A close up of a sign

Description automatically generated

**Lebanese American University**

**School of Business**

ITM302

Inventory Management System

Submitted by:

* Hisham Younes
* Husam Aldaoud
* Yousef Atrach

*Under the guidance of*

*Dr. Abbas Tarhini*

*Table of Contents*

Contents

[Company Profile: *Maliks* 3](#_Toc90302528)

[Company Overview & Mission 3](#_Toc90302529)

[Inventory Management System 4](#_Toc90302530)

[Purpose 4](#_Toc90302531)

[System Overview 4](#_Toc90302532)

[Interview Reports 6](#_Toc90302533)

[Business Scenario 8](#_Toc90302534)

[Business Rules 10](#_Toc90302535)

[Entity Description 11](#_Toc90302536)

[EERD 12](#_Toc90302537)

[Schema Diagram 13](#_Toc90302538)

[Mapping 14](#_Toc90302539)

[Relations Description 15](#_Toc90302540)

[MYSQL Table Creation 16](#_Toc90302541)

[Tables Description 18](#_Toc90302542)

[MYSQL Value Insertion 26](#_Toc90302543)

[MYSQL Queries 31](#_Toc90302544)

# 

# Company Profile: *Maliks*

## Company Overview & Mission

Maliks company has started 34 years ago when Malik Barakat, the chief executive officer, decided to start his own bookstore in Lebanon. According to Maliks’ company website, it is stated that they have numerous core beliefs and values. These core values are both internal and external to the company. For instance, under the internal umbrella, they believe in honesty, trust and team work. Whereas, according to the external prescriptive, they believe in their people, which are their customers and partners.

Four sales divisions form the foundation of this company. The first division is the stationery and office supply section, where Maliks owns three stationery sections. The wholesale section, functioning in Jnah area. The Business-2-Business section holding over 1000 corporate accounts. In addition, the retail section, which is made of over 30 branches distributed over Lebanon. The second division that Maliks has is the business and copy center. It is worth mentioning that Maliks has become the largest copy center in Lebanon. Maliks offers a full line of copy as well as computer services from a small photocopy to a large poster plotting. Maliks further has other services such as: classified advertising sworn translation etc. Maliks’ third division is the multimedia. For instance, Maliks provides accessories for computers as well as electronic gadgets. Finally, Maliks’s fourth division is the books section. It is worth noting that Maliks is the number one general as well as academic bookshop in Lebanon. They stock over a hundred-thousand titles. All their books are available on their e-commerce website that gets updated periodically.

We would like to shed the light on the fact that Maliks’ stores do also sell products coming from over 30 worldwide brands. Some of these brands are: hp, kinary, Elsoon and many more. During the years and through Maliks’ journey, this company was able to strategically position itself in the Lebanese market. Maliks’ serves an excellent service with high quality products which made them gain their customers’ satisfaction as well as loyalty. One of the systems that is being used inside Maliks’ company is the inventory management system. This system will be tackled in this project.

# Inventory Management System

## Purpose

The main purpose of inventory management is to make sure that there is enough goods or materials in the company to meet the demand while avoiding having excess inventory or making overstock.

## System Overview

The business process inside Maliks’ company is mainly focused on two aspects. First, the selling aspect, which is characterized by selling school supplies, books, customized or normal papers, computer accessories as well as electronic gadgets. On one hand, the transactions can take place physically, where customers visit any Maliks’ store branch to buy the products they want. On the other hand, customers can go to Maliks’ official website, where they can view the products and place online orders. Their orders will be tracked and delivered under few working days. Second, the purchasing aspect, which means buying raw materials from suppliers to compensate the lack of any product in the company’s warehouse. This is a crucial aspect for the company by which it keeps them both aware of what they have in their inventory and ready to meet the market’s demand.

Inventory management plays a significant role in identifying the type and quantity of stock to be ordered at the suitable time. It keeps track of inventory from purchase to the sale of goods. It recognizes and responds to trends to make sure that the available amount of goods is enough to meet with customers’ orders. This system also warns the company in case of having a shortage in a specific type of products.

