Review Questions

Question 1. Define each of the following terms:

a. data

b. field

c. record

d. file

Answer:

a. Data: Raw facts that have not yet been processed or organized, such as numbers, words, or images.

b. Field: A single piece of data within a record, such as a name, date, or phone number.

c. Record: A complete set of related fields, representing a single entity, such as a person or an order.

d. File: A collection of records, often stored in a database or a spreadsheet.

Question 2. What is data redundancy, and which characteristics of the file system can lead to it?

Answer:

Data redundancy occurs when the same data is stored in multiple places. It can arise in a file system due to the lack of centralized control, leading to duplicate data across different files or applications.

Question 3. What is data independence, and why is it lacking in file systems?

Answer:

Data independence refers to the ability to change the data structure without affecting the applications that use the data. In traditional file systems, data is tightly coupled with the application, making it difficult to change the data structure without impacting the entire system.

Question 4. What is a DBMS, and what are its functions?

Answer:

A Database Management System (DBMS) is software that facilitates the creation, management, and manipulation of databases. Its functions include data storage, retrieval, updating, and administration of databases, ensuring data security, integrity, and consistency.

Question 5. What is structural independence, and why is it important?

Answer:

Structural independence is the capacity to change the database structure without affecting the application's ability to access data. It's important because it allows for flexibility in managing data without disrupting business operations.

Question 6. Explain the differences between data, information, and a database.

Answer:

Data is raw, unprocessed facts. Information is data that has been processed and organized to be meaningful and useful. A database is an organized collection of data that allows for efficient retrieval and manipulation.

Question 7. What is the role of a DBMS, and what are its advantages? What are its disadvantages?

Answer:

The role of a DBMS is to manage and control access to data in a database. Its advantages include data integrity, security, and support for concurrent users. Disadvantages can include complexity, cost, and the potential for performance bottlenecks.

Question 8. List and describe the different types of databases.

Answer:

Types of databases include:

- **Relational Databases**: Use tables to organize data, with relationships between them.
- **NoSQL Databases**: Designed for unstructured or semi-structured data, offering flexibility.
- **Hierarchical Databases**: Organize data in a tree-like structure, suitable for applications with clear parent-child relationships.
- **Object-Oriented Databases**: Store data as objects, similar to object-oriented programming languages.

Question 9. What are the main components of a database system?

Answer:

The main components include the hardware, software (DBMS), data, procedures, and database users.

Question 10. What is metadata?

Answer:

Metadata is data that describes other data, such as the structure, constraints, and relationships of data within a database.

Question 11. Explain why database design is important.

Answer:

Database design is crucial because it ensures the database is structured efficiently and supports the necessary operations while maintaining data integrity and consistency.

Question 12. What are the potential costs of implementing a database system?

Answer:

Costs may include hardware, software, personnel for design and maintenance, training, and ongoing operational expenses.

Question 13. Use examples to compare and contrast unstructured and structured data. Which type is more prevalent in a typical business environment?

Answer:

Structured data is organized in a fixed format, like tables in a database (e.g., customer information). Unstructured data has no predefined structure, like emails or social media posts. Structured data is more prevalent in a business environment due to its ease of analysis.

Question 14. What are some basic database functions that a spreadsheet cannot perform?

Answer:

A database can handle large volumes of data, enforce data integrity, support multiple users simultaneously, and manage complex queries, which spreadsheets are not well-equipped to handle.

Question 15. What common problems do a collection of spreadsheets created by end users share with the typical file system?

Answer:

Both can suffer from data redundancy, inconsistency, lack of data integrity, and difficulty in managing and updating data.

Question 16. Explain the significance of the loss of direct, hands-on access to business data that end users experienced with the advent of computerized data repositories.

Answer:

The loss of direct access can lead to a dependency on IT departments, slow down data retrieval processes, and reduce users' ability to quickly adapt data to their needs.

Question 17. Explain why the cost of ownership may be lower with a cloud database than with a traditional, company database.

Answer:

Cloud databases typically offer lower costs due to reduced need for physical infrastructure, lower maintenance requirements, and flexible, pay-as-you-go pricing models.

