**Computer Repair Site**

My website allows a user to add customers update their details remove customers (only if they are not currently in a repair) and list all their details. I have a section in my site for Repairs where a customer that has been created can log a repair where they describe the problems that they have with there computer. The data is then saved in the repairs table.

The Log function sets the default total cost to 30 euro as a standard service charge. It also records the date at which the Repair was logged as well as setting the status of the repair to (‘**L**’)which means that the repair has been logged.

Once the repair has been logged the repair staff can add parts to the repair. The site validates all the information checking to see if the repair ticket ID is valid and that its status is logged, whether the parts status is available(‘**A**’) (‘**D**’ means delisted) and that they have at least 1 piece in stock. This function also makes sure that the qty value is an integer and that none of the fields are left blank. I also check to see that the repair status is logged **OR** Commenced ***(I allowed to add parts if the repair is commenced in case a repair person needed more parts than they first anticipated)*** as if a repair is completed or finished then there should be no need to add anymore parts to it. When the validation is complete it records the data in the repair items table. This function also adds the cost of the part multiplied by the total Qty to the total cost of the repair. It also updates the quantity of parts in the parts table. If any of the validation conditions are not met, then an appropriate error message will be displayed.

After all the parts are added then the repair can be commenced. This is a simple function signalling that the repair is currently underway. This is a simple piece that simply validates that the repair Ticket ID is a valid ID, that the field is not left blank and that the status is logged. Once the validation is complete the repair status is set to (‘R’) meaning being repaired (I know that ‘**C**’ was probably a better status choice however I wanted to reserve ‘**C**’ for complete).

When the repair is finished then the function acts very similar to the commence function in that it verifies the data however it checks that the status of the repair is equal to (‘**R**’) instead of (‘**L**’).

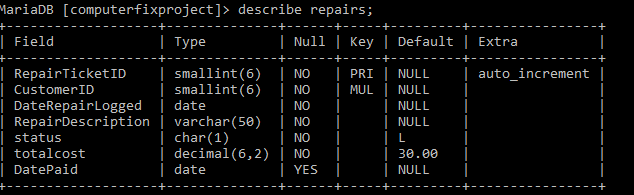
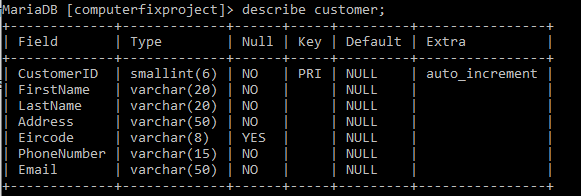
Finally, the customer can pay for the repair. The repair ID is validated, and all fields are checked to see if they are not blank. The status of the repair also must be finished (‘**F**’). There is a cost field that the user enters the amount they want to pay for the repair. They must enter a valid decimal number here and that number must be less than 10000(I wanted 4 digits before the decimal because some computers from new cost more than 1000 so that maximum value is 9999.99)

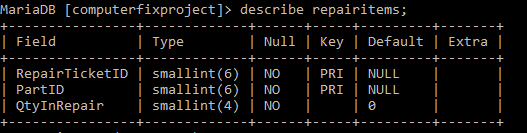
If the amount paid is less than the total cost of the repair, then a message displaying the value deducted appears and updates the total cost of the repair to reflect this value.

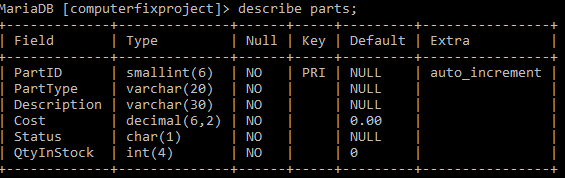
If the amount paid is greater than the total cost of the repair, then the payment will be rejected. Once the Repair has been complete paid off then the Status of the repair will be set to (‘**C**’) meaning complete and the Date that the final payment was received will be recorded in the repairs table under DatePaid column.

* **Database name:** computerfixproject (don’t ask me why I called it that 😊)
* **.SQL file name:** ComputerRepair
* **Explanation:** that’s on the first page of this document. Customer.php would be my homepage
* **Where to find Insert, Update, Delete and List:** The easiest place to see basic forms of this is in the customer page. However, there is more complex Updating and Inserting in the repairs page especially in the Add Part piece.
* **Where to find a process:** The Repairs page is a process page where a computer starts off by entering the system and ends up completely repaired.

**Here are screenshots of my table’s column data:**







I used dummy data for my tables for testing purposes, but all tables are used on my site (even though the parts table and the repair items table are not seen they work behind the scenes with the parts table keeping track of the quantity of parts in stock and the Repair items table recording what parts belong to what repair and the quantity of each)

**Here are screenshots of my tables inserted data:** 