Also, Maliks has more than 15 branches all over Lebanon. Thus, the need of a sufficient inventory system is crucial to enable the visibility between the branches.

In this project, we seek to keep records of the inventory of Maliks within the branches and those in the warehouse.

# Interview Reports

|  |  |
| --- | --- |
| **Participant** | Bilal Zeenni  Bilal.zeenni@hotmail.com |
| **Position** | Assistant Manager |
| **Date** | 25/10/2021 |
| **Duration** | Around 21 minutes |

|  |  |  |
| --- | --- | --- |
| Time (min) | Question | Answer |
| 2 | What are the general business processes of a library? | Borrowing is unallowed. We mainly do requests to purchase. |
| 1 | Is the POS system related to the inventory? | Point of sales (POS) is not related to the inventory |
| 2 | Do you have to manually add/ remove a book that is sold or purchased from the system? | Processes such as adding and removal of books after being sold are processed automatically by the system. |
| 1 | Please give me an overview of the current inventory system, if any. | Every quarter, manually counted. |
| 1 | Is there a system for borrowed books? | We do not borrow nor lend books |
| 2 | Have you ever experienced excess or shortage in the inventory? | We experienced shortage not excess. Manual request is better due to the market and we can’t return bought books. |
|  | If any, how was it faced? | Manual ordering |
| 1 | Is there a single inventory system for each branch? | Yes. Each branch has its own inventory system. |
| 1 | Can you access the inventory information of other branches? | Yes, inventory systems of each branch are linked together |
| 1 | Is there a single warehouse for the whole company? | The whole company of Maliks has two warehouses |
| 2 | Does the system send automatic purchase requests in case of shortage? | In case of shortage, purchase requests are done manually to avoid ordering excess books. |
| 2 | Is there an automatic ID generated for each book? | Each book is saved with its unique ID number |
| 1 | If yes, does is the ID generated linked to the branch? | ID numbers are linked to the branch |
| 2 | What are the main points to consider when creating an inventory management system? | Visibility, request to purchase new stuff, pricing, minimum and maximum budget, viewing and assessing the overall performance quarterly. |
| 1 | Do you do maintenance for the system periodically | Yes, it is updated accordingly whenever it starts works abnormally. |
| 1 | Do you use the same inventory management system for all your products “books, electronic gadgets etc.?” | No, Each system is used for a category. |
|  |  |  |

# Business Scenario

To achieve the company’s objectives by meeting the huge demand as well as providing consumers with significant service, Maliks high managers use an outstanding database system to create a link among the company’s numerous processes. This system is being used in all the company’s branches. According to the assistant manager, this system plays a remarkable role in tracking as well as following up to the daily sales, internal and external cash flows. Since Maliks had to cope with numerous transactions with its customers and suppliers, therefore, it was extremely important to own an inventory management system which could keep track of the company’s inventory.

Customers may buy one or more items including books and other accessories. Customers can’t lend any books. The point-of-sale is not linked to the inventory system. Books should be manually counted to find the available books and those which need to be ordered. This process is done quarterly. Books are identified through the title, author name, and year of publish. They are classified based on the subject and academic level. There is no ID number linked to each book in the current system, which will be done in the new system. The ID number will be unique for each book to be able to track it in case of an error.

Branch managers must order the books manually. We will try to create an automatic request system in case of a shortage in the inventory. Before the warehouse runs out of books of specific subject, the inventory management system sends a notification for the branch to place orders for the specified book type. The system will be able to view the inventory of other branches and order from the most suitable one. Constraints will be added. The order can’t be placed if the branch has less than 3 stocks for example. It should also find the closest branch.

Branch managers may access the inventory of other branches and must request for items in case of shortage. Items may be requested from other branches or from one of the two warehouses of the company. Received orders should be manually added to the system.