Problems

1. The File Structure for Questions 1 - 4: Project Information

PROJECT_CODE	PROJECT_MANAGER	MANAGER_PHONE	MANAGER_ADDRESS	PROJECT_BID_PRICE
21-5Z	Holly B. Parker	904-338-3416	3334 Lee Rd., Gainesville, FL 37123	16833460.00
25-2D	Jane D. Grant	615-898-9039	218 Clark Blvd., Nashville, TN 36362	12500000.00
25-5A	George F. Dorts	615-227-1245	124 River Dr., Franklin, TN 29185	32512407.00
27-4Q	George F. Dorts	615-227-1245	124 River Dr., Franklin, TN 29185	10314545.00
29-2D	Holly B. Parker	904-338-3416	3334 Lee Rd., Gainesville, FL 37123	25559900.00
31-7P	William K. Moor	904-445-2719	216 Morton Rd., Stetson, FL 30155	56850000.00

Question 1. How many records does the file contain? How many fields are there per record?

Answer:

- The file contains 6 records.
- There are 5 fields per record: PROJECT_CODE, PROJECT_MANAGER, MANAGER_PHONE, MANAGER_ADDRESS, and PROJECT_BID_PRICE.

Question 2. What problem would you encounter if you wanted to produce a listing by city? How would you solve this problem by altering the file structure?

Answer:

- **Problem:** The MANAGER_ADDRESS field combines multiple pieces of information (street address, city, state, and ZIP code) into a single field. This makes it difficult to sort or filter the records by city alone.

- **Solution:** Split the MANAGER_ADDRESS field into separate fields for STREET_ADDRESS, CITY, STATE, and ZIP_CODE. This would allow easy sorting or filtering by any of these components, including the city.

Question 3. If you wanted to produce a listing of the file contents by last name, area code, city, state, or zip code, how would you alter the file structure?

Answer:

- To sort by last name: Split the PROJECT_MANAGER field into MANAGER_FIRST_NAME and MANAGER_LAST_NAME.

- To sort by area code: Split the MANAGER_PHONE field into PHONE_AREA_CODE and PHONE_NUMBER.

- For city, state, or ZIP code: As mentioned in Question 2, split the MANAGER_ADDRESS field into STREET_ADDRESS, CITY, STATE, and ZIP_CODE.

Question 4. What data redundancies do you detect? How could those redundancies lead to anomalies?

Answer:

- **Redundancies:** The MANAGER_ADDRESS and MANAGER_PHONE fields are repeated for each project managed by the same person. For example, George F. Dorts appears twice with the same address and phone number.

- **Potential Anomalies:** If the address or phone number for a manager changes, it would need to be updated in multiple places. Failure to update all instances could lead to inconsistent data (e.g., some records with the old address and some with the new one).

- **Solution:** Normalize the data by creating a separate table for MANAGERS, where each manager's information (name, phone, address) is stored only once. The PROJECTS table would then reference the MANAGERS table using a MANAGER_ID key. This reduces redundancy and minimizes the risk of anomalies.

2. The File Structure for Questions 5 - 8: Project Employee Information

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CODE	JOB_CHG_HOUR	PROJ_HOURS	EMP_PHONE
1	Hurricane	101	John D. Newson	EE	85.00	13.3	653-234-3245
1	Hurricane	105	David F. Schwann	CT	60.00	16.2	653-234-1123
1	Hurricane	110	Anne R. Ramoras	CT	60.00	14.3	615-233-5562
2	Coast	101	John D. Newson	EE	85.00	19.8	653-234-3245
2	Coast	108	June H. Sattlemier	EE	85.00	17.5	954-554-7812
3	Satellite	110	Anne R. Ramoras	CT	60.00	11.6	615-233-5562
3	Satellite	105	David F. Schwann	CT	60.00	23.4	653-234-1123
3	Satellite	123	Mary D. Chen	EE	85.00	19.1	615-233-5432
3	Satellite	112	Allecia R. Smith	BE	85.00	20.7	615-678-6879

Question 5. Identify and discuss the serious data redundancy problems exhibited by the Project Employee Information Table.

Answer:

- **Redundancy Problems:**
  - **Employee Information Repetition:** The EMP_NUM, EMP_NAME, and EMP_PHONE fields are repeated for each project an employee works on. For example, John D. Newson appears twice with the same employee number and phone number, but for different projects (Hurricane and Coast).
  - **Job Information Repetition:** The JOB_CODE and JOB_CHG_HOUR fields are also repeated for employees who have the same job across different projects. For instance, David F. Schwann has the same JOB_CODE and JOB_CHG_HOUR across two different projects.

