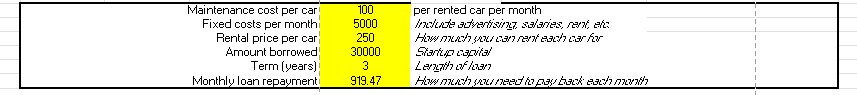
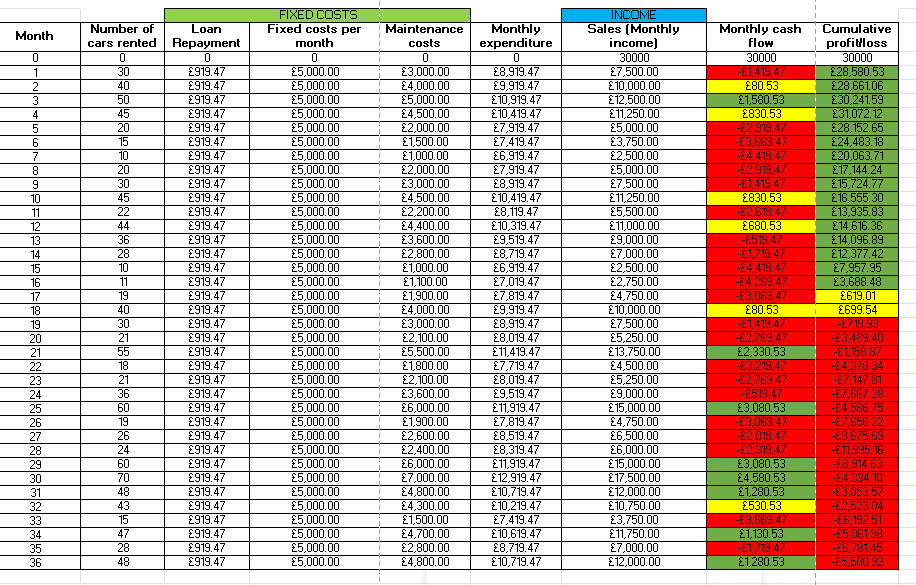
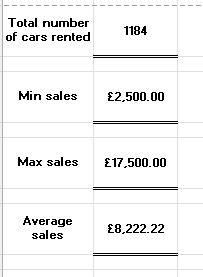
**4204COMP-Coursework**

**Part 1:**

Below is a copy of all the elements from the excel spreadsheet in part 1 of the coursework.

****





My excel document contains the original table and values that were given to us, with the table now filled out and with some conditional formatting and corresponding graphs. The first 2 columns I filled out were the C and D columns. These were filled out by using absolute referencing for the whole column. For the loan repayment we wanted to use the value from E7. This was done using the function =$E$7. Using this function means that if we change the value in the top table the value would be changed throughout. I set this function in C12 and dragged it down to C47 so the same function would be carried on throughout. A similar technique was used in column D, but this time the function =$E$3 was used.

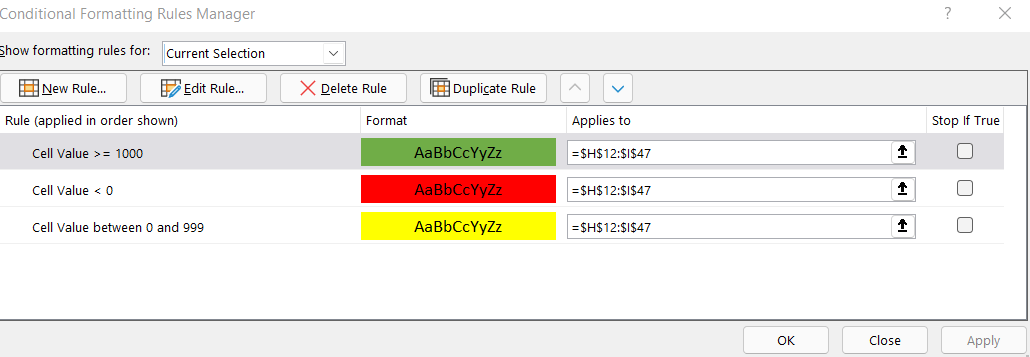
Next we had to do some different calculations for the rest of the columns. Column E was the for the maintenance costs. These costs were calculated by multiplying the fixed fee in E2 by the number of cars that were rented. For example in B12 this was done using the function =B12\*$E$2. This was then dragged down the whole column, which would changed the B value to whichever row it is multiplied by the fixed rate because we used absolute referencing.

Monthly expenditure was a bit more simple as all we had to do for this one was adding the previous 3 columns together, for example in F12 the function was =C12+D12+E12. This was dragged down the whole column and automatically calculated. If any of the values in the top table all of the values would be recalculated accordingly.

For the column G we had to calculate the sales made per month. For this we went back and used the same method as column E which was using the correct value from the top table and multiplying it by the number of cars rented out in column B. Therefore the function in G12 is =B12\*$E$4 and is dragged down the whole column.

For column H we worked out the monthly cash flow which was just simply taking the corresponding value from row G and subtracting the value from F. This function from H12 looks like =G12-F12, and then dragged down the whole column. The final column to complete is the cumulative profit/loss in column I. To work this out we had to take the previous number from the column and add the number from column H. For example we started with 30,000 so we take the first month’s cash flow from H12 and add it to I11 to get I12. This was then dragged down the column to show us the cumulative profit/loss.

