

## Questions

### 1. Create the Tables

Write the SQL queries to create all five tables with appropriate constraints.

### 2. Insert Data

Insert the following data into the tables:

#### Table:Authors

Create Primary Key as Author\_id.

author_id	name	country
1	J.K. Rowling	UK
2	George R.R. Martin	USA
3	Haruki Murakami	Japan

#### Table:Books

Create Book id as Primary Key

And Author\_Id as foreign key

book_id	title	genre	price	author_id
1	Harry Potter	Fantasy	20.99	1
2	Game of Thrones	Fantasy	25.99	2
3	Norwegian Wood	Fiction	15.49	3

#### Table:Customers

Create Customer Id as Primary key

customer_id	name	email
1	Alice	alice@gmail.com
2	Bob	<a href="mailto:bob@yahoo.com">bob@yahoo.com</a>
3	Charlie	charlie@hotmail.com

#### Table:Orders

Create Order\_Id as Primary Key

Customer\_Id as Foreign key

order_id	customer_id	order_date
----------	-------------	------------

1	1	2023-12-01
---	---	------------

2	2	2023-12-02
---	---	------------

### **Table:Order Details**

Create Order\_Id as Foreign Key

order_detail_id	order_id	book_id	quantity
-----------------	----------	---------	----------

1	1	1	2
---	---	---	---

2	2	2	1
---	---	---	---

3	2	3	1
---	---	---	---

---

### **Queries to Solve**

#### **1. Create Queries**

- Write SQL to create the Authors and Books tables with proper foreign key relationships.

#### **2. Insert Queries**

- Write SQL to insert the given data into the tables.

#### **3. Joins**

- Fetch all orders along with the names of customers and the titles of books they ordered.
- Show a list of authors and the total revenue generated from their books.

#### **4. Subqueries**

- Find the author who has the most expensive book.
- List customers who have ordered more than one type of book genre.

### **Expectations from Learners:**

1. Design normalized relational tables with proper constraints (Primary Key, Foreign Key, etc.).
2. Insert sample data correctly into the created tables.
3. Write SQL queries involving Joins, Subqueries, and Aggregations.
4. Ensure data integrity and handle relationships between multiple tables.
5. Present optimized, readable, and well-structured SQL code.
6. Provide meaningful comments or explanations where necessary.

## **Overview of Deliverables:**

1. A single SQL script file (.sql) containing all the CREATE, INSERT, and SELECT queries in sequential order.
2. A word document with naming convention as '**<learner\_official\_id>\_MySQL\_SolutionDocument\_PL1**' and having the below details:
  - Explaining the approach taken for solving the use case.
  - Screenshots of successful query execution, results in MySQL Workbench.
3. Upload the above '**<learner\_official\_id>\_MySQL\_SolutionDocument\_PL1**' file in the IIHT portal, while submitting the solution of the Use Case.