



DATABASE ANALYSIS AND DESIGN

Introduction to Web and Database Concepts
– Assignment 1

Danny Thomas (2004365)

Contents

Initial Findings	2
ER Diagram	3
Documentation	4
Additional Access Functionality	6
References	11

Initial Findings

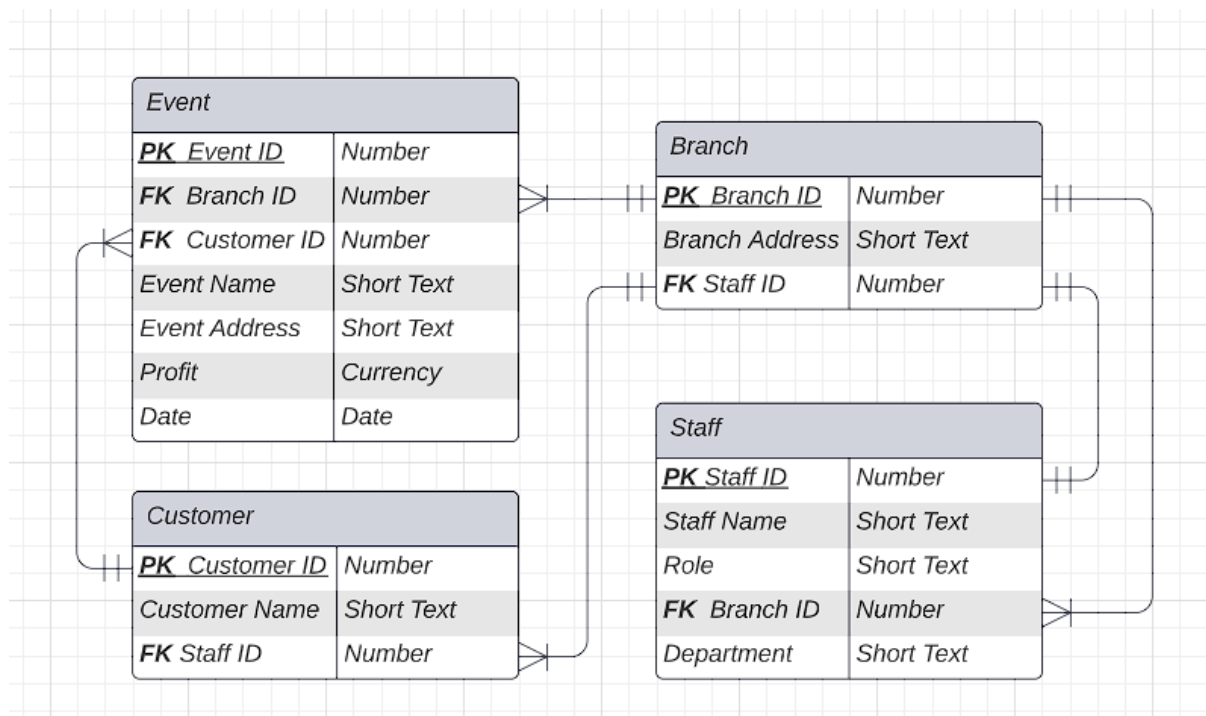
For this assignment we were asked to design a database system for an event organising company. They are moving from a manual system therefore we can build the database on MS access from the ground up. The first step is identifying who will be accessing the database, and from reading the brief we can find that each branch will have to access the database to enter event and customer data that comes from each branch. Next is identifying the departments in each branch, in the brief it lists the kind of roles that they need for their events, therefore it makes sense to declare those as our departments; so our departments will be audio and visuals. Tickets, maintenance, cleaning, and admin which we will apply to each branch manager.

ER Diagram

Now that we have noted who will be using our database software, and what departments we need for the staff, we can start creating our tables and populating them with relevant information.