In case of deficit in the warehouses, manager requests books from the supplier. In case branch is empty, manager requests books from warehouses to the branch.

Implementing this system will automate the work, enhance the visibility, and reduce the errors caused by manual work. All departments exploited this system, where it is going to make tracking process of the inventory easier than it has ever been.

# Business Rules

Each book must have an author

Each author must have at least one book

Each book must have a publisher

Each publisher must have at least one book

Customer must place at least one order

Each order must be ordered by customer

Customer may be student or notStudent

Each order must have at least one book

Book may be placed in an order

Manager must request books from supplier

Manager must manage one branch

Each branch must be managed by one manager

Supplier must supply books to the warehouse

Manager must request books from the warehouse to the branch

Branch must have at least one warehouse

Warehouse must have code, address, and phone number

Branch must have code, name, address, and phone number

Managers must have email, phone number, and name

Order must have order ID and amount

Customer must have customer ID, customer name, phone number, address, and email

Each book must have an ISBN, title, and price

Each author must have name, address, and URL

Publisher must have name, address, URL, and phone number

Each supplier must have name, email, and phone number

# Entity Description

1. **Book:** Maliks company sells various types of books. Each book has international standard book number (ISBN), title, price, author URL and publisher URL.
2. **Branch:** Maliks company has many branches around Lebanon. Each branch has a code, name, phone number, city and street location in Lebanon.
3. **Warehouse:** is the place where stocks are saved in case the branch gets out of specific products. Each branch has at least 1 warehouse. Each warehouse has a code, phone number, city, street locations in Lebanon.
4. **Manager:** Is the one responsible for ordering books from suppliers to the warehouse. Also, he is responsible for ordering books from the warehouse to be delivered to the branch that has shortage. A manager has an email, First name, Last Name, phone number.
5. **Customer:** people who are buying books from Maliks’ branches. A customer has a customer ID, type, First Name, Last Name, phone number, email, city and street locations.
6. **Order:** is the customer’s desired thing to buy. An order has orderID, order amount, and customer ID.
7. **Author:** the person who wrote the book. The author has a URL, first name, last name, city and street locations.
8. **Publisher:** the one who published the book. A publisher has a URL, First Name, Last Name, phone number, city and street locations.
9. **Supplier:** the one that Maliks buys books from when facing a shortage in the warehouse. A supplier has an email, First Name, Last Name, and phone number.

# EERD

Graphical user interface, application, Teams

Description automatically generated

# Schema Diagram

# Graphical user interface Description automatically generated with low confidenceMapping

Street

City

Phone

Street

City

Type

Street

City

L name

F namee

L name

F namee

L name

F namee

L name

F namee

L name

F namee

# Relations Description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BookSupplied** |  |  |  |  |
| ManagerEmail | SupplierEmail | ISBN | Quantity | Date |

In case of deficit in the warehouse items, manager requests books from the supplier. this process is done to make sure that warehouse has always items to meet the demand of the branch.

ManagerEmail is linked to ManagerEmail in manager table

SupplierEmail is linked to SupplierEmail in supplier table

ISBN refers to the book

Quantity and Date are assigned as associative entities between the tables Manager, Supplier and Book.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **BooksSuppliedTo** |  |  |  |  |  |
| ISBN | ManagerEmail | BranchCode | WarehouseCode | Quantity | Date |

In case the branch is empty, manager requests books from warehouses to the branch. this process is done in order to meet with the customers demand, so that the branch doesn’t fall into shortage.

ISBN refers to the book

ManagerEmail is linked to ManagerEmail in manager table

BranchCode is linked to the BranchCode in Branch table

WarehouseCode is linked to the WarehouseCode in the Warehouse table.

Quantity and Date are assigned as associative entities between the tables Book, Manager, Branch and Warehouse.

By referring to the Enhanced Entity Relationship Diagram (EERD), we can see that a customer must be either Student or NonStudent, Student has the ID and his/her school/university, while the NonStudent has the ID and his/her occupation.

