1. **Create Table with Primary Key and Foreign Key Constraints:**

```
-- Create Books table with primary key

CREATE TABLE Books (

    book_id INT PRIMARY KEY,

    title VARCHAR(255),

    author VARCHAR(255),

    publication_year INT,

    price DECIMAL(10, 2),

    stock_quantity INT

);

-- Create Orders table with foreign key constraints

CREATE TABLE Orders (

    order_id INT PRIMARY KEY,

    customer_id INT,

    book_id INT,

    quantity INT,

    order_date DATE,

    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id),

    FOREIGN KEY (book_id) REFERENCES Books(book_id)

);
```

2. **Alter Table with Add and Modify:**

a. **Add a new column:**

```
ALTER TABLE Books

ADD column_name datatype;
```

Example:

```
ALTER TABLE Books

ADD genre VARCHAR(50);
```

b. **Modify an existing column:**

```
ALTER TABLE Books
MODIFY column_name new_datatype;
```

```
ALTER TABLE Books
MODIFY publication_year YEAR;
```

```
DROP TABLE table_name;
```

1. **Create Table with Primary Key and Foreign Key Constraints:**

```
-- Create a table for Employees with primary key and foreign key constraints
CREATE TABLE Employees (
   emp_id INT PRIMARY KEY,
   emp_name VARCHAR(100),
   department_id INT,
   FOREIGN KEY (department_id) REFERENCES Departments(department_id)
);
```

2. **Alter Table with Add and Modify:**

a. **Add a new column to the Employees table:**

```
ALTER TABLE Employees
ADD emp_salary DECIMAL(10, 2);
```

b. **Modify the datatype of an existing column in the Employees table:**

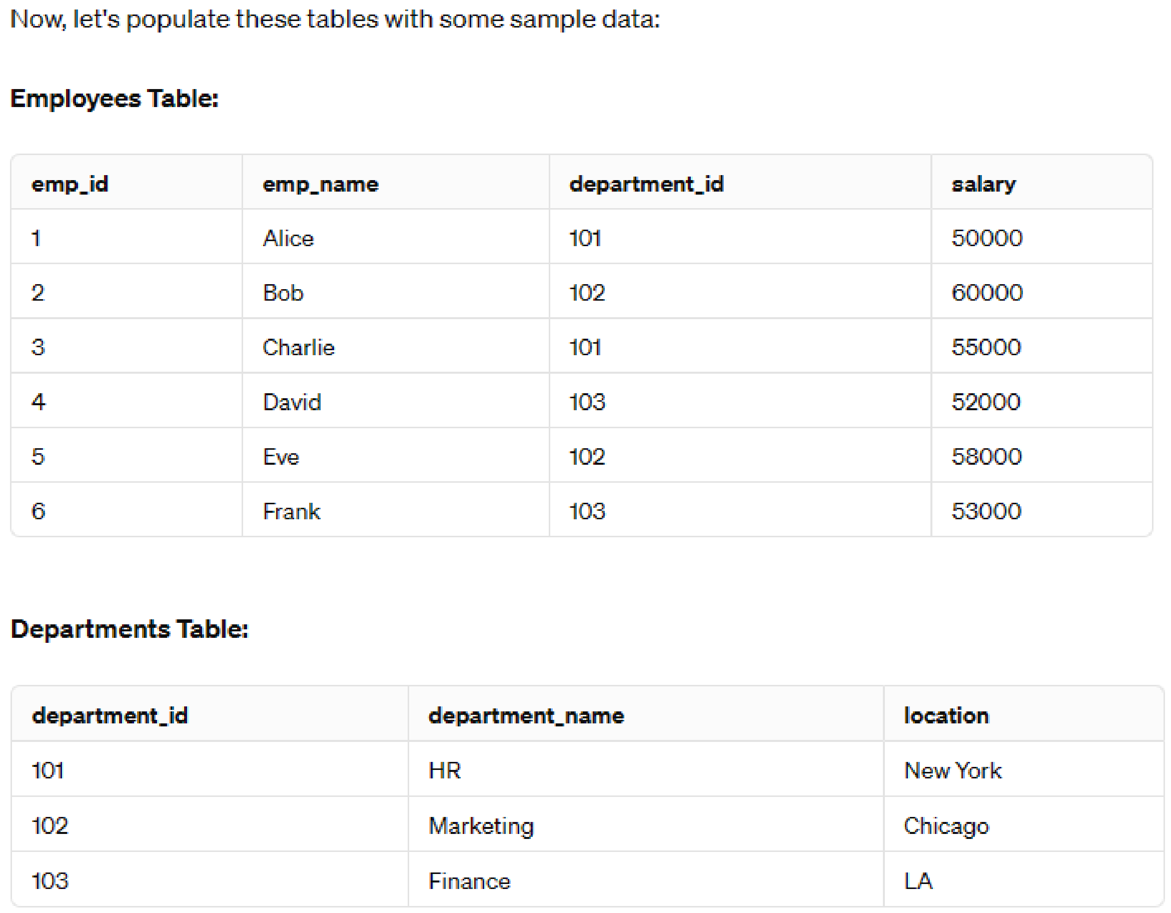
```
ALTER TABLE Employees
MODIFY emp_name VARCHAR(255);
```

3. **Drop Table:**

```
-- Drop the Employees table from the database
DROP TABLE Employees;
```

Let's consider two tables, "Employees" and "Departments", for demonstration purposes:

1. **Employees Table:**
   - emp_id (Primary Key)
   - emp_name
   - department_id (Foreign Key referencing department_id in Departments table)
   - salary
2. **Departments Table:**
   - department_id (Primary Key)
   - department_name
   - location



Now, let's populate these tables with some sample data:

Employees Table:

emp_id	emp_name	department_id	salary
1	Alice	101	50000
2	Bob	102	60000
3	Charlie	101	55000
4	David	103	52000
5	Eve	102	58000
6	Frank	103	53000

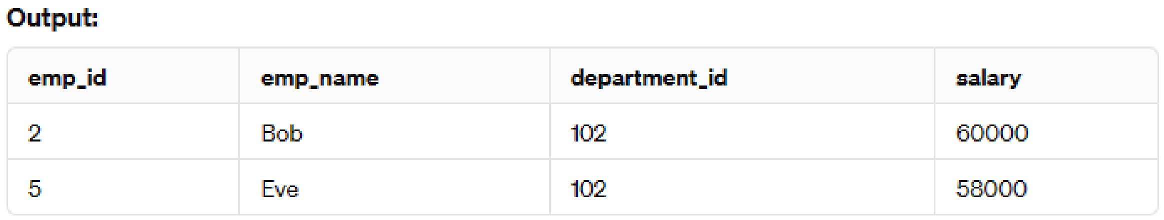
Departments Table:

department_id	department_name	location
101	HR	New York
102	Marketing	Chicago
103	Finance	LA

Now, let's demonstrate the SQL relational operators:

1. **Selection (σ):**
   - Select employees with a salary greater than 55000.

SELECT * FROM Employees WHERE salary > 55000;



Output:

emp_id	emp_name	department_id	salary
2	Bob	102	60000
5	Eve	102	58000

Projection (π):

- Select only the employee names and salaries.
  SELECT emp_name, salary FROM Employees;

Output:

emp_name	salary
Alice	50000
Bob	60000
Charlie	55000
David	52000
Eve	58000
Frank	53000

sql