For the final 2 columns we were also tasked with conditionally formatting them to show instances were there was profits or loss using colour. Green was to represent a profit of £1000 or more, yellow to show a profit up to 1000, and red to show any loss. To do this I highlighted these 2 columns and added some conditional formatting rules. These rules can be seen below,



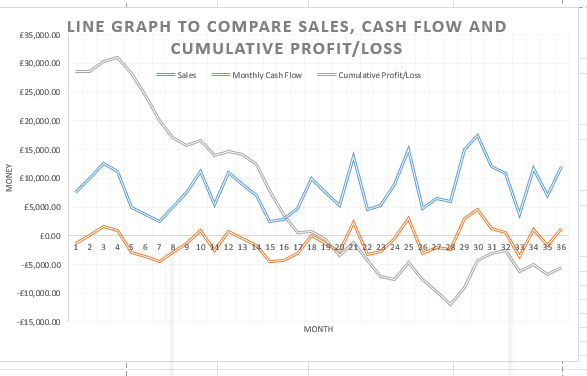
The first rule I made was if the value in the cell was greater or equal to 1000, the cells back colour would be made green. I then made 2 rules which were similar, the first being if the value of the cell was below 0 then to change the back colour to red, and if the cell value was between 0 and 999 then the back colour would change to yellow. As you can see in the rules this only applies to columns H and I, between rows 12 to 47 as these were the values we wanted to format.

Now we are finished our table there are still a few more values that need to be calculated. The first value we need to find is the total number of cars that were rented and put it into B50. To work this one out I used the sum function, followed by the B column as this was where the cars rented where put. This function looked like, =SUM(B11:B47) as it was only to take the values for B within these rows and add them.

The next 2 values we need to calculate are similar as they are both from the sales columns. First we needed to find the minimum value from the sales column, and then the maximum. To find these values we used the min and max functions for the same boundaries which was the G column. These functions would look like, =MIN(G12:G47) and =MAX(G12:G47) respectively. A quick manual check shows that these are indeed the max and min values in the column.

The final calculation we need to work out is the average sales that were made during the 3 years. To do this we use the average function which will automatically take all the values we give in and give us their average. As we wanted to know the average sales we gave it the G column again and asked it to work out the average. This looked like this, =AVERAGE(G12:G47).

Our final task was to create 2 graphs from the table, the first of which is a line graph that will show us the sales, monthly cash flow, and the cumulative profit/loss over the 3 years. To do this I highlighted the 3 columns and inserted a line graph I thought would best represent the values. To complete the graph I made sure to add a correct title, legend and axis. The also made sure to change the colours of the lines so they were more easily distinguishable from each other. The end product looked like this,



The second graph was to be a column graph to show the sales throughout the 3 years. To do this I highlighted the sales column and inserted a column chart I felt was best suited, which was a chart with clear axis and spacing in between the columns. Just like the line graph I made sure to add a title and labels for the axis. The end product looked like this,

A picture containing chart

Description automatically generated

**Part 2:**

In this section we have been tasked with putting all of the rental information into a database, and then show this data in 3NF. First off we have been told to make these assumptions about the data,

Text

Description automatically generated

To start off we will take the table given to us and put it through 1NF. Below is the original table that we will normalise,

Table

Description automatically generated

Some issues we can see off the bat are that there are some cells that hold multiple values. For 1NF we need to make sure that this is corrected, along with making sure that the data is stored logically. Below is the first normalisation of this data,

**1NF:**

**Customer(Customer ID, Customer First Name, Customer Surname, Customer Age, Customer Address, Customer City, Customer Postcode, Customer Phone Number, Customer Email)**

**Customer\_Rental(Rental ID, Rental Start Date, Rental End Date, Rental Cost, Vehicle ID, Vehicle Make, Vehicle Model, Vehicle Colour, Vehicle Year, Category ID, Category Desc, Category Rental Rate)**

The next step is to normalise again into 2NF. To do this we need to make sure there is no partial dependency and that there are unique entities for the data,

**2NF:**

**Customer(Customer ID, Customer First Name, Customer Surname, Customer Age, Customer Address, Customer City, Customer Postcode, Customer Phone Number, Customer Email)**

**Customer\_Rental(Rental ID, Rental Start Date, Rental End Date, Rental Cost)**

**Customer\_Vehicle(Vehicle ID, Vehicle Make, Vehicle Model, Vehicle Colour, Vehicle Year, Category ID, Category Desc, Category Rental Rate)**

Finally we need to normalise on last time to get the 3NF. There must be no transitive functional dependency for it to be in 3NF.

**3NF:**

**Customer(Customer ID, Customer First Name, Customer Surname, Customer Age, Customer Address, Customer City, Customer Postcode, Customer Phone Number, Customer Email)**

**Customer\_Rental(Rental ID, Rental Start Date, Rental End Date, Rental Cost)**

**Customer\_Vehicle(Vehicle ID, Vehicle Make, Vehicle Model, Vehicle Colour, Vehicle Year)**

**Customer\_Category(Category ID, Category Desc, Category Rental Rate)**

It is now time to add these tables to access. Below are these tables,

Table

Description automatically generated Application, table

Description automatically generated

Table

Description automatically generated Graphical user interface, application, table

Description automatically generated

**2.2:**

Now we are moving onto some diagrams, starting off with a conceptual entitiy relationship diagram, using crows foot notation, which can be seen below, Chart, box and whisker chart

Description automatically generated

**2.3:**

The next diagram we need to show is a Logical ERD, continuing to use Crow’s foot notation,

Graphical user interface, application

Description automatically generated

**2.4:**

Finally, we have to use the last diagram to make a physical ERD,

Graphical user interface

Description automatically generated with medium confidence