But before starting that, we must create an ER diagram to plan how the relationships are going to work in our database as we will be creating a relational database; a relational database is a database where the entities within the table have relationships, these relationships are determined through primary and foreign keys. [1]

Primary keys are used as a unique identifier for each record in a table, you can see in the ER diagram below that we have the table “Event” then labelled within the table with “PK” is the primary key, the primary key for each table is also underlined to make it clear. A foreign key is simply a primary key from another table being used in a different table, as you can see again in “Event” we take the primary key from the “Branch” and “Customer” tables and denote them as “FK” for foreign key. This is the foundation of a relational database [2]



Documentation

Below are the four tables I used for the database.

Main Menu × Branch × Customer × Event × Staff ×				
Branch ID	Branch Address	Staff ID	Click to Add	
1	Swansea, 12 Swan Road	1		
4	Neath, 9 Neat Road	4		
5	Carmarthen, 3 Car Road	5		
(New)		0		

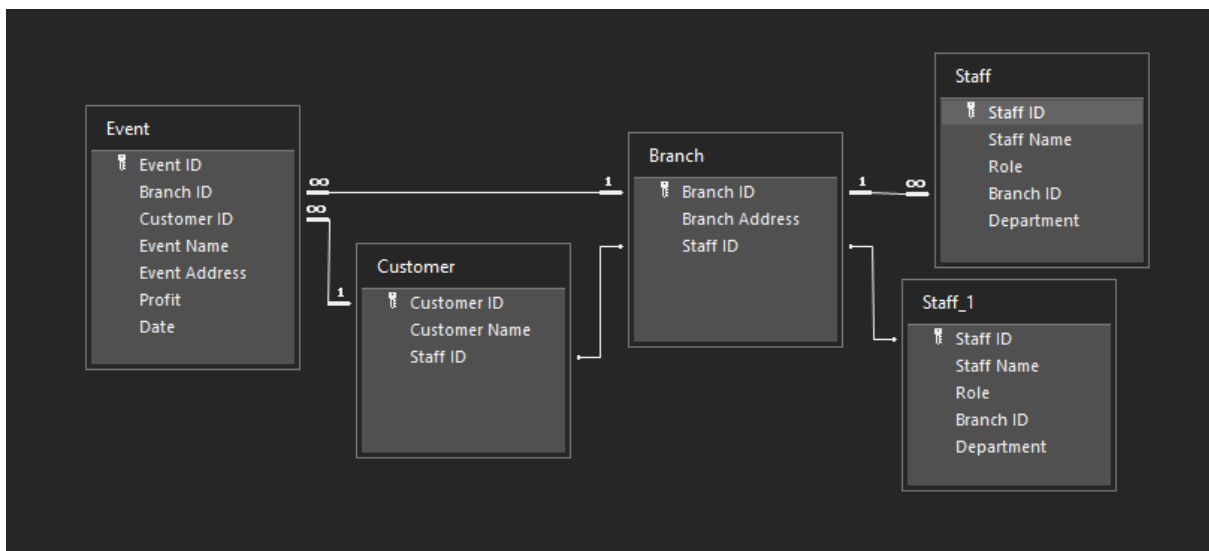
Main Menu × Branch × Customer × Event × Staff ×				
Customer IC	Customer N	Staff ID	Click to Add	
1	Steve Irwin	1		
2	John Lennon	1		
3	Divya Database	1		
4	Gareth Jones	2		
5	Limmy Lime	2		
6	Terry Orange	2		
7	Barry Scott	3		
8	Danny Thomas	3		
9	Gabriella Curl	3		
(New)		0		

Main Menu × Branch × Customer × Event × Staff ×							
Event ID	Branch ID	Customer IC	Event Name	Event Address	Profit	Date	Click to Add
1	1	1	Concert	Neath	£120.00	17/04/2023	
2	1	2	Fesitval	Neath	£200.00	11/04/2023	
3	1	3	Workshop	Port Talbot	£30.00	24/04/2023	
4	4	4	Seminar	Llanelli	£240.00	14/07/2023	
5	4	5	Conference	Carmarthen	£130.00	26/07/2023	
6	4	6	Concert	Singleton	£300.00	06/10/2022	
7	5	7	Workshop	Neath	£70.00	14/09/2023	
8	5	8	Seminar	Mumbles	£190.00	06/08/2022	
9	5	9	Conference	Swansea	£270.00	18/01/2023	
(New)	0	0			£0.00		

