictionaries together and returns a list of dictionaries. ChainMaps basically encapsulates a lot of dictionaries into one single unit with no restriction on the number of dictionaries.

ORDEREDDICT is a dictionary that ensures its order is maintained. For example, if the keys are inserted in a specific order, then the order is maintained. Even if you change the value of the key later, the position will remain the same.

**19) What is a class? What is an object?**

- <https://docs.python.org/3/tutorial/classes.html>

Classes provide a means of bundling data and functionality together. Creating a new class creates a new type of object, allowing new instances of that type to be made. Each class instance can have attributes attached to it for maintaining its state. Class instances can also have methods (defined by its class) for modifying its state.

Objects have individuality, and multiple names (in multiple scopes) can be bound to the same object. This is known as aliasing in other languages.

**20) What are the pillars of Object Oriented Programming? (OOP) Explain them.**

- https://medium.com/swlh/the-4-pillars-of-oop-in-python-9daaca4c0d13

Encapsulation (to not let outside effects impact that code.)

Inheritance (share some things in common)

Abstraction (hiding the real implementation of an application from the user and emphasizing only how to use the application)

Polymorphism (same function name is used for different types)

**21) Know the different flow control statements (if/elif/else, for loops, while loops, etc.)**

- https://docs.python.org/3/tutorial/controlflow.html

if > (elif)\* > else

BREAK (terminates the current loop and resumes execution at the next statement)

CONTINUE (stop the current iteration of the loop, and continue with the next)

PASS (nothing happens, but you avoid getting an error when empty code is not allowed.)

for x in y:

if a == b :

c

for i in range:

x

**22) What are exceptions? How can we handle them?**

- https://docs.python.org/3/tutorial/errors.html

Errors detected during execution

TRY:

xyz

raise EXCEPTION(abc)

EXCEPT:

(catches specific exception and executes code if said exception is caught)

All exceptions inherit from BASE EXCEPTION

except BaseException

**23) CRUD:**

-CREATE

-READ

-Update

-Delete

**24) ACID:**

**Atomicity**: all the modified data are performed or none of them will be

**Consistency**: maintains data integrity constraints, leaving the data consistent.

**Isolation**: ensuring that the transaction will not be changed by any other concurrent transaction

**Durability**: changes are persisted permanently in the database

**25) ERD**

stands for entity relationship diagram. Objects of type dj. ERD allow visualizing portions of the data pipeline in graphical form. Tables are depicted as nodes and dependencies as directed edges between them. The draw method plots the graph.

Week 4-MongoDB

1. **What does BASE stand for?**
   1. Basic Availability, Soft-state, Eventual consistency
2. **What is a database in Mongo?**
   1. A database in Mongo is a grouping of unstructured data in JSON format
3. **What is a collection?**
   1. Equates to a table
4. **What is a document?** 
   1. Equates to a row
5. **What rules does Mongo enforce about the structure of documents inside a collection?**
6. **What is a distributed application? A distributed data store?**
7. **What is High Availability? How is it achieved in Mongo?**
   1. High availability is the ability of a system to operate continuously without failing for a designated period of time. Mongo achieves it through replica sets
8. **What is Scalability? How is it achieved in Mongo?**
   1. Scalability is the ability of a database to grow in size. It is achieved both vertically and horizontally in MongoDB. Horizontally through sharding and replica sets
9. **Explain replica sets and sharding?**

**Replica Set** is a group of mongod processes that maintain the same data set. Replica sets provide redundancy and high availability, and are the basis for all production deployments. This section introduces replication in MongoDB as well as the components and architecture of replica sets.

**Sharding** is the process of storing data records across multiple machines and it is MongoDB's approach to meeting the demands of data growth. As the size of the data increases, a single machine may not be sufficient to store the data nor provide an acceptable read and write throughput. Sharding solves the problem with horizontal scaling. With sharding, you add more machines to support data growth and the demands of read and write operations.

1. **What are NoSQL databases? What are the different types of NoSQL databases?**
   1. NoSQL is a type of nonrelational database query language.
   2. Wide column, document, key-value, graph
2. **What kind of NoSQL database MongoDB is?**
   1. It is a document database
3. **Which are the most important features of MongoDB?**
   1. Can handle unstructured data
4. **What is a Namespace in MongoDB?**
   1. The name of the collection, including the database.
5. **Which all languages can be used with MongoDB?**
6. **Compare SQL databases and MongoDB at a high level.**
   1. SQL databases are structured while MongoDB is used to save unstructured data
7. **How is MongoDB better than other SQL databases?**
   1. It is much faster than other SQL databases
   2. Much more scalable
   3. More flexible
8. **Does MongoDB support foreign key constraints?**
   1. No
9. **Does MongoDB support ACID transaction management and locking functionalities?**
   1. YES, but not innately
10. **How can you achieve primary key - foreign key relationships in MongoDB?**
    1. By embedding a document inside another
11. **When should we embed one document within another in MongoDB?**
    1. When we want to relate the two documents
12. **Mention what is Objecld composed of?**
    * 1. 4-byte timestamp value representing object’s creation
      2. 5-byte random value
      3. 3-byte incrementing c