

Assignment 1

by Charlie Keirl

Submission date: 24-Apr-2023 02:47PM (UTC+0100)

Submission ID: 203203686

File name: 41750_Charlie_Keirl_Assignment_1_486136_345977744.docx (304.65K)

Word count: 2241

Character count: 10580



L4 – INTRODUCTION TO WEB AND DATABASE CONCEPTS - ASSIGNMENT 1

Charlie Keirl (2103411)

Contents

Introduction	2
Task 1 – Entity Relationship Diagram	2
The Tables	2
The Relationships	3
Task 2 - My Database	3
The Tables	3
The Forms.....	4
Queries.....	6
Reports.....	7
Macros	8
Brief User Guide	9
References.....	10

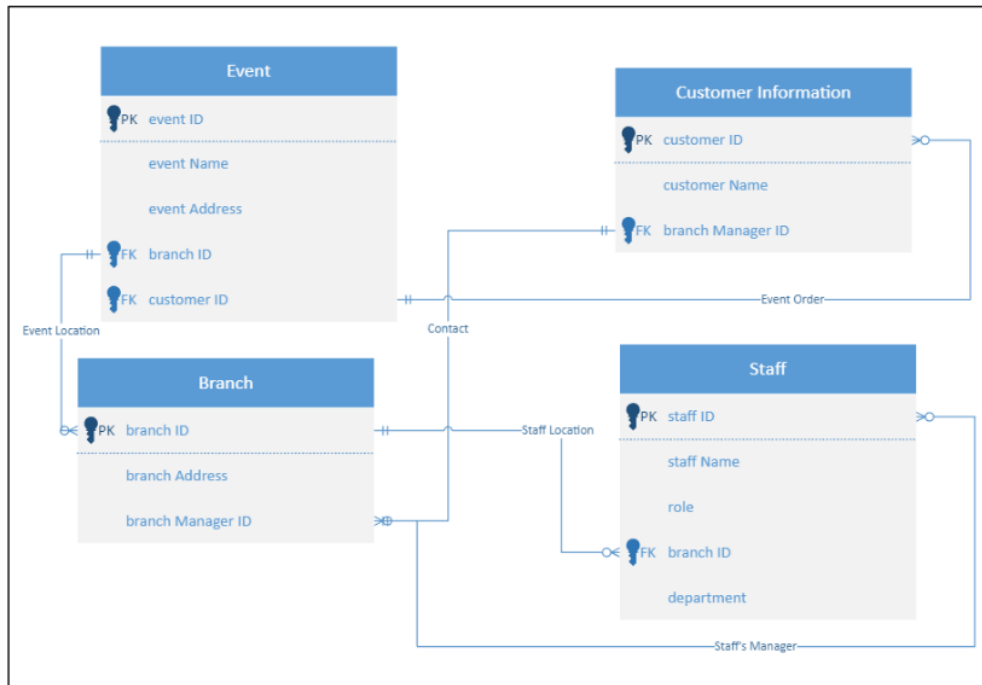
Introduction

I have been tasked with creating a database for Northern Lights pvt Ltd, a company based on event organisation. Within this database, it will need to be able to hold information about customers and their events and about each branch and their staff. The database is required to have queries and reports as well as forms so that a user can input data to the tables easily, all to be access via a main menu upon opening the database.

Task 1 – Entity Relationship Diagram

When it comes to creating a new database, it is essential that we create an Entity Relationship Diagram so that we can better understand what we will need for our database. ERDs helps us to visualise our design and shows us the relationships that we must develop between the tables in our database. [1] It gives us a basic structure for our database and a good starting point within the design process.

Based on the brief we were given, here I have created an ERD relating to the relationships between the different tables required for our database.



The Tables

I have created four tables for this database. The table we have are an Event, a Customer Information table, Branch and Staff tables. For the Event table, our primary key is set as "event ID" and we have two foreign keys: "branch ID" and "customer ID". For our Branch table our primary key is set to "branch ID". Our Customer Information table has the primary

key set to “customer ID” and a foreign key called “branch Manager ID”. The final table called Staff has a primary key called “staff ID” and a foreign key called “branch ID”.

