DATA BASE MANAGEMENT SYSTEM

project



**Submitted to:**  **Submitted by:**

**Dr. Brij Bihari Dubey** Honey Bhardwaj

Professor (Data base management system) 1800221C203

School of Engineering and Technology

BML Munjal University

**INTRODUCTION:**

What is inventory control?

Inventory control regulates the inventory that is already in your warehouse. This involves knowing what is in stock inside and out – how much is available, where is it located in the warehouse and in what condition it’s in. It is also about ensuring your warehouse is set up in a way that allows warehouse staff to quickly pick and pack to speed up customer order fulfilment. In controlling the inventory, you have on hand, you’ll also be aiming to keep inventory costs down. This can involve identifying the least popular items and reducing the stock, accurately forecasting changes in demand to avoid overstocking. For food and beverage manufacturers, this involves minimizing waste by using inventory before they expire.

What is inventory management?

On the other hand, inventory management involves forecasting and product replenishment. Inventory management determines when to order products, in what quantities and from which supplier. This ensures that your business will always have the right quantity of the right item in the right location at the right time. The scope of inventory management is arguably wider than inventory control. While inventory control only requires understanding of your warehouse, inventory management requires you to understand the supply chain and maintain good relationships with your suppliers. In managing your inventory, you’re aiming to to get inventory at the right place at the right time. This involves quickly reordering stock, having resources in the right place and having efficient process in place to receive and store inventory stock.

**OBJECTIVES:**

Some of the main objective of the project are as follows:

* Detection of out of out of stock products
* Easily adding up of products by admin
* Efficient way of calculating of purchases by user
* Automatic control in ordering out of stock product
* Well maintained login and registering based system

**DESCRIPTION OF DATABASE SCHEMA:**

There are total of 4 tables

* Admin Table
* Users Table
* Items Table
* Bill Table

Each table consist of **primary key** that are set to **auto increment.** Some tables have foreign key to link them to other tables. The following picture gives the details of fields and attributes of tables:

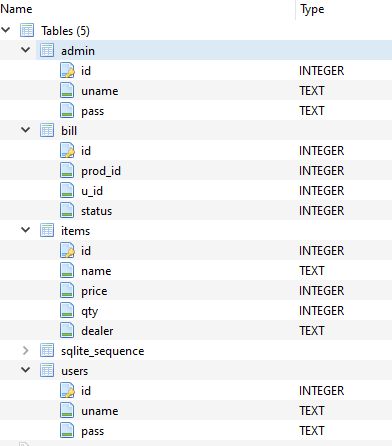


FIG: 1(“displays the details of tables.”)

**TOOLS USED:**

The following are the list of tools used in making up of the project:

* Atom text editor
* SQLite browser
* Windows snipping tool (helped in report pics)
* Command prompt

**Language used:**

* Python 3
* SQL

**FEATURES OF PROJECT:**

* Login menu for Admin .
* Login menu for n number of user they can login or register their account.
* There is proper management of rights of users.
* Admin can add, update, delete, display and can check for out of stock items.
* User can buy and generate bill.
* There is automatic deduction of quantity as per purchase by users.
* Bill generator for user purchases.
* Default setting of admin password for system.
* Loading bar mechanism

**Concepts used:**

* **Insertion in database**
* **Deletion in database**
* **Updation in database**
* **Keys**
* **Joins**
* **Creation of database**

**CODE:**

The code is provided with this document in attachment

**Commands used:**

1.

CREATE TABLE admin (

id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,

uname TEXT UNIQUE,

pass TEXT UNIQUE

);

2.

CREATE TABLE items (

id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,

name TEXT UNIQUE,

price INTEGER,

qty INTEGER,

dealer TEXT

);

3.

CREATE TABLE bill (

id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,

prod\_id INTEGER,

u\_id INTEGER,

status INTEGER

);

4.

CREATE TABLE users (

id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,

uname TEXT UNIQUE,

pass TEXT UNIQUE

5.

SELECT uname, pass from admin

SELECT uname, pass from users

6.

INSERT OR IGNORE INTO items (name,price,qty,dealer)

VALUES ( ?,?,?,?)''', ( name,price,quantity,dealer )

7.

'UPDATE items SET price = ? WHERE id = ?',

(price,pid))

8.

'UPDATE items SET qty = ? WHERE id = ?',

(quantity,pid))

9.

'DELETE from items WHERE id = ?',

(pid,))

10.

'SELECT name from items where qty=?',

(0))

11.

'UPDATE items SET dealer = ? WHERE id = ?',

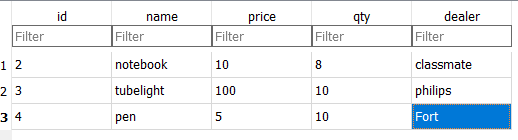
(quantity,pid)

12.select bill.id,bill.u\_id,items.name,items.price from bill join items on bill.prod\_id=items.id where bill.status=?",(0,)

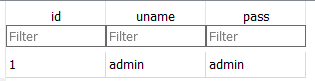
**NOTE: there are many more commands in the program do see in source code I have listed some out of them**

**TABLES IMAGES:**

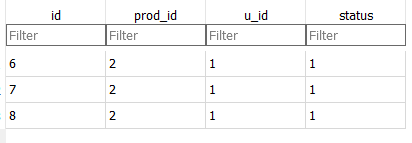
1. ITEMS:



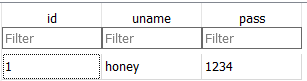
1. ADMIN:



1. BILL:



1. USERS:



**NOTE: there is a Video uploaded with attachment to show output.**