	Staff ID	Staff Name	Role	Branch ID	Department	Click to Add
+	1	Charlie Curl	Manager	1	Admin	
+	4	Daniel Bames	Manager	4	Admin	
+	5	James Thomas	Manager	5	Admin	
+	6	Osian Sutton	Staff	1	Audio and Visuals	
+	7	Jessie Thomas	Staff	1	Tickets	
+	8	Georgina Dangerfield	Staff	1	Maintenance	
+	9	Charmaine Asda	Staff	1	Cleaning	
*	(New)			0		

I chose these four tables for the database because the brief required a simple structure with different branches serving customers and having different staff and hosting different events therefore there is no need for an incredibly complex data structure.

The fields I chose for each record in each table are also simple, but I ensured that relevant primary and foreign key information was included so that proper relationships could be made. Each table has all the relevant information needed to complete an event and view what staff work at what branch and what they're role is.



From the relationship diagram above we can see how the data relates to each other and the different relationship used to ensure the database functions correctly. I chose a one-to-many relationship for "Events" to "Customer" and "Branch" because there will always be many events but each event can only take place in one branch and pertain to one customer. Likewise for "Branch" to "Staff" where I used another one-to-many relationship as one branch can have many staff.

Additional Access Functionality

Lastly is the additional functionality that MS Access offers to enhance the user experience, after creating all our tables and relationship I created forms that allow us to enter data into the database with a more user-friendly form shown below:

Branch

Branch ID: 1

Branch Address: Swansea, 12 Swan Road

Staff ID: 1

Customer

Customer ID: 1

Customer Name: Steve Irwin

Staff ID: 1

Event ID	Branch ID	Event Name	Event Address	Profit	Date
1	1	Concert	Neath	£120.00	17/04/2023
(New)	0			£0.00	