# MYSQL Table Creation

create Table Author(

Author\_URL varchar(40) primary key Not Null,

author\_First\_name varchar(40),

author\_Last\_name varchar(40),

Author\_city varchar (40),

Author\_street varchar (40));

create Table Publisher (

Publisher\_URL varchar(40) primary key Not Null,

Publisher\_First\_name varchar(40),

Publisher\_Last\_name varchar(40),

Publisher\_city varchar(40),

Publisher\_street varchar(40),

Publisher\_phoner\_number varchar(40));

create Table Book (

ISBN Integer primary key Not Null,

Title varchar (20),

price Integer,

Author\_URL varchar(40),

Publisher\_URL varchar(40),

Foreign key (Publisher\_URL) References Publisher on update cascade,

Foreign key (Author\_URL) References Author on update cascade);

create Table Warehouse (

Warehouse\_code Integer Primary Key Not Null,

Warehouse\_city varchar(40),

Warehouse\_street varchar(40),

Warehouse\_phone\_number varchar (40));

create Table Branch (

Branch\_code integer primary key Not Null,

Branch\_name varchar(40),

Branch\_phone\_number varchar(40),

Branch\_city varchar(40),

Branch\_street varchar(40));

create Table Supplier (

Supplier\_email varchar(40) primary key Not Null,

Supplier\_First\_name varchar(40),

Supplier\_Last\_name varchar(40),

Supplier\_phone\_number varchar(40));

create Table Manager (

Manager\_Email varchar(40) primary key Not Null,

Manager\_First\_name varchar(40),

Manager\_Last\_name varchar(40),

Manager\_phone\_number varchar(40));

create Table Customer(

Customer\_ID Integer primary key Not Null,

Customer\_First\_Name varchar(40),

Customer\_Last\_Name varchar(40),

Customer\_phone\_number varchar(40),

Customer\_email varchar(40) Not Null,

Customer\_city varchar(40),

Customer\_street varchar(40),

Customer\_Type varchar(40));

create Table Orders (

Orders\_ID Integer primary key Not Null,

Orders\_Amount Integer,

Customer\_ID Integer,

Foreign key (Customer\_ID) References Customer on update cascade);

create Table Student (

Student\_ID Integer primary key Not Null,

Student\_School varchar(40));

create Table NonStudent (

nonStudent\_ID Integer primary key Not Null,

occupation varchar(40));

create Table SuppliedBy (

Supplier\_Email varchar(40) Not Null,

Warehouse\_code Integer Not Null,

ISBN Integer Not Null,

Quantity Integer,

Foreign key (Supplier\_Email) References Supplier on update cascade,

Foreign key (Warehouse\_code) References Warehouse on update cascade,

Foreign key (ISBN) References Book on update cascade,

primary key (Supplier\_Email,Warehouse\_code,ISBN));

create Table BooksSuppliedTo (

ISBN Integer Not Null ,

Manager\_Email varchar(40) Not Null,

Branch\_code Integer Not Null,

Warehouse\_code integer,

Dates Date,

Quantity Integer,

Foreign key (Warehouse\_code) References Warehouse on update cascade,

Foreign key (Branch\_code) References Branch on update cascade,

Foreign key (Manager\_Email) References Manager on update cascade,

Foreign key (ISBN) References Book on update cascade,

primary key (ISBN,Manager\_Email,Branch\_code, Warehouse\_code));

create Table BooksSupplied (

Manager\_Email varchar(40) Not Null,

Supplier\_email varchar(40) Not Null,

ISBN Integer Not Null,

Quantity Integer,

Dates Date,

Foreign key (Manager\_Email) References Manager on update cascade,

Foreign key (Supplier\_email) References Supplier on update cascade,

Foreign key (ISBN) References Book on update cascade,

primary key ( Manager\_Email, Supplier\_email, ISBN));

create Table BooksOrdered (

ISBN Integer Not Null,

Orders\_ID Integer Not Null,

Foreign key (Orders\_ID) References Orders on update cascade,

Foreign key (ISBN) References Book on update cascade,

primary key ( ISBN , Orders\_ID));

