Assignment 4 (F18) RD – PROG2110

*Please complete this assignment individually.*

**IMPORTANT NOTE:** You will need to complete this project in order to be ready for the Final Exam in this course. During the final exam, you will be able to bring your solution from Assignment 4 and extend its functionality to meet the needs of the Exam questions. Although the exam will be open book/internet, you complete it on your own. **This comprehensive exercise is a way for you to have some fun creating small, prototype of a fully functional, DB-backed application with real-world applications.**

# Objectives:

* Design and implement a relational database using MySQL
* Design and implement a front-end application that connects to and uses the DB you created
* Demonstrate your ability to create a SQL script to create and populate the database
* Demonstrate that your application front-end can properly complete the required workflow
* Have a working code-base prepared that will help you complete the final exam in this course.

# Assignment Background:

Now that you’ve learned about Relational Database design and SQL, Wally Los Gatos would like to ask you to create a prototype application that will show off your skills in these areas. Referring back to Assignment 2, you will create a portion of the Wally’s World database as a live database with a front-end application that allows us to demonstrate to Wally that we can complete essential workflow.

# Requirements:

1. **Database Setup**
   1. Please use an installation of MySQL as we’ve covered in class.
   2. The DB should be located on the same PC as your front-end application, and you MUST reference access to the DB server from your code with 127.0.0.1 as the IP address and MySQL’s default port (3306).
   3. For this exercise, you should use ‘root’ and ‘Conestoga1’ for access from your code (e.g. within the DB connection string).
   4. In your DB creation script (and within your front-end code) you must name your version of Wally’sWorld DB with your initials, like this: **J**ohn **L**ennon, the DB must be named **JL**Wally.
   5. Why do all this??? In the case where I need to review your code and re-run your app for marking purposes, I cannot loose time to having to do DB setup. If we all follow this setup, it will help immensely!
2. **Database Creation** – You must review assignment 2 and develop a SQL script that will build a portion of Wally’s World Database, enforce all required uniqueness identifiers and key constraints, and populate the database with test data. This script must be properly commented and must be executed in ‘one-shot’, with no circumvention of default MySQL controls or safeties.
   1. The focus of this effort will be on PoS – Point of Sale. Our main goals will be to demonstrate that Customers can place orders, at a dealership, for vehicles. There will be some deviations from Assignment 2 within this assignment, so please look for these items!
   2. Relations that must be included: Customer, Order, OrderLine, Dealership, Vehicle
   3. Relations you should not include: Employee, Dependent, Skill, or other not needed for this assignment. You might think about how to implement them, but do not risk problems on this assignment by making things more complex than needed.
   4. Optional relations not mentioned here: You may implement other relations (tables) in your DB design if you feel it meets the needs of normalizing the data model to 3NF, or fulfilling any M:M relationships we’ve not explicitly covered here. This isn’t expected to be needed, but so long as you do not overcomplicate this prototype scenario, you should be good to extend the model if you wish.
   5. See Appendix A – Required Sample Data for additional requirements
3. **Front-End Application** – You need to create an application that one might see on a PoS (Point of Saile) terminal/salesperson’s computer. This application needs to help us complete Sales Records (which interact with and change data in the DB) and record new orders, but also to check in on any pending orders. You should try to envision that the staff at Wally’s use this to do their job of selling vehicles to customers. Staff members will never know or need to know how to run SQL queries – this is your job in creating a functional front-end system for them ☺

**Here’s the workflow scenarios you need to hit:**

* 1. The general workflow here is to create a sales order covering the situations below. One important note is that for this assignment, we will not add any new Dealerships to the database, but we must support buying and selling Vehicles, and adding a new customer to the database when he/she comes to the checkout for their first order.
  2. Customer places a new order for one vehicle (and one trade-in if applicable) per order (Ministry of Transportation Regulations!) at a single Dealership; Order is completed (PAID), and inStock flag is adjusted.
  3. Customer places a new order for one vehicle at a dealership; Order cannot be completed yet, likely because financing must be arranged (HOLD). The inStock status of each vehicle needs to be properly supported (e.g. a vehicle can be Yes/No/Hold for inStock status).
  4. Customer wishes to cancel a HOLD order; The system should return the vehicle from Hold to Yes inStock status; Order is retained for records, indicating that a customer attempted to purchase a vehicle, but all dollar amounts are zeroed, and the inStock status of the vehicle is handled per above. Final Order state in this case should be CNCL.
  5. Staff member wishes to look up an order (E.g. to possibly move it from HOLD to PAID, or PAID to RFND, or from HOLD to CNCL).
  6. Staff member wishes to view inventory levels for all products at a selected branch.