- **Issues Caused by Redundancy:**
  - **Data Inconsistency:** If an employee's information (e.g., phone number) changes, it must be updated in every record where the employee appears. If not updated consistently, this can lead to data discrepancies.
  - **Increased Storage Requirements:** Storing redundant data increases the file size unnecessarily, which can be problematic for large datasets.
  - **Update Anomalies:** Updating redundant data increases the chance of introducing errors, such as forgetting to update all instances of an employee's information.

Question 6. Looking at the EMP_NAME and EMP_PHONE contents in the Project Employee Information Table, what change(s) would you recommend?

Answer:

Recommended Changes:

- **Normalization:** Create a separate table for employee information that includes EMP_NUM, EMP_NAME, and EMP_PHONE. The Project Employee Information Table would then reference this new EMPLOYEES table using EMP_NUM as a foreign key.

- **Benefits of the Change:** This change would eliminate the redundancy of storing EMP_NAME and EMP_PHONE multiple times, reduce the risk of inconsistencies, and make it easier to update employee information.

Question 7. Identify the various data sources in the file you examined in Problem 5.

Answer:

Data Sources Identified:

- **Project Information:** The fields PROJ_NUM and PROJ_NAME relate to specific projects.
- **Employee Information:** The fields EMP_NUM, EMP_NAME, and EMP_PHONE pertain to employees.
- **Job Information:** The fields JOB_CODE and JOB_CHG_HOUR describe the job roles and corresponding hourly charges.
- **Work Assignment Information:** The fields PROJ_HOURS capture the number of hours each employee worked on a project.

Question 8. Given your answer to Problem 7, what new files should you create to help eliminate the data redundancies found in the file shown in the Project Employee Information Table?

Answer:

- **Suggested New Files:**
  - **EMPLOYEES Table:**
    - Fields: EMP_NUM, EMP_NAME, EMP_PHONE.
    - Purpose: Store each employee's details uniquely, eliminating redundancy across projects.
  - **PROJECTS Table:**
    - Fields: PROJ_NUM, PROJ_NAME.
    - Purpose: Store details about each project uniquely, preventing redundancy.
  - **JOBS Table:**
    - Fields: JOB_CODE, JOB_DESCRIPTION, JOB_CHG_HOUR.
    - Purpose: Store job-related information, preventing redundant job data across different projects.
  - **ASSIGNMENTS Table:**
    - Fields: PROJ_NUM, EMP_NUM, JOB_CODE, PROJ_HOURS.
    - Purpose: Record the work assignment details by linking PROJ_NUM to EMP_NUM and JOB_CODE, capturing the unique work hours per project without redundancy.

By normalizing the data across these new tables, you can eliminate redundancies, simplify updates, and ensure data integrity.

3. The File Structure for Questions 9 - 10: Teacher Schedule

BUILDING_CODE	ROOM_CODE	TEACHER_LNAME	TEACHER_FNAME	TEACHER_INITIAL	DAYS_TIME
KOM	204E	Williston	Horace	G	MWF 8:00-8:50
KOM	123	Cordoza	Maria	L	MWF 8:00-8:50
LDB	504	Patroski	Donald	J	TTh 1:00-2:15
KOM	123	Hawkins	Anne	J/V	MWF 10:00-10:50
JKP	225D	Risell	James		TTh 9:00-10:15
LDB	301	Robertson	Jeanette	P	TTh 1:00-2:15
KOM	204E	Cordoza	Maria	L	MWF 9:00-9:50
LDB	504	Williston	Horace	G	TTh 11:00-12:15
KOM	344	Cordoza	Maria	L	MWF 11:00-11:50
LDB	504	Patroski	Donald	J	MWF 2:00-2:50

Question 9. Identify and discuss the serious data redundancy problems exhibited by the file structure shown in the Teacher Schedule Table. (The file is meant to be used as a teacher class assignment schedule. One of the many problems with data redundancy is the likely occurrence of data inconsistencies—two different initials have been entered for the teacher named Maria Cordoza.)

Answer:

- **Redundancy Problems:**
  - **Teacher Information Repetition:** The TEACHER_LNAME, TEACHER_FNAME, and TEACHER_INITIAL fields are repeated for each class a teacher is assigned to. For example, Maria L. Cordoza appears multiple times across different classes and rooms.
  - **Data Inconsistency:** As indicated in the problem, there is an inconsistency in the TEACHER_INITIAL field for Maria Cordoza. In one record, her initials are listed as "L", while in another, they are "J/V". This could lead to confusion and errors in identifying the correct teacher.
  - **Scheduling Data Repetition:** The BUILDING_CODE, ROOM_CODE, and DAYS_TIME fields are repeated for every teacher who has a class at the same time and location. For instance, both Horace Williston and Maria Cordoza have classes in KOM 204E at MWF 8:00-8:50, resulting in redundancy.