I decided to make our database in third normal form (3NF). 3NF is to keep our database in its most efficient state by ensuring that there is no “transitive dependency” meaning that there is no dependency on non-primary attributes [2]. By using 3NF, this will massively reduce the risk of data anomalies as well as the duplication of data and to keep the referential integrity of the data. [3]

The Relationships

Between these tables we have five relationships. Our first one was between the tables Event and Branch where we have the branch ID in the event table. I added this here as we needed to have a link between said tables so that we can see what events are happening with what branch of the company. Another relationship that is connected to our Event table is one with the Customer Information table. We have this relationship as the Event table will require the customer information too.

The Customer Information table has a second relationship which connects to the Branch table. This is because as stated in the brief, each customer will require a branch manager as a point of contact. The Branch table has another two relationships, both going to the Staff table. One of these relationships has the branch ID in the Staff table so that we can see what staff works at what branch and another relationship that connects the branch Manager ID to the staff ID so that we can see what staff has what manager.

Task 2- My Database

Using Microsoft Access, I have created a database for “Northern Lights pvt LTD” with the aid of the ERD I created and discussed earlier on. I started off with creating the four tables required for the database: Branch, Customer Information, Event and Staff. With each of these tables, I also created the fields needed each with their own data type to ensure that the correct data is implemented. With each of these tables, I also added some sample data so that we can see how this database works.

The Tables

The first table we have is Branch and within this table we have the following fields: branch ID, branch Address and branch Manager ID. This is the table that is used to document what branch has what ID as well as which manager is assigned to said branch.

branch ID	branch Address	branch Manager ID
1	Hooligan Lane, Carmathen	2
2	Happy Street, Swansea	5
3	Unhappy Road, Neath	3

Our second table is the Customer Information table. This table contains the following fields: customer ID, customer Name and branch Manager ID. This table is used so that we can track the customers we will receive as well as what manager they will have as a point of contact.

Customer Information		
customer ID	customer Name	branch Manager ID
1	Bobby Bojangles	2
2	Maul Tano	3
3	John Wickes	5
4	James Williams	3
5	Luke Starkiller	5
6	Jennifer Jeffinston	2

The third table I created was the Event table. This table contained the following fields: event ID, event Name, event Address, branch ID, customer ID, profit and event Date. Here we can track all the events that have taken place across all of the branches as well as know where they took place, how much money each one made and when said event was held.

Event						
event ID	event Name	event Address	branch ID	customer ID	profit	event Date
2	Rock The House	Carmarthen	1	1	£2,000.00	15/04/2023
3	The Mandalore Boo	Neath	3	2	£12,000.00	12/04/2023
4	Parabellum	Swansea	2	3	£699.00	24/03/2023
5	Brain Fun Magic	Neath	3	4	£5,000.00	25/12/2022
6	Rebelz	Swansea	2	5	£653.00	07/04/2023
7	Left 4 Fun Party!	carmarthen	1	6	£56.00	17/04/2023

My final table was the Staff table containing the following fields: staff ID, Staff Name, role, branch ID and department. In this table we can access all of the information we might need about the staff at each branch including what department they work in and at what branch.

Staff				
staff ID	staff Name	role	branch ID	department
1	Danny Thomas	team member	1	tickets
2	Daniel James	manager	1	admin
3	Ben Sonic	manager	3	admin
4	Billy Whizzy	staff	3	catering
5	Phillip Schofie	manager	2	admin
6	Nick Valentine	team member	2	sound

The Forms

For a user to be able to interact with our database, forms are required to give the user that interface between them and the database. Within my own database, I have used two types of forms, bound and unbound. Bound forms are forms that are linked to tables and queries which could alter your database by adding/ deleting data whereas unbound forms are used to allow the user to interact with your database via buttons and labels but will not alter your data in anyway. [4]

