# Eduardo Molina

**Final Report: Sakila Database Management in MySQL Workbench**

1. **Identifying Tools and Statements for Modifying Database Content**

* INSERT adds a new entity to the table.
* UPDATE changes information of the existing entities.
* DELETE deletes a entity of a table.
* ALTER changes the table structure.

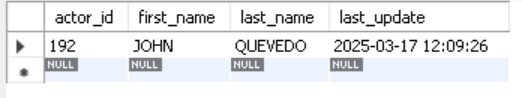
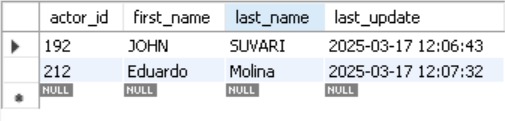
Functionality of the tools in MySQL Workbench:

* SQL Editor is used to run queries.
* Schema Inspector to explore the structure of the databases.
* Query Builder allows complex queries to be more visual.

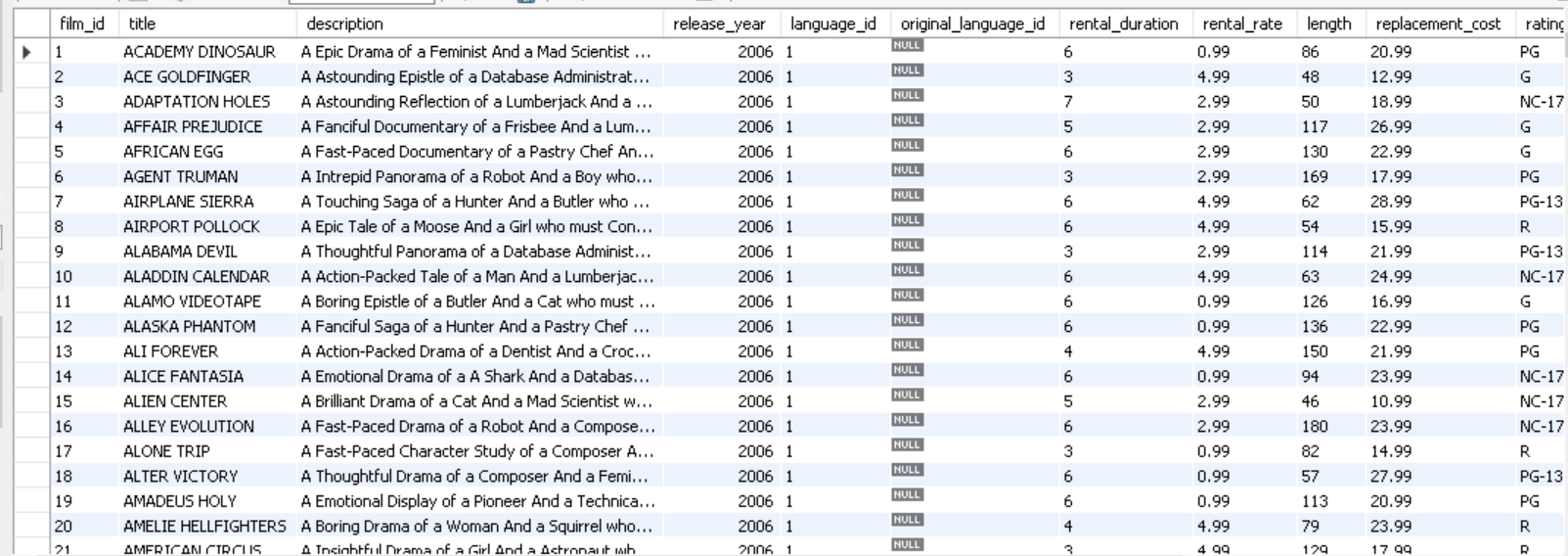
1. **Data Insertion, Deletion, and Update**

I added a new carácter Eduardo Molina,

updated JOHN SUVARI to JOHN QUEVEDO and deleted Eduardo Molina.

****

1. **Creating a Table from a Query Result**

Extracted every film made after 2005 and moved it to a new table called recent\_films. 

1. **Complex SQL Queries**

I retrived all the recent costumers , identified the most rented film and obteind the total revenue of the stores.

1. **Understanding Transactions**

Transactions ensure data integrity and consistency.

I inserted a new rental record, updated the inventory to reflect the rental and committed the changes to ensure database integrity.

1. **Rolling Back Transactions**

Rollbacks are really important so no incorrect data is saved in the database.I performed a rollback on a renting of a out of stock movie

1. **Record Locking Policies**

To comprehend how mysql handles concurrent data modifications I used pessimistic and optimistic Lockings.

Optimistic Lockings :Allows multiple transactions and checks for posible conflicts before committing changes.

Pessimistic Locking:Does not let users modify a record until the current transaction is not complete.

1. **Ensuring Data Integrity and Consistency**

Data Integrity is important in every database system.Sakila database had data duplication, data inconsistencies and missing foreign keys so I had to implement a few Triggers to valídate data that is inserted or updated and establish relationships between tables .