- **Issues Caused by Redundancy:**
  - **Inconsistent Data:** As shown by the different initials for Maria Cordoza, redundant data can lead to inconsistencies, where the same entity is represented differently in various records.
  - **Increased Data Storage:** Repeating information like teacher names and class times for each schedule entry unnecessarily increases the file size.
  - **Maintenance Challenges:** Updating information for a teacher or a class location must be done in multiple places, increasing the risk of errors.

Question 10. Given the file structure shown in the Teacher Schedule Table what problem(s) might you encounter if building KOM were deleted?

Answer:

Problems Encountered:

- **Data Loss:** If the KOM building were deleted from the file, all the records associated with KOM would be lost, leading to the loss of not only the building information but also the associated class schedules, teacher assignments, and room information.
- **Unintended Cascade Effects:** Deleting the KOM building could result in the accidental deletion of valid class schedules and teacher assignments, which could disrupt the entire scheduling system.
- **Orphaned Records:** If the deletion of KOM is not handled correctly, it could leave orphaned records in the system, where references to KOM still exist but the building itself does not, causing data integrity issues.

4. Student Schedule Data Format

STU_ID	STU_NAME	CLASS_CODE	CLASS_NAME	CLASS_CREDHRS	INSTR_NAME	CLASS_DAYS	CLASS_TIMES	ROOM

Question 1. Create a spreadsheet using the template shown in the Student Schedule Data Format Table and enter your current class schedule.

Answer:

You can create a spreadsheet with the following columns, based on the provided data format:

STU_ID	STU_NAME	CLASS_CODE	CLASS_NAME	CLASS_CREDHRS	INSTR_NAME	CLASS_DAYS	CLASS_TIMES	ROOM
1	YourName	ENG101	English I	3	Dr. Smith	MWF	9:00-10:00	Room 101
1	YourName	MTH202	Calculus II	4	Dr. Jones	TTh	10:00-11:30	Room 202
1	YourName	HIS303	World History	3	Dr. Brown	MWF	11:00-12:00	Room 303

Question 2. Enter the class schedule of two of your classmates into the same spreadsheet.

Answer:

For Classmate 1:

STU_ID	STU_NAME	CLASS_CODE	CLASS_NAME	CLASS_CREDHRS	INSTR_NAME	CLASS_DAYS	CLASS_TIMES	ROOM
2	Alice	ENG101	English I	3	Dr. Smith	MWF	9:00-10:00	Room 101
2	Alice	CHE102	Chemistry	4	Dr. Adams	TTh	10:00-11:30	Room 204
2	Alice	HIS303	World History	3	Dr. Brown	MWF	11:00-12:00	Room 303

For Classmate 2:

STU_ID	STU_NAME	CLASS_CODE	CLASS_NAME	CLASS_CREDHRS	INSTR_NAME	CLASS_DAYS	CLASS_TIMES	ROOM
3	Bob	ENG101	English I	3	Dr. Smith	MWF	9:00-10:00	Room 101
3	Bob	MTH202	Calculus II	4	Dr. Jones	TTh	10:00-11:30	Room 202
3	Bob	PHY301	Physics I	3	Dr. Taylor	MWF	1:00-2:00	Room 205

Question 3. Discuss the redundancies and anomalies caused by this design.

Answer:

Redundancies:

1. **Instructor Information Repeated:** The instructor's name, class days, times, and room number are repeated for each student in the same class.
2. **Class Information Repeated:** The class code, name, and credit hours are also repeated for each student in the same class.
3. **Room Information Repeated:** The room number for the class is repeated for every student taking that class.

Anomalies:

1. **Update Anomalies:** If the instructor's name or class time changes, it must be updated in multiple rows for every student in that class. If any row is missed, it leads to inconsistency.
2. **Deletion Anomalies:** If a student drops a class and their row is deleted, information about the class, instructor, or room might be lost if that was the last student in the class.
3. **Insertion Anomalies:** Adding a new class or instructor requires adding all the relevant details (class name, instructor, room, etc.) for each new student, leading to potential errors or inconsistencies.

This design leads to a lot of redundancy and is prone to inconsistencies, which is why normalization and separating data into related tables (such as students, classes, and instructors) is usually preferred.