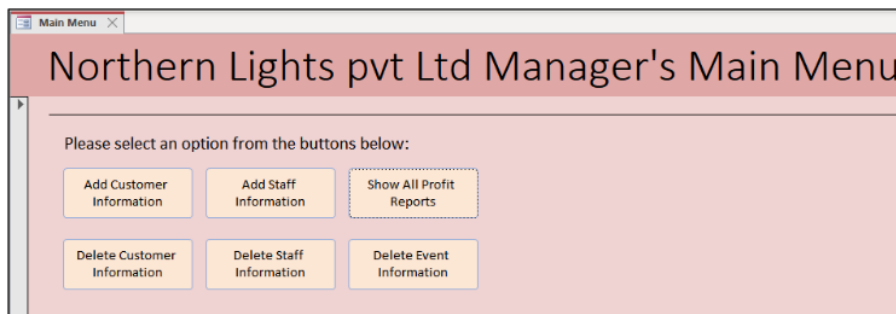
The first form I created was a bound form named "Customer Information Form". This form allows the user to be able to create and insert customer information as well as event information into our database. Every time the user wants to enter in new customer information, it will ask the user to enter data into the fields for the table Customer Information as well as then enter data for the fields for my Event table. Because of the relationship with one another, we can enter all of this information for multiple tables in one form and the data will then be sent off to the correct table and then stored.

Here we can see the form I created for Customer Information:

event ID	event Name	event Address	branch ID	profit	event Date
2	Rock The House	Carmarthen	1	£2,000.00	15/04/2023
*	(New)			0	£0.00

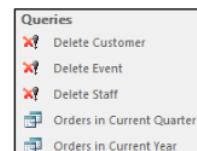
The second form I created was named "Staff Form". Just like with the above form, this form allows us to enter the details of a new staff member and for it to then be stored within the Staff table.

The third form I created was an unbound form named "Main Menu". Unlike the other two forms, this form will not alter any of the tables or the data within them. This form simply is used as a menu for the user to do what they wish to the database. On this form I have added a title, a label and buttons. Every time the user pushes a button, whatever the button description displays will happen. For example, if the user selects the button that will add customer information to the database, the Customer Information form will be displayed for the user to then be able to enter in the data on their customer. This is all possible thank to the use of Macros.



Queries

When it comes to databases, finding out all of the information you need could take a long time so instead of hunting down your data table by table, we can use queries and reports to make our lives a lot easier. Queries are a tool we can use to find certain pieces of data easily by setting parameters and then showing the data that sits within those parameters. [5]



Within my own database, I created five queries in total. The first one I made was to show how many orders were placed within the current quarter. As you can see from the image below, this query will be taken from the Event table and take the data from the fields "event ID", "event Name" and "profit". The query will also take into account the current date that the query is being used and the field "event Date" and only take the information that sits nicely between within the current quarter of the year we are in.

```
SELECT Event.profit, Event.[event ID], Event.[event Name]
FROM Event
WHERE Year(event.[event Date]) = Year(Now()) And DatePart("q", event.[event Date]) = DatePart("q", Now());
```

Our results from this query are here:

event ID	event Name	profit
2	Rock The House	£2,000.00
3	The Mandalore Boo	£12,000.00
6	Rebelz	£653.00
7	Left 4 Fun Party!	£56.00

The second query was very similar to the previous one. It used the exact same table and fields however instead of taking data from within the current quarter, it takes all the data from the current year. Below are the results:

```
SELECT Event.[event ID], Event.[event Name], Event.profit, Event.[event Date]
FROM Event
WHERE Year(event.[event Date]) = Year(Date());
```

event ID	event Name	event Date	profit
2	Rock The House	15/04/2023	£2,000.00
3	The Mandalore Boo	12/04/2023	£12,000.00
4	Parabellum	24/03/2023	£699.00
6	Rebelz	07/04/2023	£653.00
7	Left 4 Fun Party!	17/04/2023	£56.00

My other three queries are used to delete data from within your tables. The first one (named "Delete Customer") is so that a user can delete customer information by typing in the customer ID and then the query will then delete any data saved to that ID.

```
DELETE [Customer Information].*, [Customer Information].[customer ID]
FROM [Customer Information]
WHERE ((([Customer Information].[customer ID])=[Please enter Customer ID you wish to delete:]]);
```

The second one is called "Delete Event", and this does the same as the last one however instead of deleting the customer using the customer ID, it instead deletes and event by deleting the event ID and all data attached to it.

```
DELETE Event.*, Event.[event ID]
FROM Event
WHERE (((Event.[event ID])=[Please enter the Event ID you wish to delete:]]);
```

