**CONESTOGA COLLEGE**



**MUSIC STORE DATABASE**

**DATABASE DESIGN ANMAINTENANCE**

**PROF.DR. ANGELINA ZIESEMER**

**HARMANPREET KAUR**

**SUBMISSION DATE -16 AUG 2021**

**Table of contents**

1. **Introduction**
2. **General scenario**
3. **Specific scenario**
4. **Documentation**
5. **Conclusion**
6. **Appendix**

**Introduction**

The music store database is an online store that has a collection of music albums of many artist.in this music store database there are seven tables and three linking tables.The music store database is a online music store that display the data about the albums such as no of albums, instruments used in the albums, orders total in the store and the data related to the music album i.e. release date, sold items, available items, discount, purchases, name of the artist, price etc.

**General scenario**

1. **Display a list of clients that spent more than the average spent by client in the past month.**

**I have first calculated the amount spent by the customer by including tax and discount in the past month and got an average of result set. Then I added a query to select customers from table having amount they spent greater than the average.**

**2. The top sold products and least sold products over a week.**

**I have first calculated the amount spent by the customer by including tax and discount in the past month and got an average of result set. Then I added a query to select customers from table having amount they spent greater than the average.**

**3. The maximum price of products in the same genre (for example, rock, pop, country, hip-hop). Use GROUP BY to list all the genres and their maximum price.**

**I have joined albums and genre table with genre\_id and used group by to get the required result set.**

1. **List how many customers the system has by location (Country, Province, and City), and then sort them.**

In this query I have joined the two tables’ customers and address and used the count function on customer\_id of customers to get the no. of customers and used group by city for location and arranged in ascending order

1. **List how many products the store has sold for a particular month.**

In this query I have joined two tables orders and order\_items. sum function is used to add all the products sold and by using cast to convert in between the dates to get for a particular month.

1. **List how many distinct albums each singer has.**

In this Query, I have joined three tables albums ,

Albums\_has\_artists and artists to get the no of albums each singer has and arranged by ascending order by singer name.

1. **List how many copies of an album are available of a particular singer.**

-In this query to find the total number of available copies I need to subtract No\_of\_albums from No\_of\_samples as No\_of\_albums indicate the total number of albums and No\_of\_samples display the total number of sold items. As all columns are not in a single table so to get the data from all tables I need to connect three tables using the JOIN statement.Those tables are artist, artist\_has\_albums, order\_items.

**Specific scenario**

1. **Filter albums according to instruments used in the album.**

* In this query I need to find the instrument name that is used in a particular album. So to filter the instrument name with respect to the album name I need album name and instrument name and to get those two things I need to connect two tables using the inner join. Those tables are album, instrument.

**2.Most popular instrument.**

- In this query to get the most popular instrument. I need to get the instrument name that is used in many albums. By using the MAX function, I get the name of the instrument that is used in most of the albums.

**Documentation**

This project includes following documentation: -

1. **Tracking sheet**

**A computer screen capture

Description automatically generated with low confidence**

1. **Requirements**
2. To fulfill the requirements firstly we created a rough model of the music store database. Created a rough sketch of ER diagram in MySQL Workbench.A screenshot of a computer

   Description automatically generated

**Roadblocks: -** Increating relationships between tables.

**Changes: -**

**1-** change the products table name to an album table as albums are the only product sold in store, albums are more readable.

**2-** The Albums (products in the above figure) cannot be directly connected to Customers table as the details relative to the sales are stored in order items table.

**3-**The relationships between some tables are changed acc. to the logic.

**Diagram

Description automatically generated**

**Remarks-**

* There is a column Order\_items that is violating the 3NF. Model already contains the foreign keys that are needed .If you want to add a column to say how many samples of the same album are being sold, you can add.
* Also, phoneNo should not be INT... you will only use INT for those columns that you need to perform an operation... for example, AVG, SUM, etc... if you don't perform any operation for a column, you should not use int... instead, you can use varchar.
* Column unit price on table orders is not necessary and it violates the 3NF... the price is based on the album and the quantity as well... again, take a look at my\_guitar\_shop model so you can understand what I mean.

