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

**Question 1: Research**

1.1. In a dynamic website like an online store, SQL plays a crucial role in managing data behind the scenes. SQL databases stores product information such as descriptions, prices, and availability. User accounts with details like usernames, passwords (usually encrypted), and personal information are managed securely. Order details, including customer information, purchased items, payment details, and shipping addresses, are also stored and managed using SQL.

1.2. SQL (Structured Query Language) is essential in web applications because it manages relational databases. It also allows developers to create, read, update, and delete data stored in databases efficiently. Moreover, SQL queries enable dynamic content generation, data validation, and retrieval based on user inputs and application requirements.

1.3. Three benefits of using SQL for web applications:

* Data Integrity - SQL databases enforce constraints (like primary keys, foreign keys, and data types) to ensure data accuracy and reliability.
* Security - SQL databases offer robust security features, including authentication, authorization, and encryption, to protect sensitive data from unauthorized access.
* Scalability - SQL databases can handle large amounts of data and accommodate growth by optimizing storage and query processing.

1.4. Brief explanations for:

* Efficiency - SQL's optimized query processing and indexing mechanisms ensure quick data retrieval and manipulation, enhancing overall application performance.
* Data Retrieval Capabilities - SQL's powerful querying capabilities enable complex data retrieval operations, supporting dynamic content generation and personalized user experiences.
* Data Organization - SQL databases use tables with predefined schemas, allowing for structured data storage that facilitates efficient data management and organization.

1.5. Three Database Management Systems (DBMS):

* MySQL
* PostgreSQL
* Microsoft SQL Server

**Part 2: Database Fundamentals (45 minutes)**

2.1. **Tables -** A database table is a structured collection of data organized into rows and columns. Each row represents a record, and each column represents a specific attribute or field within that record. Similar to a spreadsheet, tables allow data to be organized in a tabular format where each cell (intersection of a row and column) holds a specific data value. Both tables and spreadsheets are important because they facilitate easy storage, retrieval, and manipulation of data in a structured manner.

2.2. **Columns**: In a database table, columns represent the different types of data attributes that define the structure of each record. For instance, in a "Customers" table, columns might include "CustomerID" (a unique identifier), "Name" (text data type for customer names), and "BirthDate" (date data type for customer birthdates). Columns ensure that each piece of data within a record is categorized appropriately and can be efficiently queried and manipulated.

**Data Types**: They define the kind of data that can be stored in each column, ensuring data integrity and efficient storage and retrieval.

1. **Text (VARCHAR) -** Used for storing variable-length alphanumeric characters such as names, addresses, and descriptions. Example: VARCHAR (50) for storing up to 50 characters.
2. **Number (Integer, Decimal) -** Used for numeric data such as quantities, prices, and counts. Example: INTEGER for whole numbers or DECIMAL (8, 2) for numbers with up to 8 digits, 2 of which are decimals.
3. **Date (DATE) -** Used for storing dates without time components, like birthdates, order dates, and event dates. Example: DATE for storing dates in YYYY-MM-DD format.

2.3. Data types are important because they ensure data integrity by enforcing constraints on what type of data can be stored in each column. Moreover, they optimize storage space and query performance by defining how data is stored internally.

* **Text (VARCHAR)**: Ensures efficient storage and retrieval of variable-length textual data. It optimizes storage by only using as much space as needed for the actual data stored.
* **Number (Integer, Decimal)**: Ensures accurate arithmetic operations and efficient storage of numeric data. Integer data types are used for whole numbers, while decimal data types provide precision for calculations involving fractions.
* **Date (DATE)**: Ensures consistent storage and comparison of dates, allowing for easy date-based queries and calculations without the complexity of time components.

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

3.1. **The data Points that are needed to tracked are:**

1. Expense ID (unique identifier for each expense)
2. Amount (numeric value of the expense)
3. Date (date when the expense occurred)
4. Category (category or type of expense)
5. Description (optional description or notes about the expense)

3.2. **Database Schema Design:**

**Table name: Expenses**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| expense\_ID | INT | Unique identifier for each expense (primary key) |
| amount | Decimal (10,2) | Numeric value representing the expense value |
| date | DATE | Date when the expense occurred |
| category | VARCHAR (50) | Category of the expense |
| description | TEXT | Description of the expense |