W4111 – Introduction to Databases
Sections 002, V002; spring 2022

Homework 1 – Written Assignment

Instructions

- The homework submission date/time is 06-Feb-2022 at 11:59 PM.
- Submission format is a PDF version of this document with your answers. Place your answers in the document after the questions.
- The name of your PDF must be <UNI>_S22_W4111_HW1_Written.pdf. For example, mine would be dff9_S22_W4111_HW1_Written.pdf
- You must use the Gradescope functions to mark the location of your questions/answers in the submitted PDF. Failure to mark pages will cause point deductions.
- You can use online sources but you must cite your sources. You may not cut and paste text..
- Questions typically require less than five sentences for an answer. You will lose points if your answer runs on and wanders.

  "Verbosity wastes a portion of the reader's or listener's life."

Questions

Question 1: Briefly explain the terms *structured data, semi-structured data* and *unstructured data.* Give an example of each type.

Structured Data: Data that is easy to search and organize since it can be converted to a tabular format with all of its elements mapped to pre-defined domains. Example: Bank transactions (sender, receiver, amount, memo, etc.)

Unstructured Data: Data that cannot be fit into a tabular format and doesn't have an associated data model. Example: Image, Audio datasets.

Semi-Structured Data: In between structured and unstructured data, this data has some definition or characteristics but is not structured enough to fit into a database structure. Example: Email (from, to, subject, date/time etc.) however the content of the email itself is unstructured (may contain images, attachments etc.)

Ref: https://www.forbes.com/sites/bernardmarr/2019/10/18/whats-the-difference-between-structured-semi-structured-and-unstructured-data/

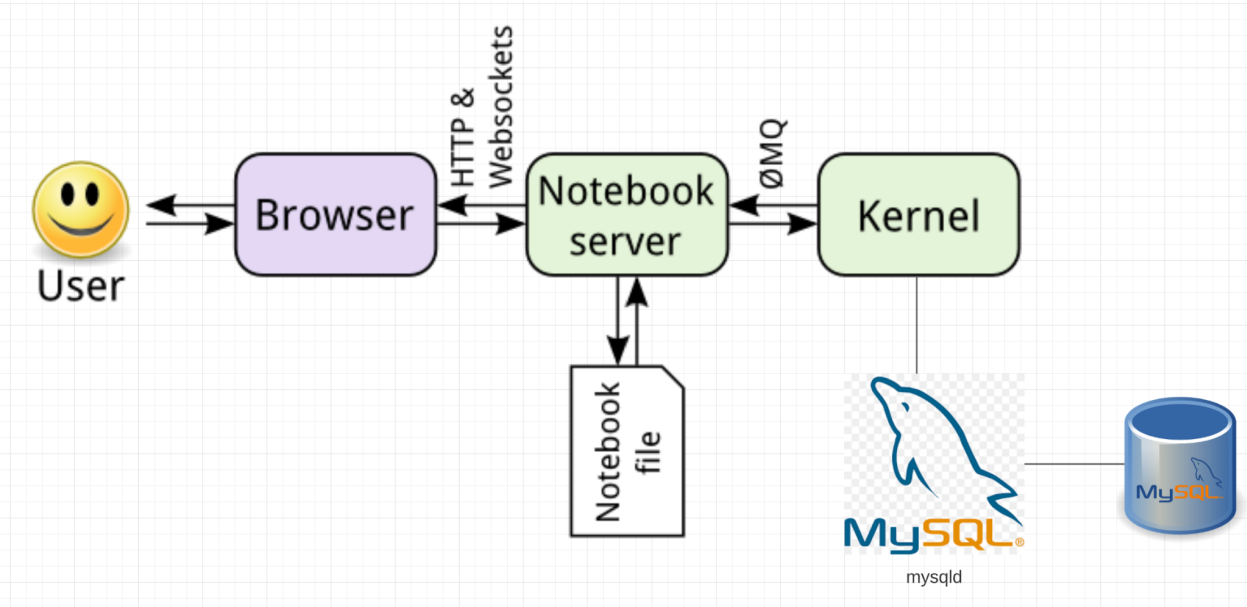
Question 2: Briefly explain the concept of *metadata*. For a presentation (PowerPoint, Google Slides), what would be some examples of metadata?

Metadata is data that provides information about other data but not about its contents directly. It means "data about data".

For a presentation (Google Slides/PowerPoint) some metadata might be number of slides, author, date created/modified, resolution of slide, orientation of slides, animation/transition information etc.