Record: 1 of 1

Staff

Staff ID: 1

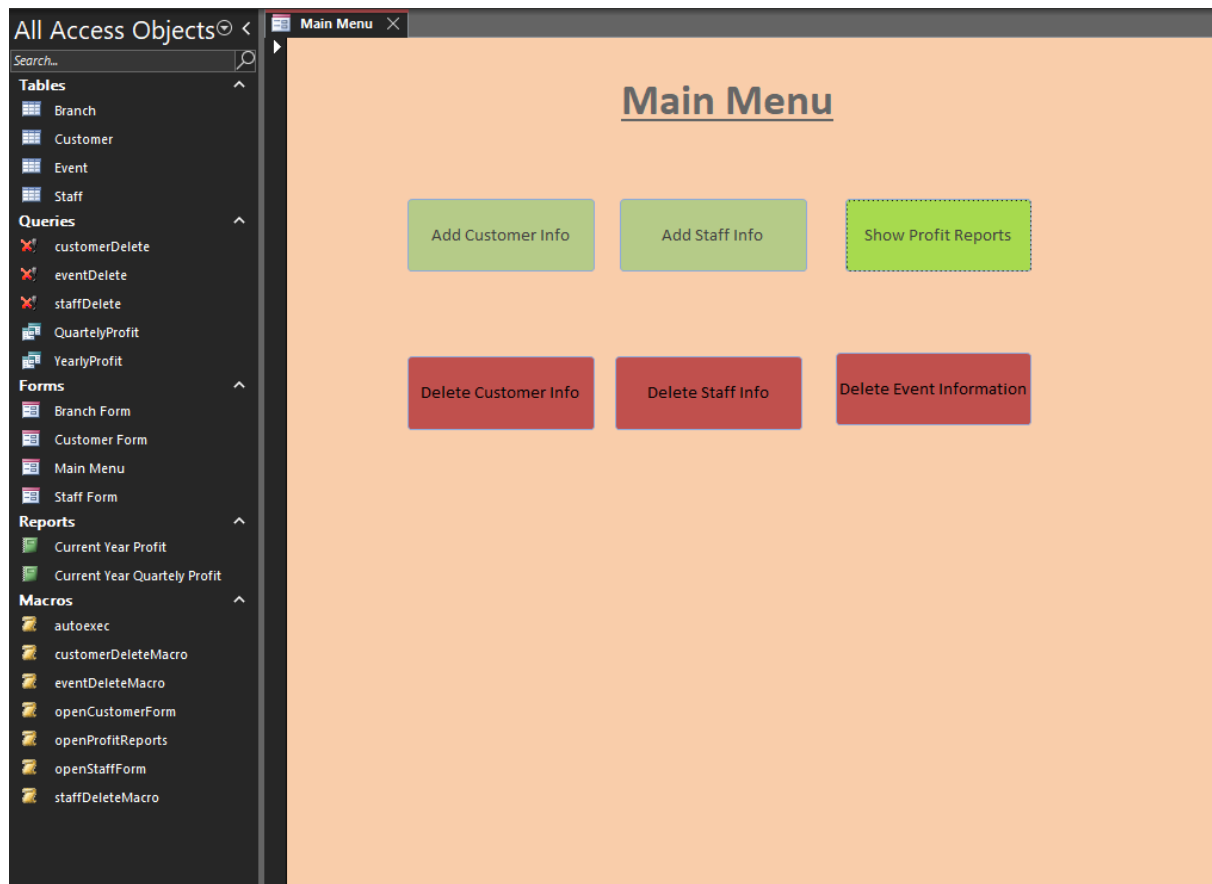
Staff Name: Charlie Curl

Role: Manager

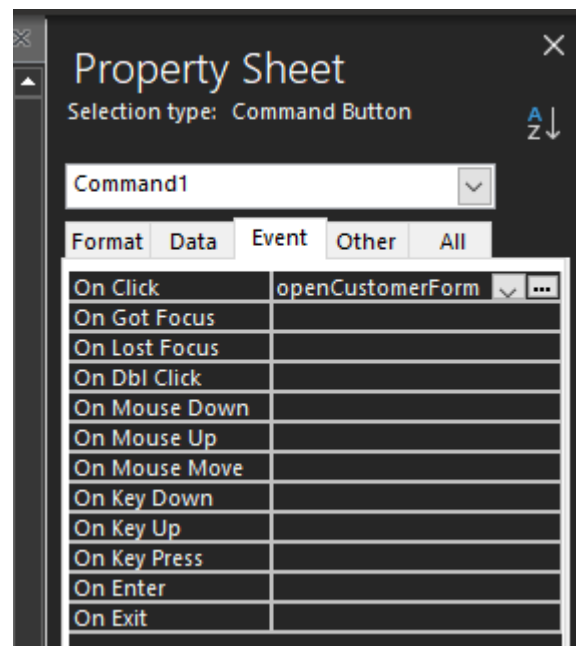
Branch ID: 1

Department: Admin

After I created the forms, I created a Main Menu to make accessing all the forms very easy with the use of buttons.



All these buttons work through the property sheet tab which allows us to link the buttons to macros that do commands such as opening the correct form, like the first two buttons on the menu.