Just like with the last two queries, this query called "Delete Staff" simply deletes any staff details attached to the staff ID that wishes to be deleted.

```
DELETE Staff.*, Staff.[staff ID]
FROM Staff
WHERE (((Staff.[staff ID])=[Enter Staff ID you wish to delete:]]);
```

Reports

Using these queries, we can now produce reports based on the information given to us from said reports. Reports are basically just queries that are put into a document format. Within my own database I created two reports: one for the orders in the current quarter and another for the orders in the current year. All that is needed is to select the query you wish to make a report from and then simply click "create report" and its all done for you. Here are what my two reports look like:

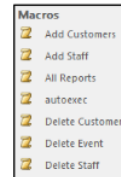
event ID	event Name	profit
2	Rock The House	£2,000.00
3	The Mandalore Boogie	£12,000.00
6	Rebelz	£653.00
7	Left 4 Fun Party!	£56.00
		£14,709.00

event ID	event Name	profit	event Date
2	Rock The House	£2,000.00	15/04/2023
3	The Mandalore Boogie	£12,000.00	12/04/2023
4	Parabellum	£699.00	24/03/2023
6	Rebelz	£653.00	07/04/2023
7	Left 4 Fun Party!	£56.00	17/04/2023
		£15,408.00	

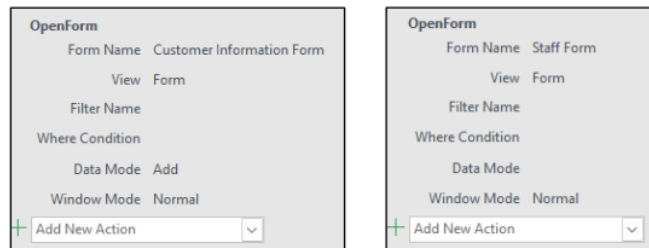
Macros

Within MS Access, Macros are a tool we can use that will add some form of automation to your database. Depending on what you chose to do with them, they can help to automate tasks for users and can also add functionality to controls such as buttons and labels. [6]

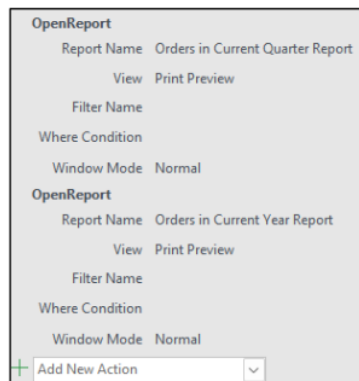
For my own database, the use of macros was essential. As my menu had six buttons, at least six macros were required. I also required one so that my menu would be displayed to the user as soon as they open the database, bringing my total number of macros to seven.



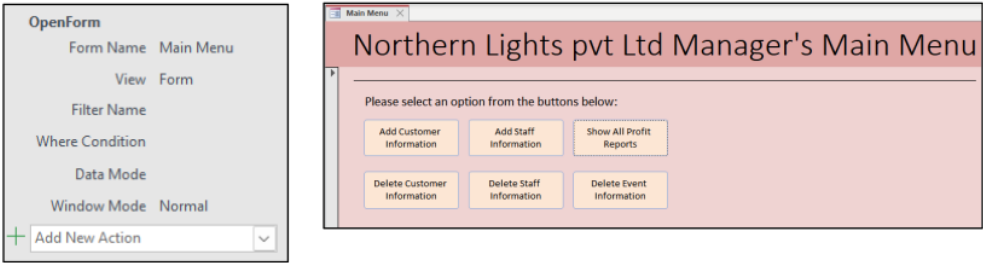
The first two macros I created were called “Add Customers” and “Add Staff”. Both of these were for buttons which would need to bring up the corresponding forms. All I did was create an “Open Form” macro for both buttons and assigned the button clicks to open the Customer Information form and Staff form. Now that the buttons have been assigned, when the user uses the menu and selects a button, the Customer Information form or Add Staff form will now be displayed and ready to be used.



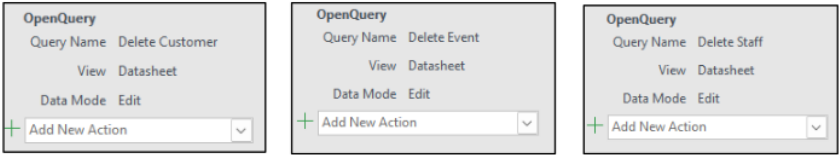
My third macro is used for bringing up both of the reports we created earlier to show the user how much money they made in different time frames. Just like with the other two macros, all you need to do is assign the button click to open up both of the reports at the same time, instead of the forms.