1. **Testing in Class**
   1. This assignment includes an in-class demo portion, where some/all of your grade is determined. See below (‘What is marked in class?’)
   2. You may run the demo from a lab PC, or your own PC. Be advised that your solution must be easily compiled/workable on a Lab PC, so please avoid the use of overly exotic libraries, and such.
2. **Code Base**
   1. You may use any code-base you wish on the front-end, so long as there is a useful API library for it. Really strong suggestions here include a) Windows app (forms or similar are fine) with appropriate MySQL API, b) Go LAMP/WAMP/XAMP … Remember that MySQL and PHP have been strongly associated for EVER, and thus a web-app using PHP scripts to run the DB connections/queries might be very strong!
   2. Recommendation: It is the end of the semester. Think carefully about teaching yourself a net-new code framework versus using some other base you’ve become familiar with.

# What is handed in?

1. As a PDF, provide me a copy or screen shot of your schema FROM the MySQL workbench. This needs to represent the DB as you script will implement it, not a design model/ideal.
2. In zipped file, please include your a) SQL script and b) your front-end code/project. Ensure you followed the DB Setup notes above! This zipped file is uploaded to the D2L Assignment folder as required before the deadline.

# What is marked in class?

During a designated class period, I will hold brief demos where you will need to show me key workflow including (but not limited to): a) Run your SQL script to create/populate the DB, b) Enroll a new customer while creating a new order, c) Retrieve a past sales record, d) Display Inventory

# What is the marking breakdown for this assignment?

Schema Diagram from MySQL Workbench – 0 to 5 pts (errors in design may reduce mark)

SQL script executes in one attempt – 0 or 5 pts. (it runs, or it doesn’t)

4 Workflows demonstrated – 20 pts (5 ea. Part marks possible)

Front-end application is functional and ‘elegant’ – 0 to 10 pts (subjective per instructor).

**Total marked out of 40pts.**

NOTE: Your code (Front-End and SQL) must conform to SET Code standards. Penalties up to and including a ‘C-Cap’ (60% max grade) may be applied.

**BONUS** – The following are optional goals for the Assignment. They are not required for A4 or the Final Exam, but each will boost your mark in this course by .5% each if you complete them. These can be demonstrated to the instructor on demo day, and the

1. Allow us to search for a customer when we are completing a Sales Order by looking up their last name or phone number.
2. Allow us to easily add a new customer to the DB from the Sales Order screen.
3. Ensure that the ordering system allows a dealership to purchase

# Appendix A: Sample Data and Workflow Notes

**Customer Table:**

* Attributes you need to track are: ID, first name, last name, phone number

**Dealership Table:**

* Attributes you need to track are: ID, dealership name

**Vehicle Table:**

* Attributes you need to track are: VIN, Year, Make, Model, Colour, Kms, wPrice, inStock
* wPrice here refers to the wholesale price of the vehicle

**Order and Order Line tables:**

* Attributes you need include: ID, Date, Vehicle, sPrice (price of the final sale), Status (e.g. PAID, CNCL, RFND, HOLD)
* I won’t tell you all you need for this, or which table the attributes above should be in which of these two tables.
* You should not carry derived attributes in your table… your Front-End UI should calculate and display tax amount or pre-post tax totals (etc.) for the order.
* Speaking of calculating amounts – the sPrice should be equal to (wPrice \* 1.4). This means that there’s a 40 percent markup on all vehicles sold to a customer. This markup is NEVER displayed on the sales record!
* If we accept a vehicle on trade in or purchase it through other means (e.g. auction), the markup above is not applied when we bring it into inventory. It must be properly applied when the car is sold, however.

**What is required on a Sales Record?**

1. Header message thanking the customer ( by firstName, lastName) for shopping at Wally’s <DealershipName> location on <date>. See the example that has been provided.
2. Order Number should be clearly displayed, along with PAID, HOLD, CANCEL, or REFUND state.
3. Products are summarized through an OrderLine. Look at nearly any receipt or sales invoice you can find to see this kind of pattern. We need to precisely record a) The name of the product on the line, b) the quantity ordered, c) the unit price of a single item, d) the extended price (Quantity\*UnitPrice) for each item ordered.
4. Summary – At the bottom of the invoice/Sales Record, we should see a summary price for all the items above, the amount of tax (e.g. HST 13%) for the order, then a total (Summary + Tax).