Ref: https://en.wikipedia.org/wiki/Metadata

Question 3: The following diagram is an overview of Jupyter Notebook's runtime model when the notebook is using MySQL. Is this a 2-tier application or 3-tier application? Briefly explain why.



A 3-tier application, the application logic or process is separate from the data and the user interface. In a 2-tier application, it is merged within one of the other two layers.

In this diagram about Jupyter Notebooks, the process is in the notebook server (python server) and the data lives in the Kernel (MySQL) and the UI is in the browser. Hence it is a 3-tier application.

Ref: https://www.geeksforgeeks.org/difference-between-two-tier-and-three-tier-database-architecture/

Question 4: Briefly define and explain procedural and declarative languages. Is SQL procedural or declarative?

A procedural language the entire problem is defined and the steps to implement it are provided. Essentially you provide commands and the order and method of execution of these commands.

In declarative languages, only the commands are provided, and it is left to the system to decide the method or process of execution. Hence, these languages are concerned with the result of a command while abstracting the process of its execution.

SQL is a declarative language, as we only define the commands (SELECT, JOIN, etc.) but do not define these processes or their implementations ourselves.

Ref: https://stackoverflow.com/questions/1619834/what-is-the-difference-between-declarative-and-procedural-programming-paradigms

Question 5: List 4 advantages/differences of database management systems (DBMS) compared to programs and files for data processing. List two disadvantages of DBMS?

Advantages of DBMS:

Data redundancy and inconsistency: There is a singular source of data in a DBMS system, leading to changes made by a user reflected to all other users. Hence there is no inconsistency of the data, moreover, there is no need to keep multiple copies of the data, leading to a removal of redundancy.

Data concurrency: File systems and programs require special procedures to ensure multiple users can seamlessly access the same data (without loss of information)

Disadvantages of DBMS:

Database Failure: Since there is a singular source of data, any failure of the database will lead to the loss of the data.

High Cost: Database solutions require high performance hardware and specialized software, which leads to a high cost of setup and maintenance. Moreover, they also require trained staff to manage and maintain the databses, leading to additional costs.

Ref: https://www.geeksforgeeks.org/advantages-of-dbms-over-file-system/
https://www.tutorialandexample.com/disadvantages-of-dbms/

Question 6: In a relational DBMS, columns/attributes should be *atomic.* Briefly explain what this means. If a table has a column *name* of the form "last name, first name", is this atomic?

Column/attributes should be atomic (indivisible) in a relational DBMS. That means that any columns/values should not be divisible into two or more columns/values.

No, a table with a column name of the form "last name, first name" is not atomic as this can be split into two columns with "last name" and "first name" respectively.

Question 7: Attributes/columns have *types,* e.g. int, varchar(128), timestamp. An attribute/column values must be from a *domain*? What is the difference between a type and a domain (hint: domain constraints)?

A datatype defines what types of values can be stored in a column (strings, dates, integers etc.). A domain defines exactly what values can be stored (for example a integer column with all positive values or a date column with all dates before a fixed date can be implemented using a domain constraint)

Ref: https://www.quora.com/What-is-the-difference-between-domain-and-data-type-in-DBMS

Question 8: There are four common types of people that interact with a database management system. List and briefly explain each of the four types.

Database Administrators:
They administer the access to the database, coordinate and monitor its usage and manage software and hardware usage for its continuous support

Database Designers:
They design the database, creating the data structures for storing and representing this data, and identifying the types and kind of data to be stored.

End Users:
They are the main users of the database, using it for running queries or for updating the data.

Software Engineers:
They create and maintain special software and applications for the end users, and also store and document certain "canned transactions" that are frequently performed by end users.

Ref: https://www.geeksforgeeks.org/personnel-involved-in-database-management-system/

Question 9: Briefly explain the concepts of database *instance* and *schema*?

Database Schema: It is the logical structure of the database with the table and column names and types
Database Instance: It is a snapshot of the data in the database at a fixed instance of time.

Example:

Schema: instructor (ID, name, dept_name, salary)

Instance:

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

Question 10: Explain the concept of *physical data independence* and the importance of the concept.

Physical data independence is the property of a DBMS so that any changes made in the structure of the lowest level of the database (file system) doesn't affect any of the higher levels (logical/view).

This is important as any changes to the physical level (creating a file, defining a new index, changing the hardware) doesn't affect the higher levels of a DBMS and hence the database is preserved

Ref: https://www.geeksforgeeks.org/physical-and-logical-data-independence/