Certainly! Let’s create another SQL exercise with a different scenario. This time, we’ll focus on a library management system. Here’s the exercise:

**SQL Exercise: Library Management System**

**Scenario:**

You are designing a library management system for a university library. The system needs to track information about books, authors, and book loans.

**Tables:**

1. **Books**:
   * Columns:
     + book\_id (Primary Key)
     + title
     + author\_id (Foreign Key referencing Authors)
     + publication\_year
     + genre
2. **Authors**:
   * Columns:
     + author\_id (Primary Key)
     + first\_name
     + last\_name
     + birth\_year
3. **BookLoans**:
   * Columns:
     + loan\_id (Primary Key)
     + book\_id (Foreign Key referencing Books)
     + borrower\_id (Foreign Key referencing Students)
     + loan\_date
     + return\_date

**Relationships:**

* **One-to-Many**:
  + Each book can have one author.
  + Each student can borrow multiple books.

**Tasks:**

1. Create the above three tables with appropriate data types for each column.
2. Set up the necessary primary keys and foreign keys.
3. Write an SQL script to insert sample data into the Books and Authors tables.
4. Query the database to retrieve:
   * All books written by a specific author.
   * All books currently borrowed by a specific student.

**Sample Data (for reference):**

* Books:
  + Book 101: “The Catcher in the Rye”, Author: J.D. Salinger, 1951
  + Book 201: “Pride and Prejudice”, Author: Jane Austen, 1813
* Authors:
  + Author 1: J.D. Salinger, born 1919
  + Author 2: Jane Austen, born 1775

**Bonus Challenge:**

Design and implement a table to track book copies (physical copies available for borrowing) and their availability status (e.g., available, checked out, lost).

Feel free to adapt this exercise to suit your teaching needs or create additional constraints. Happy SQL practicing! 📚💻

Certainly! Let’s create another SQL exercise with a different scenario. This time, we’ll focus on an online store’s inventory management system. Here’s the exercise:

**SQL Exercise: Online Store Inventory Management**

**Scenario:**

You are designing a database for an online store that sells various products. The system needs to track information about products, categories, and customer orders.

**Tables:**

1. **Products**:
   * Columns:
     + product\_id (Primary Key)
     + product\_name
     + category\_id (Foreign Key referencing Categories)
     + price
     + stock\_quantity
2. **Categories**:
   * Columns:
     + category\_id (Primary Key)
     + category\_name
     + description
3. **Orders**:
   * Columns:
     + order\_id (Primary Key)
     + customer\_id (Foreign Key referencing Customers)
     + order\_date
     + total\_amount

**Relationships:**

* **One-to-Many**:
  + Each product belongs to one category.
  + Each customer can place multiple orders.

**Tasks:**

1. Create the above three tables with appropriate data types for each column.
2. Set up the necessary primary keys and foreign keys.
3. Write an SQL script to insert sample data into the Products and Categories tables.
4. Query the database to retrieve:
   * All products in a specific category.
   * Total revenue generated by a specific customer.

**Sample Data (for reference):**

* Products:
  + Product 101: Laptop, Category: Electronics, Price: $800, Stock Quantity: 50
  + Product 201: T-shirt, Category: Apparel, Price: $20, Stock Quantity: 200
* Categories:
  + Category 1: Electronics, Description: Gadgets and devices
  + Category 2: Apparel, Description: Clothing and accessories

**Bonus Challenge:**

Design and implement a table to track customer reviews and ratings for products.

Feel free to customize this exercise further or add additional constraints based on your teaching objectives. Happy SQL practicing! 📚💻

Certainly! Let’s create another SQL exercise, this time focusing on a **human resources (HR) management system**. Here’s the scenario and exercise:

**SQL Exercise: HR Management System**

**Scenario:**

You are tasked with designing a database for an HR department to manage employee information, job roles, and performance evaluations.

**Tables:**

1. **Employees**:
   * Columns:
     + employee\_id (Primary Key)
     + first\_name
     + last\_name
     + hire\_date
     + department
     + job\_title
2. **JobRoles**:
   * Columns:
     + role\_id (Primary Key)
     + role\_name
     + responsibilities
     + salary\_range
3. **PerformanceReviews**:
   * Columns:
     + review\_id (Primary Key)
     + employee\_id (Foreign Key referencing Employees)
     + review\_date
     + performance\_rating
     + comments

**Relationships:**

* **One-to-Many**:
  + Each employee has one job role.
  + Each employee can have multiple performance reviews.

**Tasks:**

1. Create the above three tables with appropriate data types for each column.
2. Set up the necessary primary keys and foreign keys.
3. Write an SQL script to insert sample data into the Employees and JobRoles tables.
4. Query the database to retrieve:
   * All employees in a specific department.
   * Average performance rating for each job role.

**Sample Data (for reference):**

* Employees:
  + Employee 101: John Smith, hired on 2022-01-15, Department: Sales, Job Title: Sales Representative
  + Employee 201: Jane Brown, hired on 2021-08-20, Department: Marketing, Job Title: Marketing Specialist
* JobRoles:
  + Role 1: Sales Representative, Responsibilities: Client meetings, Sales targets, Salary Range: $40,000 - $60,000
  + Role 2: Marketing Specialist, Responsibilities: Campaign planning, Social media management, Salary Range: $45,000 - $65,000

**Bonus Challenge:**

Design and implement a table to track employee training programs and certifications.

Feel free to customize this exercise further or add additional constraints based on your teaching objectives. Happy SQL practicing! 📊💼