**Completing Orders and Managing Inventory**

What does it mean to complete an order? Aside from marking the order PAID, we would need to adjust the value of inStock accordingly. If an order were marked HOLD, it means the inventory is not yet adjusted. A refunded order (RFND) means that we need to a) zero out the sale total for the Order in question and b) adjust the inventory to match what just came back. This allows us to complete sales, or to put them on hold, possibly reverting the sale and putting a vehicle back into available stock (e.g. customer’s financing did not go through, or they changed their mind).

# Starting Sample Data

Ensure that your DB script sets up the following sample data. Please note that you are free to make minor corrections as needed to fit the data below into a) data types you’ve chosen for the DB, and b) to support require scenarios (e.g. will you record prices paid for trade in/purchases as a negative here? This is an important decision for Wally’s accountant!).

Customers:

Wally’s World of Wheels Inc., 519-555-0000

Ringo Starr, 416-555-1111

Mick Jagger, 519-555-2222

Eric Clapton, 519-555-3333

Dealerships:

Sports World

Guelph Auto Mall

Waterloo

Vehicles:

58847722BRB, 2010, Honda, Civic, Blue, 120332, 6500, Yes

26663747GTG, 2009, Ford, Focus, Black, 89221, 8950, Yes

99277544LOL, 2012, Volkswagen, Jetta, Silver, 156233, 13450, Yes

27764534RTB, 2013, Dodge, Ram, Red, 211023, 10900, Yes

Previous orders:

On Sept 20th, 2017, Eric Clapton purchased and paid for the Dodge Ram at Sports World.

On Sept 22nd, 2017, Wally’s World of Wheels Inc. purchased a new vehicle to be sold at Waterloo (53347223WTF, 2011, Buick, Regal, Mint, 134538, 7950, Yes). Note: HST is paid when Wally’s buys a vehicle for our purposes. Additionally, when Wally’s buys a vehicle, it is added to the DB, and the money for the transaction should be recorded as negative e.g. -7950.00).

On October 6th, 2017, Mick Jagger placed a HOLD order on the Honda Civic at the Guelph Auto Mall.

On October 20th, 2017, Mick Jagger cancelled his previous, pending order.

On November 2nd, Ringo Starr traded in an older car (99146514OMG, 2008, Volkswagen, Jetta, White, 199012, 2500, Yes) before purchasing the Silver Jetta from stock, at Waterloo.

# Sample Sales Record:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Thank you for choosing Wally’s World of Wheels at

Guelph Auto Mall for your quality used vehicle!

**Date: On Sept. 20, 2017**

**Customer: Eric Clapton**

**Order ID: 5001 - PAID**

2013 Dodge Ram, Red

VIN: 27764534RTB KMS: 211023

Purchase Price: $15,260.00

Trade In: $ 0.00

**Subtotal = $ 15,260.00**

**HST (13%) = $ 1,983.80**

**Sale Total = $ 17,243.80**

**Fall 2017 – Demo Marking Guide Grade: /40 (Bonuses: 1.5)**

**Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(Print Clearly)**

**Preparation for Demo**

1. Have a printed copy of this form, and please print your name above
2. Please print a hard copy of your Schema diagram for the assignment and show it to me for part of this assessment
3. Ensure that we will be running your DB Creation script against a ‘clean’ DB Server Instance

**Part 1 – Schema Diagram 0 to 5 pts**

* Diagram displays good design choices, constraints, and normalization

**Part 2 – DB Script Runs 0 or 5 pts**

* Must run against clean DB
* Must create DB, Tables, PKeys, FKeys and populate with sample data

**Part 3 – Front End UI Assessment 0 to 10 pts**

* Subjective evaluation based on ease of use; clean design; etc.

**Part 4 – Workflow Tests 0 to 20 pts (5 ea.)**

* Instructor asks to see the following demonstrated.
  1. Use the Wally’s World customer to buy the following car (e.g. an Order is created modify inventory) 9876543NON, 2012, Toyota, Prius, Silver, 222333, 15999, Yes
  2. New Customer (Ed Barsalou, 519-555-4444) buys any car in inventory.
  3. Existing Customer buys any car from inventory and has the following trade in 4721139OMG, 1976, Chevrolet, Monte Carlo, Beige, 567888, 1200, Yes
  4. Orders need to be confirmed as per the sample invoice/sale receipt. Looking for:
* Order number, Date, Customer Name, Vehicle Details, Totals/Taxes

**Part 5 – Bonuses**

Check to see if any of these bonuses were obtained (+0.5% to course grade ea.)

1. Allow us to search for a customer when we are completing a Sales Order by looking up their last name or phone number
2. Allow us to easily add a new customer to the DB from the Sales Order screen
3. Ensure that the ordering system allows a dealership to purchase a vehicle