**Exercise 1: Create a Table with Primary Key Constraint**

**Create a table named "Students" with the following columns:**

student\_id (Primary Key)

first\_name

last\_name

age (Must be greater than or equal to 18)

Solution:

CREATE TABLE Students (

student\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

age INT CHECK (age >= 18)

);

**Exercise 2: Add Foreign Key Constraint**

**Create a table named "Courses" with the following columns:**

course\_id (Primary Key)

course\_name

student\_id (Foreign Key referencing student\_id from the Students table)

Solution:

CREATE TABLE Courses (

course\_id INT PRIMARY KEY,

course\_name VARCHAR(100),

student\_id INT,

FOREIGN KEY (student\_id) REFERENCES Students(student\_id)

);

**Exercise 3: Add Unique Constraint**

**Alter the Students table to ensure that the combination of first\_name and last\_name is unique.**

Solution:

ALTER TABLE Students

ADD CONSTRAINT unique\_name\_combination UNIQUE (first\_name, last\_name);

**Exercise 4: Add Check Constraint**

**Alter the Courses table to ensure that the course\_duration is between 1 and 12.**

Solution:

ALTER TABLE Courses

ADD CONSTRAINT check\_course\_duration CHECK (course\_duration >= 1 AND course\_duration <= 12);

**Exercise 5: Remove a Constraint**

**Remove the unique\_name\_combination constraint from the Students table.**

Solution:

ALTER TABLE Students

DROP INDEX unique\_name\_combination;

**Exercise 6: Disable and Enable Constraints**

**Temporarily disable the foreign key constraint on the Courses table, perform an update, and then enable the constraint again.**

Solution:

-- Disable the constraint

SET FOREIGN\_KEY\_CHECKS = 0;

-- Perform the update

UPDATE Courses

SET student\_id = 2

WHERE course\_id = 1;

-- Enable the constraint

SET FOREIGN\_KEY\_CHECKS = 1;

Exercise 7: **Create a Table with Composite Primary Key and Foreign Key Constraint**

**Create two tables: "Orders" and "OrderItems."**

**The "Orders" table should have the following columns:**

order\_id (Primary Key)

customer\_id

order\_date

**The "OrderItems" table should have the following columns:**

order\_id (Foreign Key referencing order\_id from Orders table)

product\_id

quantity

Solution:

-- **Create Orders table**

CREATE TABLE Orders (

order\_id INT PRIMARY KEY,

customer\_id INT,

order\_date DATE,

CONSTRAINT fk\_customer FOREIGN KEY (customer\_id) REFERENCES Customers(customer\_id)

);

-- **Create OrderItems table**

CREATE TABLE OrderItems (

order\_id INT,

product\_id INT,

quantity INT,

PRIMARY KEY (order\_id, product\_id),

FOREIGN KEY (order\_id) REFERENCES Orders(order\_id)

);

**Exercise 8: Create a Table with Multiple Constraints**

**Create a table named "Employees" with the following columns:**

employee\_id (Primary Key)

first\_name

last\_name

birth\_date

hire\_date

salary (Must be a positive value)

department\_id (Foreign Key referencing department\_id from Departments table)

supervisor\_id (Foreign Key referencing employee\_id from Employees table)

CHECK constraint to ensure that birth\_date is before hire\_date

Solution:

-- **Create Departments table**

CREATE TABLE Departments (

department\_id INT PRIMARY KEY,

department\_name VARCHAR(100)

);

-- **Create Employees table**

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

birth\_date DATE,

hire\_date DATE,

salary DECIMAL(10, 2) CHECK (salary > 0),

department\_id INT,

supervisor\_id INT,

CONSTRAINT fk\_department FOREIGN KEY (department\_id) REFERENCES Departments(department\_id),

CONSTRAINT fk\_supervisor FOREIGN KEY (supervisor\_id) REFERENCES Employees(employee\_id),

CHECK (birth\_date < hire\_date)

);

**Exercise** 9: **Creating a Table with Unique Constraint and Inserting Data**

**Scenario: You want to create a "Books" table with columns: book\_id, title, author, and ISBN. You want to ensure that ISBN values are unique.**

Solution

-- Create Books table with unique constraint

CREATE TABLE Books (

book\_id INT PRIMARY KEY,

title VARCHAR(200),

author VARCHAR(100),

ISBN VARCHAR(13) UNIQUE

);

-- Insert data into Books table

INSERT INTO Books (book\_id, title, author, ISBN)

VALUES

(1, 'The Great Gatsby', 'F. Scott Fitzgerald', '978-0743273565'),

(2, 'To Kill a Mockingbird', 'Harper Lee', '978-0061120084'),

(3, '1984', 'George Orwell', '978-0451524935');

**Exercise 10: Creating customers table with the following columns**

customer\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100) UNIQUE,

phone\_number VARCHAR(20),

birth\_date DATE,

registration\_date DATE

**Try to add check constraints to check birth\_date and registration\_date <= current date.**

-- Create Customers table

CREATE TABLE Customers (

customer\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100) UNIQUE,

phone\_number VARCHAR(20),

birth\_date DATE,

registration\_date DATE,

CHECK (birth\_date <= CURDATE()),

CHECK (registration\_date <= CURDATE())

);

-- Insert data into Customers table

INSERT INTO Customers (customer\_id, first\_name, last\_name, email, phone\_number, birth\_date, registration\_date)

VALUES

(1, 'John', 'Doe', 'john@example.com', '123-456-7890', '1990-05-15', '2022-01-10'),

(2, 'Jane', 'Smith', 'jane@example.com', '987-654-3210', '1988-10-20', '2021-07-05'),

(3, 'Michael', 'Johnson', 'michael@example.com', '555-123-4567', '1995-02-18', '2023-03-22');

**Exercise 11: Creating a Table with Foreign Key Constraint and Inserting Data**

**Scenario: You want to create a "Orders" table with columns: order\_id, customer\_id, order\_date, and total\_amount. You also want to create a foreign key constraint referencing the "Customers" table.**

-- Create Orders table with foreign key constraint

CREATE TABLE Orders (

order\_id INT PRIMARY KEY,

customer\_id INT,

order\_date DATE,

total\_amount DECIMAL(10, 2),

CONSTRAINT fk\_customer FOREIGN KEY (customer\_id) REFERENCES Customers(customer\_id)

);

-- Insert data into Orders table

INSERT INTO Orders (order\_id, customer\_id, order\_date, total\_amount)

VALUES

(1, 1, '2023-08-14', 150.00),

(2, 2, '2023-08-15', 200.00),

(3, 1, '2023-08-16', 75.00);

**Exercise 12: Creating a Table with Check Constraint and Inserting Data**

**Scenario: You want to create a "Employees" table with columns: employee\_id, first\_name, last\_name, birth\_date, and hire\_date. You want to add a check constraint to ensure that the birth date is before the hire date.**

-- Create Employees table with check constraint

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

birth\_date DATE,

hire\_date DATE,

CHECK (birth\_date < hire\_date)

);

-- Insert data into Employees table

INSERT INTO Employees (employee\_id, first\_name, last\_name, birth\_date, hire\_date)

VALUES

(1, 'John', 'Doe', '1990-05-15', '2021-03-10'),

(2, 'Jane', 'Smith', '1988-10-20', '2020-07-05'),

(3, 'Michael', 'Johnson', '1995-02-18', '2022-01-22');