# Tables Description

**Author**

Graphical user interface, text, application, email

Description automatically generated

**Publisher**

Graphical user interface, application

Description automatically generated

**Book**

Graphical user interface, text, application, email

Description automatically generated

**Branch**

Graphical user interface, text, application, email

Description automatically generated

**Supplier**

Graphical user interface, text, application, email

Description automatically generated

**Manager**

Graphical user interface, text, application, email

Description automatically generated

**Customer**

Graphical user interface, text, application

Description automatically generated

**Orders**

Graphical user interface, application

Description automatically generated

**Warehouse**

Graphical user interface, text, application, email

Description automatically generated

**Student**

Graphical user interface, text, application, email

Description automatically generated

**NonStudent**

Graphical user interface, text, application, email

Description automatically generated

**BooksOrdered**

Graphical user interface, text, application, email

Description automatically generated

**BooksSupplied**

Graphical user interface, text, application, email

Description automatically generated

**BooksSuppliedTo**

Graphical user interface, application, email

Description automatically generated

**SuppliedBy**

Graphical user interface, text, application, email

Description automatically generated

# MYSQL Value Insertion

INSERT INTO Customer

VALUES ('11103', 'darren', 'powers', '03-987827', 'darren.powers1@gmail.com', 'tripoli', 'azmi', 'S');

INSERT INTO Customer

VALUES ('11189', 'anet', 'molinare', '03-987679', 'anet.molinare1@gmail.com', 'beirut', 'hamra', 'S');

INSERT INTO Customer

VALUES ('11123', 'ted', 'butterfield', '03-127623', 'ted.butterfield1@gmail.com', 'beirut', 'hamra', 'S');

INSERT INTO Customer

VALUES ('11245', 'kunst', 'nuklert', '71-982737', 'kunst.nuklert2@gmail.com', 'byblos', 'the broad', 'N');

INSERT INTO Customer

VALUES ('11976', 'paul', 'stevenson', '70-982737', 'paul.stevenson1@gmail.com', 'byblos', 'the broad', 'N');

INSERT INTO Customer

VALUES ('18657', 'joel', 'eaton', '81-876278', 'joel.eaton1@gmail.com', 'jounieh', 'al-mataaem', 'S');

INSERT INTO Customer

VALUES ('12123', 'ken', 'brennan', '71-003873', 'ken.brennan01@gmail.com', 'saida', 'al-kalea', 'N');

INSERT INTO Customer

VALUES ('11223', 'stewart', 'carlenon', '71-098782', 'stewart.carlenon1@gmail.com', 'saida', 'al-qadeem', 'S');

INSERT INTO Customer

VALUES ('11267', 'duane', 'coonaren', '70-981278', 'duane.coonaren1@gmail.com', 'byblos', 'al-ghuraba', 'S');

INSERT INTO Customer

VALUES ('11769', 'christopher', 'schild', '03-987980', 'christopher.schild1@gmail.com', 'beirut', 'al-masaref', 'N');

INSERT INTO BooksSuppliedTo

VALUES ('108', 'paul.khadra11@gmail.com', '1102', '12002', '2021-10-2', '2');

INSERT INTO BooksSuppliedTo

VALUES ('109', 'tamara.hamad02@gmail.com', '1103', '13001', '2021-10-3', '2');

INSERT INTO Publisher

VALUES ('emilyburns.com', 'emily', 'burns', 'beirut', 'hamra', '71723652');

INSERT INTO Publisher

VALUES ('erichoffman.com', 'eric', 'hoffman', 'beirut', 'hamra', '71820983');

INSERT INTO Publisher

VALUES ('genehale.com', 'gene', 'hale', 'beirut', 'al-masaref', '70986553');

INSERT INTO Publisher

VALUES ('stevekaram.com', 'steve', 'karam', 'tripoli', 'azmi', '70816513');

