**NAME: KEVIN KEPHER OTIENO**

**Part 1: Understanding SQL (30 minutes)**

**Question 1. Research**

1. **In a single Word document, summarize your findings in a short paragraph (3-5 sentences). Web Applications:**

**Imagine a dynamic website like an online store. How do you think SQL plays a role in managing data behind the scenes? Consider how product information, user accounts, and order details might be stored and accessed.**

# **Understanding SQL in Web Applications**

SQL (Structured Query Language) is essential in managing data for dynamic websites like online stores. It allows for the efficient storage and retrieval of product information, user accounts, and order details in relational databases. When a user browses products, SQL queries fetch the relevant data from the database to display on the website. Similarly, when a user creates an account or places an order, SQL commands insert or update the necessary records in the database, ensuring data consistency and integrity.

1. **Write a short explanation (3-5 sentences) in your document about the role of SQL in web applications.**

## Role of SQL in Web Applications

SQL plays a critical role in web applications by enabling the management and manipulation of data stored in relational databases. It allows web applications to perform various operations like querying data, updating records, inserting new entries, and deleting obsolete data. This ensures that dynamic websites can provide up-to-date information and services to users, such as displaying current product availability, processing user registrations, and managing customer orders efficiently.

1. **List 3 benefits of using SQL for web applications.**

 **Relational Data Management**: SQL databases provide robust relational data management capabilities, allowing web applications to efficiently organize and retrieve structured data. This ensures data integrity through normalization and relationships between tables.

 **Scalability**: SQL databases are highly scalable, capable of handling large volumes of data and supporting concurrent users effectively. This scalability is essential for web applications that need to grow and handle increased traffic over time.

 **Data Security**: SQL databases offer strong data security features, including access control, encryption, and transaction management. This helps protect sensitive information stored in the database, ensuring compliance with security standards and regulations.

1. **Think about efficiency, data organization, and data retrieval capabilities. Briefly explain each benefit in your document (1-2 sentences per benefit).**

 **Efficiency**

SQL databases optimize data storage and retrieval operations through indexing, query optimization, and transaction management. This efficiency ensures that web applications can handle large datasets and complex queries swiftly, improving overall performance.

 **Data Organization**

SQL databases enforce relational data models, allowing developers to organize data into tables with defined relationships. This structured approach ensures data integrity, reduces redundancy, and facilitates easier maintenance and updates of the application's data.

 **Data Retrieval Capabilities**

SQL databases support powerful querying capabilities, including JOINs, aggregations, and filtering. This enables web applications to retrieve specific subsets of data efficiently based on complex criteria, ensuring responsiveness and flexibility in serving user requests.

1. **List any 3 Database Management Systems.**

 MySQL

 PostgreSQL

 MongoDB

## Part 2: Database Fundamentals (45 minutes)

**Question 2.1: Tables**

A database table is a structured collection of data organized into rows and columns. Each row represents a unique record, while each column represents a specific attribute or field of the data. This structure is similar to a spreadsheet where rows are similar to spreadsheet rows and columns are akin to spreadsheet columns. Both tables and spreadsheets allow for organized storage and retrieval of data based on specific criteria or relationships.

**Question 2.2: Columns**

Columns in a database table define the attributes or properties of the data being stored. Each column has a defined data type that specifies the kind of data it can hold. For example, a column named "CustomerName" in a table might hold text data types like names, addresses, and other information related to customers. Data types ensure that the database can efficiently store and retrieve data based on the nature of each attribute.

**Data Types: Why are data types important in a database?**

Data types are crucial in databases for several reasons:

1. **Data Integrity**: Data types enforce constraints on the kind of data that can be stored in a column, ensuring that only valid and appropriate data is inserted. For example, a "Date" data type ensures that only date values are stored in a column meant for dates, preventing errors and maintaining data consistency.
2. **Storage Efficiency**: Data types determine how data is stored on disk, optimizing storage space and retrieval performance. For instance, storing integers takes less space compared to storing text or dates in a specific format.
3. **Query Optimization**: Database engines use data types to optimize query execution. By knowing the data type of each column, the database can choose the most efficient way to process and retrieve data, improving overall query performance.

**Common Data Types:**

1. **Text (or String)**: Used for storing alphanumeric characters, text data types accommodate variable-length strings of characters. They are essential for storing textual information such as names, addresses, and descriptions.
2. **Number (or Numeric)**: Numeric data types store numerical values, including integers (whole numbers) and floating-point numbers (decimals). They are used for storing quantities, measurements, and calculations.
3. **Date and Time**: Date and time data types store temporal information, such as specific dates, times of day, or both. They ensure accurate representation and manipulation of chronological data in databases.

## Part 3: Expense Tracker Database Design (45 minutes)

### 3.1. Planning

To build an Expense Tracker application, we need to track various data points related to expenses. Here are at least five essential data points:

1. **Expense Amount**: The monetary value of the expense.
2. **Date**: The date when the expense was made.
3. **Category**: The category under which the expense falls, such as food, transportation, entertainment, etc.
4. **Description**: A brief description or note about the expense.
5. **Payment Method**: The method used to pay the expense, such as cash, credit card, debit card, etc.

### 3.2. Tables

Given the data points identified, we can design a basic database schema with one main table named "Expenses". Below is the definition of the columns needed for this table along with their data types.

#### Table: Expenses

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| expense\_id | INT | Primary key, unique identifier for each expense |
| amount | DECIMAL(10, 2) | The amount of the expense |
| date | DATE | The date when the expense occurred |
| category | VARCHAR(50) | The category of the expense |
| description | TEXT | A brief description of the expense |
| user\_id | INT | users |

Below is a more detailed breakdown of each column with its data type and constraints:

* **expense\_id**: INT, PRIMARY KEY, AUTO\_INCREMENT
* **amount**: DECIMAL(10, 2) (e.g., 1234.56)
* **date**: DATE
* **category**: VARCHAR(50)
* **description**: TEXT
* **user\_id:** INT, FOREIGN KEY