1.  List four applications you have used that most likely employed a database system to store persistent data.  Give specific examples of what kind of information was stored in that database.

1) KASPI:  For account information, transfer of funds, banking transactions.

2) WSP: For student information, online assignment submissions, course registrations, and grades.

3) AirAstana:  For reservation of tickets, and schedule information.

4) Informburo, Tengrinews:    For updating new, maintenance of archives.

2. List four significant differences between a file-processing system and a DBMS.

|  |  |
| --- | --- |
| Used to manage and organise the files stored in the hard disk of the computer | A software to store and retrieve the user’s data |
| Redundant data is present | No presence of redundant data |
| Query processing is not so efficient | Query processing is efficient |
| Data consistency is low | Due to the process of normalisation, the data consistency is high |
| Less complex, does not support complicated transactions | More complexity in managing the data, easier to implement complicated transactions |

3. Explain the concept of physical data independence, and its importance in database systems.

Physical data independence is the ability to modify the physical scheme without making it necessary to rewrite application programs. Such modifications include changing from unblocked to blocked record storage, or from sequential to random access files. Such a modification might be adding a field to a record; an application program’s view hides this change from the program.

4. List five responsibilities of a database management system.  For each responsibility, explain the problems that would arise if the responsibility were not discharged.

**a. interaction with the file manager.**

No DBMS can do without this. If there is no file manager interaction then nothing stored in the files can be retrieved.

**b. integrity enforcement.**

Consistency constraints may not be satisfied, for example an instructor may belong to a non-existent department, two students may have the same ID, account balances could go below the minimum allowed, and so on.

**c. security enforcement.**

Unauthorized users may access the database, or users authorized to access part of the database may be able to access parts of the database for which they lack authority. For example, a low-level user could get access to national defense secret codes, or employees could find out what their supervisors earn (which is presumably a secret).

**d. backup and recovery.**

Data could be lost permanently, rather than at least being available in a consistent state that existed prior to a failure.

**e. concurrency control.**

Consistency constraints may be violated despite proper integrity enforcement in each transaction. For example, incorrect bank balances might be reflected due to simultaneous withdrawals and deposits on the same account, and so on.

**1.11 – 1.15**

The component of the **database**that prevents both students from getting the **last seat**is: **transaction isolation**

**Data**that enters the **database**are expected to maintain **accuracy**and also be **consistent**with the **database structure**.

So, when both students request for the **last seat**, the possibilities are:

* *Student A gets the seat*
* *Student B gets the seat*

The following is not a possibility

* *Both students get the seat*
* *None of the students gets the seat*

The above highlights means that, **only one**of the students would get the **seat.**

This is possible because of the concept called **transaction isolation.**

The **transaction isolation**ensures that the **data**requested by a user is *complete*and such data maintains *competency*.

So, when a **student**gets the **last seat**, the *next student*would not get the **same seat**(*or any other seat*), because a **transaction**has already been completed

2

ANSWER: 1. 2Tier is Client server architecture and 3Tier is Client, Server and Database architecture. 2. 3Tier has a Middle stage to communicate client to server, Where as in 2Tier client directly get communication to server. 3. 3Tier is like a MVC, But having difference in topologies 4. 3Tier is linear means in that request flow is Client>>>Middle Layer(SErver application) >>>Databse server and Response is reverse. While in 2Tier it a Triangular View >>Controller>>Model 5. 3Tier is like Website while web browser is Client application(middle layer), and ASP/PHP language code is server application.

3

Answer:

• To backup data

• In some cases, to create the schema definition

• To define the storage structure and access methods

• To modify the schema and/or physical organization when necessary

• To grant authorization for data access

• To specify integrity constraints

4

The only way to increase the capacity of these databases was to upgrade the servers – processors, memory, and storage – to scale up.**NoSQL** databases **emerged**as a result of the exponential growth of the internet and the rise of web applications.

* **SQL**pronounced as "S-Q-L" or as "See-Quel" is primarily called RDBMS or Relational Databases whereas**NoSQL**is a Non-relational or Distributed Database.
* SQL databases are table based databases whereas NoSQL databases can be document based, key-value pairs, graph databases.
* SQL databases are vertically scalable while NoSQL databases are horizontally scalable.
* SQL databases have a predefined schema whereas NoSQL databases use dynamic schema for unstructured data.
* SQL requires specialized DB hardware for better performance while NoSQL uses commodity hardware.

5.

**a**) A users table containing users, with attributes such as account name, real name, age, gender, location, and other profile information

**b.** A content table containing user provided content, such as text and

images, associated with the user who uploaded the content.

**c**. A friends table recording for each user which other users are connected to that user. The kind of connection may also be recorded

in this table.

**d**. A permissionstable, recording which category of friends are allowed

to view which content uploaded by a user. For example, a user may

share some photos with family but not with all friends.