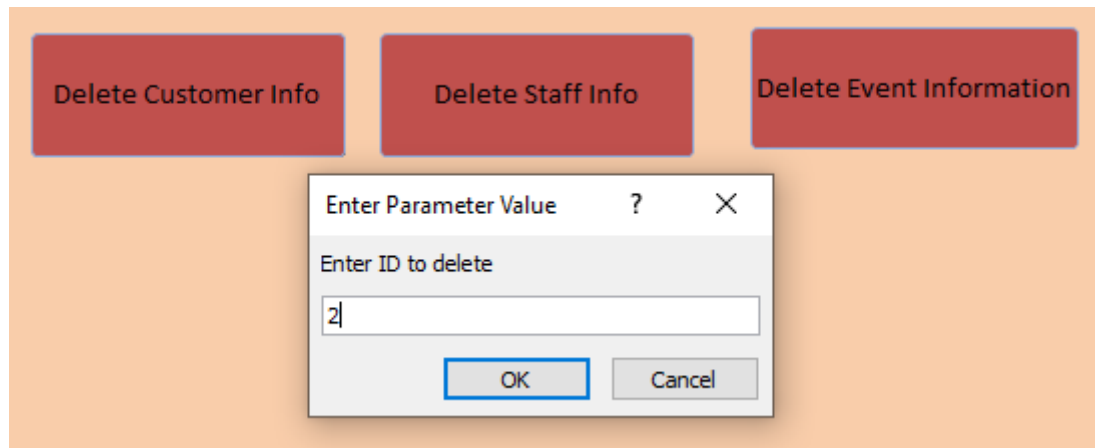


This is done similarly with the delete buttons except they have to be done through queries, then connected to the button through macros instead, as shown below:

Field:	Customer.*	Customer ID
Table:	Customer	Customer
Delete:	From	Where
Criteria:		[Enter ID to delete]
or:		

As you can see for the criteria I have entered the message that appears to the user to make it more accessible.

Here is what shows up when a delete button is pressed:



In this example I pressed the “Delete Customer Info” button and entered “2”.

	Customer ID	Customer N	Staff ID	Click to Add
+	1	Steve Irwin	1	
+	2	John Lennon	1	
+	3	Divya Database	1	
+	4	Gareth Jones	2	
+	5	Limmy Lime	2	
+	6	Terry Orange	2	
+	7	Barry Scott	3	
+	8	Danny Thomas	3	
+	9	Gabriella Curl	3	
*	(New)		0	

This would delete ID number 2 in the customer table which in this case would be “John Lennon”.

Another macro I used was “autoexec”, this is a macro that instantly runs when the database is opened, in this case I made it so that the main menu form opens when the database is opened so that the user is greeted with the buttons to edit the database instantly and don’t have to navigate MS Access or have any prior knowledge on databases.

Another important part of the brief was the ability to view yearly and quarterly profits, I did this using some very basic SQL scripts that worked out the total profits of the current year, and calculated what quarter of the year we were in and gave the total of the quarter.

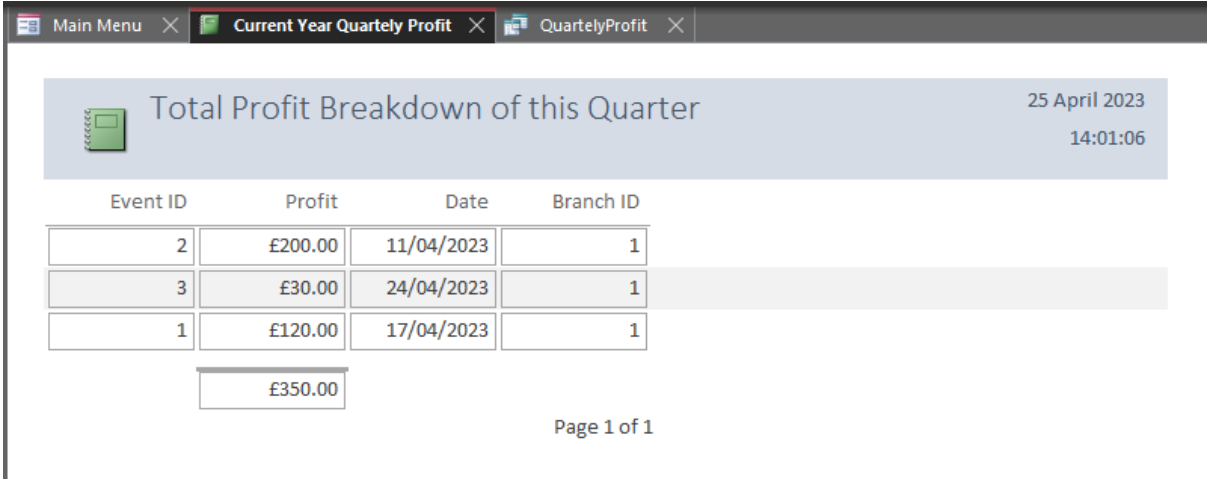
Event ID	Profit	Date	Branch ID
2	£200.00	11/04/2023	1
3	£30.00	24/04/2023	1
4	£240.00	14/07/2023	4
5	£130.00	26/07/2023	4
7	£70.00	14/09/2023	5
9	£270.00	18/01/2023	5
1	£120.00	17/04/2023	1
£1,060.00			