INSERT INTO Publisher

VALUES ('lindacazamias.com', 'linda', 'cazamias', 'mount lebanon', 'suzk', '81872732');

INSERT INTO Publisher

VALUES ('rubenasuman.com', 'ruben', 'ausman', 'saida', 'al-omara', '81982369');

INSERT INTO Publisher

VALUES ('erinsmith.com', 'erin', 'smith', 'saida', 'al-souk', '70927267');

INSERT INTO Publisher

VALUES ('odellanelson.com', 'odella', 'nelson', 'sour', 'al-athar', '71873789');

INSERT INTO Publisher

VALUES ('patrickgabe.com', 'patrick', 'gabe', 'sour', 'al-athar', '70876612');

INSERT INTO Publisher

VALUES ('lenahernandiz.com', 'lena', 'hernandiz', 'byblos', 'the broad', '03908876');

INSERT INTO Author

VALUES ('andrewallen.com', 'andrew', 'allen', 'tripoli', 'azmi');

INSERT INTO Author

VALUES ('irenemaddox.com', 'irene', 'maddox', 'beirut', 'hamra');

INSERT INTO Author

VALUES ('haroldpawlan.com', 'harold', 'pawlan', 'mount lebanon', 'suzk');

INSERT INTO Author

VALUES ('petekriz.com', 'pete', 'kriz', 'saida', 'azrak');

INSERT INTO Author

VALUES ('alejandrogrove.com', 'alejandro', 'grove', 'byblos', 'the broad');

INSERT INTO Author

VALUES ('zuschussdan.com', 'zuschuss', 'dan', 'byblos', 'al-kalea');

INSERT INTO Author

VALUES ('kenblack.com', 'ken', 'black', 'tripoli', 'two hundred');

INSERT INTO Author

VALUES ('sandraflanagan.com', 'sandra', 'flanagan', 'beirut', 'hamra');

INSERT INTO Author

VALUES ('tracyblumstein.com', 'tracy', 'blumstein', 'beirut', 'ahdab');

INSERT INTO Book

VALUES ('101', 'ENG009', '20', 'andrewallen.com', 'emilyburns.com');

INSERT INTO Book

VALUES ('102', 'ENG101', '25', 'irenemaddox.com', 'erichoffman.com');

INSERT INTO Book

VALUES ('103', 'ENG102', '25', 'haroldpawlan.com', 'genehale.com');

INSERT INTO Book

VALUES ('104', 'ENG202', '25', 'petekriz.com', 'stevekaram.com');

INSERT INTO Book

VALUES ('105', 'ENG203', '30', 'alejandrogrove.com', 'lindacazamias.com');

INSERT INTO Book

VALUES ('106', 'MTH207', '44', 'zuschussdan.com', 'rubenasuman.com');

INSERT INTO Book

VALUES ('107', 'PHYS211', '49', 'kenblack.com', 'erinsmith.com');

INSERT INTO Book

VALUES ('108', 'ITM211', '33', 'sandraflanagan.com', 'odellanelson.com');

INSERT INTO Book

VALUES ('109', 'ITM311', '33', 'tracyblumstein.com', 'patrickgabe.com');

INSERT INTO Orders

VALUES ('152345', '2', '11245');

INSERT INTO Orders

VALUES ('164825', '1', '11976');

INSERT INTO Orders

VALUES ('184637', '1', '18657');

INSERT INTO Orders

VALUES ('193727', '2', '12123');

INSERT INTO Orders

VALUES ('176453', '5', '11223');

INSERT INTO Orders

VALUES ('104726', '4', '11267');

INSERT INTO Orders

VALUES ('184638', '3', '11189');

INSERT INTO Orders

VALUES ('194736', '1', '11123');

INSERT INTO BooksOrdered

VALUES ('105', '164825');

INSERT INTO BooksOrdered

VALUES ('106', '184637');

INSERT INTO BooksOrdered

VALUES ('107', '193727');

