## 

Student No. 2224755

SWE4203 Databases 2022

NAVEED SABIR

20 DECEMBER, 2022



Contents

[0](#_Toc122454573)

[**DIAGRAM** 3](#_Toc122454574)

[**SCHEMA** 4](#_Toc122454576)

[**VALUES** 5](#_Toc122454577)

[**QUERIES** 6](#_Toc122454578)

[PERSONAL STATEMENT 14](#_Toc122454579)

[Figure 1 3](#_Toc122455254)

[Figure 2 4](#_Toc122455255)

[Figure 3 5](#_Toc122455256)

[Figure 4 6](#_Toc122455257)

[Figure 5 6](#_Toc122455258)

[Figure 6 6](#_Toc122455259)

[Figure 7 7](#_Toc122455260)

[Figure 8 8](#_Toc122455261)

[Figure 9 8](#_Toc122455262)

[Figure 10 9](#_Toc122455263)

[Figure 11 9](#_Toc122455264)

[Figure 12 10](#_Toc122455265)

[Figure 13 10](#_Toc122455266)

[Figure 14 11](#_Toc122455267)

[Figure 15 11](#_Toc122455268)

[Figure 16 11](#_Toc122455269)

[Figure 17 12](#_Toc122455270)

[Figure 18 13](#_Toc122455271)

[Figure 19 13](#_Toc122455272)

## **DIAGRAM**

Diagram used for Assignment 2.

## **Diagram Description automatically generated**

Figure

## **SCHEMA**

Schema used for assignment 2. It is normalised in 3NF.

Graphical user interface, text, application, table, Excel

Description automatically generated

Figure

This is the schema that was given to us; however some changes were applied in the ‘FAULT\_VOLUNTEER’ table, ‘DURATION’ was added. It helps us verify information in a more precise way and it helps solve one of the queries.

## **VALUES**

After the creation of the database (in uploaded files), adding values to the tables that match the data type and the limit of digits is the next step.

For example, adding values to the ‘VOLUNTEER’ table.

Table

Description automatically generated

Figure

The same way, all tables created in the database had values added to them following the same process.

## **QUERIES**

**QUERY 1**

A list of all bikes accepted for repair past, present or for a given month

Graphical user interface, text, application

Description automatically generated

Figure

**QUERY 1 OUTCOME**

The results show all bikes repaired based on past, present and future of a given date.Graphical user interface, text, application

Description automatically generated

Figure

Graphical user interface, text, application

Description automatically generated

Figure

Graphical user interface, text, application

Description automatically generated

Figure

**QUERY 2**

A list of spare parts used for a given month and the cost of each part.

A picture containing graphical user interface

Description automatically generated

Figure

**QUERY 2 OUTCOME**

Outcome shows every part’s id, the repair, the cost of the repair and the date or month the part was used. In order to get the date, the part was used we need to look at when the repair happened.

Table

Description automatically generated

Figure

**QUERY 3**

A total of spare parts used for a given month, showing how many of each type and total and average cost of the repairs.

To get the average, total of the costs and the amount of parts used; AVG, COUNT and SUM was used. Each on calculates the amount and stores it in tables created while executing the query.

Text

Description automatically generated

Figure

**QUERY 3 OUTCOME**

The table shows all the information, the part type used, the total parts of that type, the average cost of the part, the total cost of the part, the amount and finally the repair date.

Graphical user interface

Description automatically generated

Figure

**QUERY 4**

A total of the amount of time each month each volunteer spends on repairs and the average time taken for each repair by each volunteer.

The extra attribute to the ‘FAULT\_VOLUNTEER’ table was added here because it makes the time of each repair more precise.

Text

Description automatically generated with medium confidence

Figure

**QUERY 4 OUTCOME**

After a date and month was chosen, it is clear how long, which volunteer spends how much time in total and average. It also shows which repairs the volunteer works on in the given month by also specifying the date.

Table

Description automatically generated with medium confidence

Figure

**QUERY 5**

A list of all cyclists who attended classes showing which class they attended and when, filterable by month or volunteers.

Filtered by volunteers.

Text

Description automatically generated with low confidence

Figure

Filtered by month.

Text

Description automatically generated with medium confidence

Figure

**QUERY 5 OUTCOME**

Which cyclist attends which class, on which date and filtered by which volunteer is delivering the class.

Table

Description automatically generated

Figure

Which cyclist attends which class, on which date and filtered as to which month the class was delivered.

Table

Description automatically generated

Figure

**QUERY 6**

A count of how many cyclists attended each class in a given month, optionally grouped by volunteer who delivered the class.

COUNT used as it counts and shows the number of cyclists that attended each class. GROUP BY statement usually used while calculating the average sum or count so the outcome can be grouped and shown.

Initial thought was to use INNER JOIN, however this seems to be an easier way to get it done.

Text

Description automatically generated

Figure

**QUERY 6 OUTCOME**

Outcome shows the number of cyclists attended which class on which date and which volunteer was delivering that class.

Graphical user interface, text, application, table

Description automatically generated

Figure

## **PERSONAL STATEMENT**

Reflection as a whole/improvements/insight on assignment

Starting off by creating the database and creating all the tables making sure they are in the correct order so each table and each attribute can fetch data for primary and foreign keys. Adding values to the tables and the database created and making sure each value is appropriate for the type of data inserting into it so it can work. Again, making sure there are values added to some tables before some others, so the values can synchronize and work properly. Some of the difficulties faced were mostly mistakes that are easily corrected and noticed such as not adding an attribute that exists in the schema but not in the tables. Most of the time was spent on making the queries work which requires attention to the details of the values already entered and inserted.

One of the most important things to complete all queries is to try all the commands and statements in SQL that appear on the presentation slides and then attempt to complete the queries that are required in the assignment. It is easier to see which way is more effective on delivering a proper query based on what I am asked to do. Using functions or nested queries would be something I would focus on and improve. Using the GROUP BY statement was hard just because I kept getting an error, I did not know how to solve but after some research online I realized there was nothing wrong with what I was trying to execute but it was the workbench system, I had to change some settings from the MYSQL server file so the GROUP BY command can run. It seems like older versions of MySQL did not have that problem, but latest versions do.

In some queries for example the 4th query posted above, I was not able to calculate the time that each volunteer spends on a repair so adding time to the DATE format is what came to mind but later when I would run the query the numbers I would get, would not look reasonable. I decided to add an extra attribute called DURATION which made it a lot easier to calculate the time since it was a completely different value, and it was easy to gather all the information inside of it.

Plenty of time I spent working on the last query which had me thinking as to how I can connect to tables that are not related at all. I realized I could simply SELECT the information I wanted from each table and then group the information together by using GROUP BY.

Creating a database is a lot easier in my opinion that designing it since creating the diagram and then normalising the schema later is the main creation of a database in my eyes. I am sure many would agree with me, and I feel like I have done a decent job by creating a database on MySQL and running queries effectively.