Page 1 of 1

As you can see it is also presented clearly and gives the event ID if you wanted to find more information on one of the records. Here is the SQL used for this query.

```
SELECT Event.[Event ID], Event.Profit, Event.Date, Event.[Branch ID]
FROM Event
WHERE (((Year([Event].[Date]))=Year(Date())));
```

This is the same for the Quarterly Report where the form is displayed to the user with useful information.

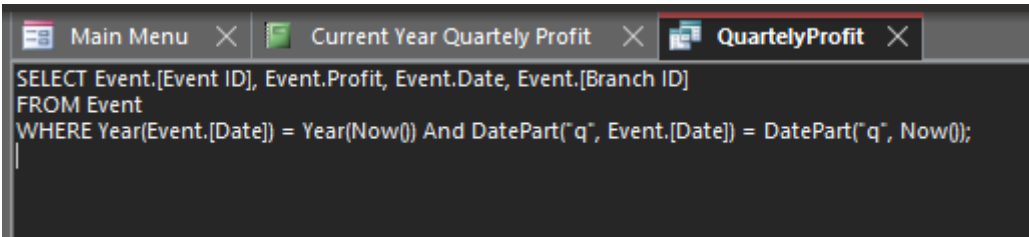


The screenshot shows a window titled 'Current Year Quartely Profit' with a sub-header 'Total Profit Breakdown of this Quarter'. The date and time are 25 April 2023, 14:01:06. The table below lists three events with their respective profits and dates, and a total profit of £350.00.

Event ID	Profit	Date	Branch ID
2	£200.00	11/04/2023	1
3	£30.00	24/04/2023	1
1	£120.00	17/04/2023	1
£350.00			

Page 1 of 1

Here is the SQL used for this query.



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

User Guide

To use the database first open the file, you will be greeted with the main menu as shown above. If you want to add a new staff member to the company, firstly have their full name, role, department, and what branch they are working in on hand. Then click on the "Add Staff Info" button on the main menu, this will take you to the correct form; fill in the information then press enter and the data has been entered correctly. If a mistake is made, go back to the form, and scroll through the staff members using the arrows on the bottom of the screen, or search if you know what information about the staff member is correct. Once you've found the correct record simply edit it and the record in the database will change accordingly.



Record: 2 of 9

This is the same for customer information, when clicking on the button on the main menu it will also open up the record for the event that the customer has where you can edit or add more events. To view the reports of the current year or quarter simply press the "Show Profit Reports" button and it will open the relevant reports for you to read.

References

[1] Anonymous []. *What Is a Relational Database and How Does It Work?*. Available: <https://www.coursera.org/articles/relational-database> [Accessed Apr 24, 2023].

[2] Anonymous []. *Primary Keys in a Database and What Makes a Good Key*. Available: <https://www.lifewire.com/primary-key-definition-1019179> [Accessed Apr 25, 2023].