INSERT INTO BooksOrdered

VALUES ('108', '176453');

INSERT INTO Manager

VALUES ('ahmad.khalil01@gmail.com', 'ahmad', 'khalil', '81-826432');

INSERT INTO Manager

VALUES ('paul.khadra11@gmail.com', 'paul', 'khadra', '70-718463');

INSERT INTO Manager

VALUES ('tamara.hamad02@gmail.com', 'tamara', 'hamad', '03-726264');

INSERT INTO Manager

VALUES ('mohammad.ali02@gmail.com', 'Mohammad', 'Ali', '03-726254');

INSERT INTO Supplier

VALUES ('claire.gute22@gmail.com', 'claire', 'gute', '03-989898');

INSERT INTO Supplier

VALUES ('darrin.huff12@gmail.com', 'darrin', 'huff', '03-676554');

INSERT INTO Supplier

VALUES ('sean.donnell@gmail.com', 'sean', 'donnell', '03-653425');

INSERT INTO Supplier

VALUES ('brosina.hoffman34@gmail.com', 'brosina', 'hoffman', '03-872570');

INSERT INTO Branch

VALUES ('1101', 'Beirut1101', '71-777771', 'beirut', 'hamra');

INSERT INTO Branch

VALUES ('1102', 'Byblos1102', '71-777772', 'byblos', 'the broad');

INSERT INTO Branch

VALUES ('1103', 'Tripoli1103', '71-777773', 'tripoli', 'azmi');

INSERT INTO Warehouse

VALUES ('11001', 'beirut', 'hamra', '03-875432');

INSERT INTO Warehouse

VALUES ('11002', 'beirut', 'al-masaref', '03-747364');

INSERT INTO Warehouse

VALUES ('12001', 'byblos', 'the broad', '03-112233');

INSERT INTO Warehouse

VALUES ('12002', 'byblos', 'the broad', '03-111222');

INSERT INTO Warehouse

VALUES ('13001', 'tripoli', 'azmi', '03-456789');

INSERT INTO Warehouse

VALUES ('13002', 'tripoli', 'two hundred', '03-101010');

INSERT INTO BooksSupplied

VALUES ('ahmad.khalil01@gmail.com','darrin.huff12@gmail.com', '103', '2', '2021-10-10');

INSERT INTO BooksSupplied

VALUES ('paul.khadra11@gmail.com','sean.donnell@gmail.com', '104', '2', '2021-10-11');

INSERT INTO BooksSupplied

VALUES ('tamara.hamad02@gmail.com', 'brosina.hoffman34@gmail.com', '105', '4', '2021-10-11');

INSERT INTO SuppliedBy

VALUES ('claire.gute22@gmail.com', '11002', '102', '3');

INSERT INTO SuppliedBy

VALUES ('darrin.huff12@gmail.com', '12001', '103', '4');

INSERT INTO SuppliedBy

VALUES ('sean.donnell@gmail.com', '12002', '104', '2');

INSERT INTO Student

VALUES ('11103', 'LAU');

INSERT INTO Student

VALUES ('11189', 'LAU');

INSERT INTO Student

VALUES ('11123', 'AUB');

INSERT INTO Student

VALUES ('18657', 'BAU');

INSERT INTO Student

VALUES ('11223', 'UOB');

INSERT INTO Student

VALUES ('11267', 'LAU');

INSERT INTO NonStudent

VALUES ('11245', 'barber');

INSERT INTO NonStudent

VALUES ('11976', 'unemployed');

INSERT INTO NonStudent

VALUES ('12123', 'cheff');

INSERT INTO NonStudent

VALUES ('11769', 'unemployed');

# MYSQL Queries

1. Show the name of the customer who placed an order

SELECT Orders.Orders\_ID, Customer.Customer\_First\_Name, Customer.Customer\_Last\_Name

FROM Customer, Orders

WHERE Customer.customer\_ID = Orders.customer\_ID;

