Creating and Modifying Tables

Concept-Based Questions

Q1. What SQL command is used to create a table?
A1. CREATE TABLE

Q2. How do you modify the structure of a table?
A2. Using the ALTER TABLE command.

Q3. What is the difference between DROP and DELETE?
A3. DROP removes the table and its data permanently. DELETE removes only the data.

Q4. Can we rename a column in SQL?
A4. Yes, using ALTER TABLE with RENAME COLUMN (MySQL 8.0+).

Q5. How do you add a default value to a column?
A5. Using DEFAULT keyword in column definition.

Code-Based Questions

Q6. Create a table ``.

```sql
CREATE TABLE departments (
  department_id INT PRIMARY KEY,
  department_name VARCHAR(50)
);
```

Q7. Add a new column ** to **.

```sql
ALTER TABLE departments ADD location VARCHAR(100);
```

Q8. Rename column ** to **.

```sql
ALTER TABLE departments RENAME COLUMN location TO office_location;
```

Q9. Delete column ``.

```sql
ALTER TABLE departments DROP COLUMN office_location;
```

Q10. Drop the `` table.**

```sql
DROP TABLE departments;
```

Scenario-Based

Q11. Add a column `` with default current timestamp.**

```sql
ALTER TABLE employees ADD created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP;
```

Indexes and Constraints

Concept-Based Questions

Q1. What is an index in SQL?
A1. An index is used to speed up retrieval of rows by using a pointer.

Q2. What are constraints in SQL?
A2. Rules enforced on data in tables. Examples: PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK, DEFAULT.

Q3. What is a primary key?
A3. A column or combination of columns that uniquely identifies each row.

Q4. What is the difference between UNIQUE and PRIMARY KEY?
A4. Both enforce uniqueness. PRIMARY KEY also enforces NOT NULL.

Q5. What is a foreign key?
A5. A column that establishes a relationship between two tables.

Code-Based Questions

Q6. Create a table with primary key and unique constraint.

```
CREATE TABLE products (
  product_id INT PRIMARY KEY,
  product_name VARCHAR(50) UNIQUE
);
```

Q7. Add a NOT NULL constraint to a column.

```
ALTER TABLE employees MODIFY last_name VARCHAR(50) NOT NULL;
```

Q8. Add a foreign key to ** referencing **.

```
ALTER TABLE employees
ADD CONSTRAINT fk_dept
FOREIGN KEY (department_id) REFERENCES departments(department_id);
```

Q9. Create an index on salary column.

```
CREATE INDEX idx_salary ON employees(salary);
```

Q10. Drop an index.

```
DROP INDEX idx_salary ON employees;
```

Scenario-Based

Q11. Prevent negative salaries using a constraint.

```
ALTER TABLE employees ADD CONSTRAINT chk_salary CHECK (salary >= 0);
```