Diagram

Description automatically generated with medium confidence

1. After creating a rough model, we converted a model into script by using the MySQL workbenches by default function of converting the E-r diagram into script.

File > export > forward engineer SQL CREATE SCRIPT.

1. After created the script we entered the data in tables.
2. After entering the data, we created a backup of the music store database.
3. Then we started working on the scenarios: - the general scenario and the specific scenario
4. THE GENERAL SCENARIO: -
   * + 1. **Display a list of clients that spent more than the average spent by client in the past month.**
   1. **I have first calculated the amount spent by the customer by including tax and discount in the past month and got an average of result set. Then I added a query to select customers from table having amount they spent greater than the average.**
5. **The top sold products and least sold products over a week.**
   1. **I have first calculated the amount spent by the customer by including tax and discount in the past month and got an average of result set. Then I added a query to select customers from table having amount they spent greater than the average.**
6. **The maximum price of products in the same genre (for example, rock, pop, country, hip-hop). Use GROUP BY to list all the genres and their maximum price.**
   1. **I have joined albums and genre table with genre\_id and used group by to get the required result set.**
7. **List how many customers the system has by location (Country, Province, and City), and then sort them.**
   1. In this query I have joined the two tables’ customers and address and used the count function on customer\_id of customers to get the no. of customers and used group by city for location and arranged in ascending order
8. **List how many products the store has sold for a particular month.**
   1. In this query I have joined two tables orders and order\_items. sum function is used to add all the products sold and by using cast to convert in between the dates to get for a particular month.
9. **List how many distinct albums each singer has.**
   1. In this Query, I have joined three tables albums ,
   2. Albums\_has\_artists and artists to get the no of albums each singer has and arranged by ascending order by singer name.
10. **List how many copies of an album are available of a particular singer.**
    1. -In this query to find the total number of available copies I need to subtract No\_of\_albums from No\_of\_samples as No\_of\_albums indicate the total number of albums and No\_of\_samples display the total number of sold items. As all columns are not in a single table so to get the data from all tables I need to connect three tables using the JOIN statement.Those tables are artist, artist\_has\_albums, order\_items.
11. THE SPECIFIC SCENARIO: -

**1.Filter albums according to instruments used in the album.**

* In this query I need to find the instrument name that is used in a particular album. So, to filter the instrument name with respect to the album name I need album name and instrument name and to get those two things I need to connect two tables using the inner join. Those tables are album, instrument.

**2.Most popular instrument.**

- In this query to get the most popular instrument. I need to get the instrument name that is used in many albums. By using the MAX function, I get the name of the instrument that is used in most of the albums.

1. Then we work on the presentation.

**3.Peer evaluation**

**Table

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**4.PowerPoint presentation**

Powerpoint is also created to present in the class.

**Conclusion**

To conclude the music store database has been created successfully that will display the no of albums, instruments used in the albums, orders total in the store and the data related to the music album i.e. release date, sold items, available items, discount, purchases, name of the artist, price etc.

**Appendix**

**Diagram

Description automatically generated**

Iworked on three tables and one linking table:- artist,artist\_has\_albums,Albums,Genre.

* Primary\_keys:-Artist\_id ,Album\_id,Genre\_id

The purpose of making these fields unique is to avoid the Duplicacy in data and also to identify the artist ,album, genre with their particular id.Many persons has same name but their id ‘s are unique that give a identification to that person.

* Foreign\_keys:-Artist\_idartist,Albums\_Album\_id,Genre\_Genre\_id

Foreign\_keys are created automatically when we use a many to many relationship between two tables then the foreign keys are displayed in a linking table. The purpose of these keys is to create link between 2 tables.