The fourth macro is named “autoexec”. The reason behind this name is that when a macro is called this it will be the first thing the database automatically runs when the database is opened. I assigned the menu form to be for this macro so that now whenever a user opens the database, the menu will be the first thing they see.



My final three macros are linked to the queries about deleting data. We have one for deleting customers, deleting events and deleting staff information. Whenever the relating buttons are clicked on the menu, it will run the macro assigned to the button and then proceed with deleting whichever data the user requires to.



Brief User Guide

When the user opens up the database, they are greeted with the main menu for ease of access. From here, the user can click one of the six buttons available and from there be taken to the corresponding area of the database.

For example, if the user wanted to add a new customer to the database, all they would need to do is to click on the button “Add Customer Information” and it will then open the form to add new customer/event details to the database. To return to the menu, all the user needs to do is to close the form that’s currently open.

References

- [1] Anonymous (). *What is Entity Relationship Diagram (ERD)? | Definition from TechTarget*. Available: <https://www.techtarget.com/searchdatamanagement/definition/entity-relationship-diagram-ERD> [Accessed Apr 19, 2023].
- [2] Anonymous (). *Third Normal Form in DBMS | GATE Notes*. Available: <https://byjus.com/gate/third-normal-form-in-dbms/> [Accessed Apr 19, 2023].
- [3] Anonymous (). *What is Referential Integrity and Why do You Need it?*. Available: <https://intelligent-ds.com/blog/what-is-referential-integrity> [Accessed Apr 19, 2023].
- [4] Anonymous "Introduction to forms - Microsoft Support," Available: <https://support.microsoft.com/en-us/office/introduction-to-forms-e8d47343-c937-44e8-a80f-b6a83a1fa3ae>[AccessedApr 19, 20232023].
- [5] Anonymous (). *Access 2016: Designing a Simple Query*. Available: <https://edu.gcfglobal.org/en/access2016/designing-a-simple-query/1/> [Accessed Apr 21, 2023].
- [6] Anonymous "Introduction to macros - Microsoft Support," Available: <https://support.microsoft.com/en-us/office/introduction-to-macros-a39c2a26-e745-4957-8d06-89e0b435aac3>[AccessedApr 20, 20232023].

Assignment 1

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

61 / 100

Instructor

Generic Assessment Criteria	61
1. Engagement with Literature Skills	
Not Applicable	
2. Knowledge and Understanding Skills	
It was a very good attempt trying to provide a brief description about the scenario. Suitable design methodologies for creating the database system was employed. Identification of different type of users accessing the database system is missing. Setting the branch manager as the primary key in the table is a good approach however his access rights to the system should have been mentioned. Different departments of the organization who might use the database system should have been identified.	5

3. Cognitive and Intellectual Skills	
- Screen shots showing the evidence of tables with the required fields is provided. - Evidence showing the justification for the links between tables is missing. - Verification of third normal form in table design is not been provided. - Some of the screen shots provided for creating tables are missing the design specifications. - usage of an MS access functionality to produce few calculations in forms ,for example tickets sold for an event, number of events books in a room etc... and generating invoice reports are missing. - Some additional querying features and it's screen dumps could have been provided.	1 7
4. Practical Application Skills	
Primary key and foreign key in class diagram are clearly represented however the description about the degree of relationships is missing.	3 4

· Design techniques that you have been studying in class should have been used in creating an Entity Relationship Diagram with Entities, and different types of attributes emphasizing the relationships between tables. This was the requirement for the assignment.	
5. Transferable Skills for Life and Professional Practice	
· All the tasks are well presented and documented however, Proper grammar and punctuation are essential when writing in a professional setting. · Make sure you proofread for any errors before submitting it. · Some of the screen shots along with the description doesn't seem to be relevant. · A brief user guide for navigation between the system is not complete and missing few essential descriptions.	5

PAGE 1
PAGE 2
PAGE 3
PAGE 4
PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11