Table

Description automatically generated

1. Show the quantity of books that the manager ordered from the supplier

SELECT Manager.Manager\_First\_name, Manager.Manager\_last\_name, Supplier\_email, Quantity

FROM Manager, BooksSupplied

WHERE Manager.Manager\_Email = BooksSupplied.manager\_email

Table

Description automatically generated

1. Show the total number of supplied books done on 11.10.2021

SELECT dates, SUM(Quantity) AS SumOfQuantity

FROM BooksSupplied

WHERE dates ='20211011'

GROUP BY BooksSupplied.Dates;



1. Show the customer name that ordered books with more than 50$

SELECT Orders.orders\_ID, Customer.Customer\_First\_name, Customer.Customer\_Last\_name, [price]\*[orders\_amount] AS TotalPrice

FROM Customer, Orders, BooksOrdered, Book

WHERE Customer.customer\_ID = Orders.customer\_ID AND Book.ISBN = BooksOrdered.ISBN AND Orders.orders\_ID = BooksOrdered.orders\_ID

GROUP BY Orders.orders\_ID, Customer.Customer\_First\_name, Customer.Customer\_Last\_name, Orders.orders\_amount, Book.price, [price]\*[orders\_amount]

HAVING ((([price]\*[Orders\_Amount])>50));

Table

Description automatically generated

1. Show the address of the customer who ordered the book 107

SELECT Customer.Customer\_city, Customer.Customer\_street

FROM Customer, Orders, BooksOrdered

WHERE Customer.customer\_ID = Orders.customer\_ID AND Orders.orders\_ID = BooksOrdered.Orders\_ID AND (((BooksOrdered.ISBN)=107));

Shape

Description automatically generated with medium confidence

1. Show all the orders ordered by each student

SELECT Orders.orders\_ID, BooksOrdered.ISBN, Student.\*

FROM Student, Customer, Orders, BooksOrdered

WHERE Student.Student\_ID = Customer.customer\_ID AND Customer.Customer\_ID = Orders.Customer\_ID AND Orders.Orders\_ID = BooksOrdered.orders\_ID;

Table

Description automatically generated

1. Show orders placed by customer where the order amount is greater than 2

SELECT Orders.orders\_ID, BooksOrdered.ISBN, Customer.Customer\_First\_Name, BooksOrdered.Orders\_ID, Orders.Orders\_Amount amount

FROM Customer, Orders, BooksOrdered

WHERE Orders.Orders\_ID = BooksOrdered.orders\_ID AND Customer.Customer\_ID = Orders.customer\_ID AND (((Orders.[orders\_amount])>2));

Graphical user interface, text, application, email

Description automatically generated

1. Show All the books where the author city is Tripoli

SELECT Book.ISBN, Book.Title, Book.author\_URL, Author.author\_First\_name, Author.author\_Last\_name

FROM Author, Book

WHERE Author.Author\_URL = Book.Author\_URL AND (((Author.Author\_city)= 'tripoli'));

Graphical user interface, application

Description automatically generated

1. Show the name of the manager that requests more than 2 books from the supplier

SELECT Manager.Manager\_First\_name, Manager.Manager\_Last\_name

FROM Manager, BooksSupplied, BooksSuppliedTo

WHERE BooksSupplied.manager\_email = BooksSuppliedTo.manager\_email AND Manager.manager\_email = BooksSupplied.Manager\_Email AND (((BooksSupplied.quantity)>2));

A picture containing table

Description automatically generated

1. Show the quantity of book ordered by branch and to which address

SELECT BooksSuppliedTo.ISBN, Book.Title, Book.price, BooksSuppliedTo.quantity, Branch.Branch\_name, Branch.Branch\_city, Branch.Branch\_street

FROM Book, Branch, BooksSuppliedTo

WHERE Branch.Branch\_code = BooksSuppliedTo.Branch\_code AND Book.ISBN = BooksSuppliedTo.ISBN;

Table

Description automatically generated