CSE 3330: Database and File Structures

Team Members:

Zahraa Hasan 1001550871

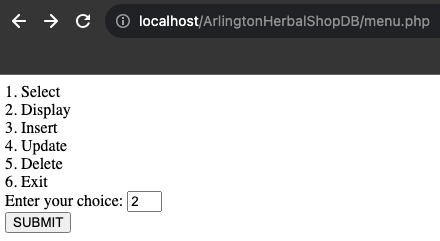
Faris Soepangat 1001374988

PROJECT PHASE 3 FALL 2023

Due Date: 23rd Nov 2023

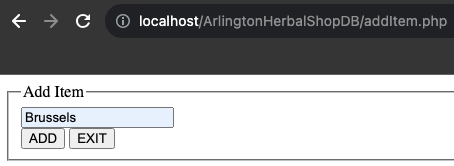
Part 1: Create a web interface

Q1. Display the ITEM details based on any one of the following: Item name or Item Id

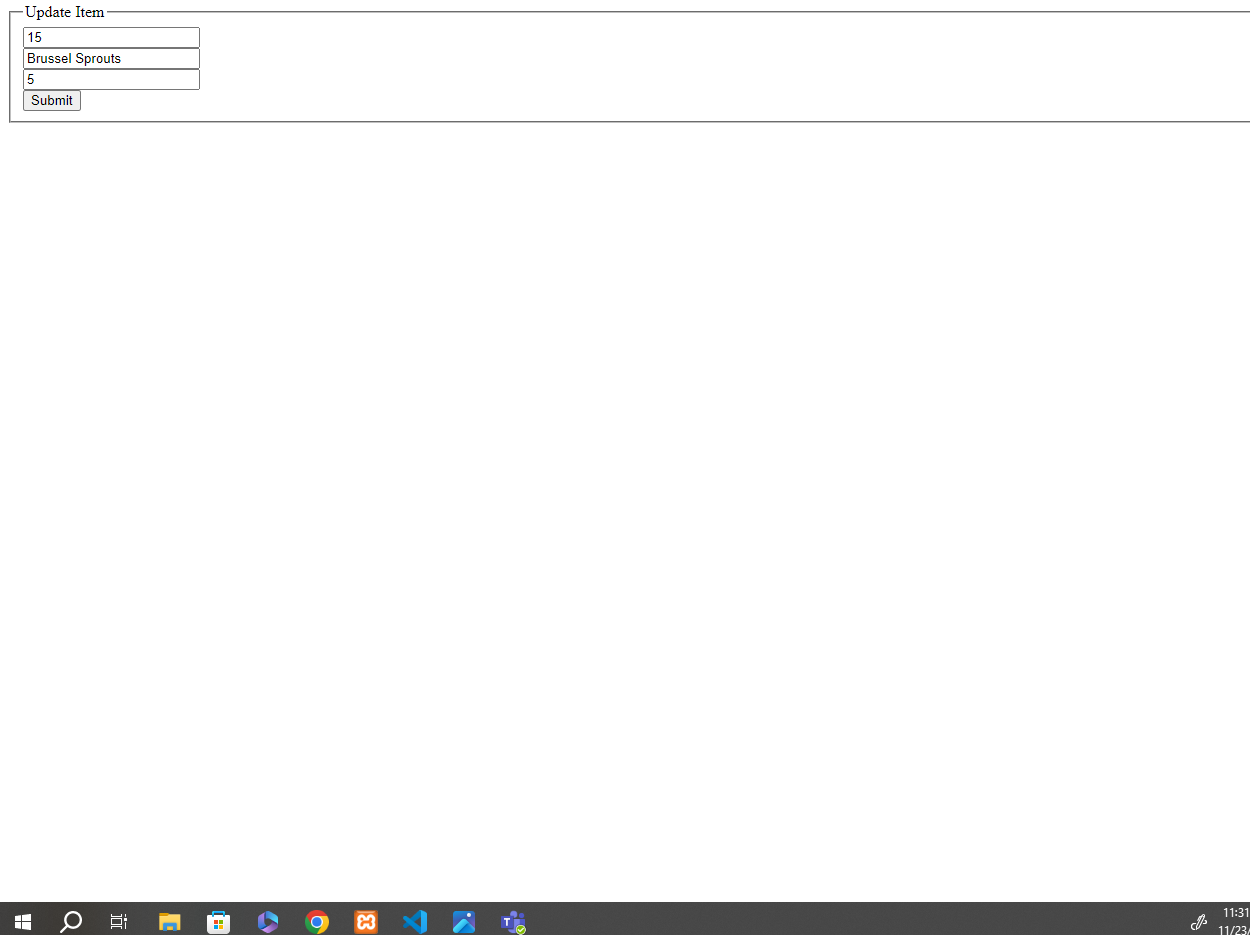




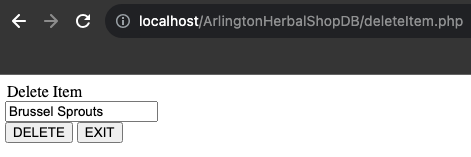
Q2. Insert a new item “Brussels” in the Arlington Herbal Shop database using the web interface you created



Q3. Update the item record that you just added “Brussels” to “Brussel Sprouts” using the web interface you created



Q4. Delete the item record for “Brussel Sprouts” that you just added using the web interface you created



Part 2: View-based questions:

Create a view **ItemView**that displays a list of records where each record is comprised of the itemId as **iId,** item name as **ItemName**, the number of items sold as **NoOfBoxes**, the item price as **ItemPrice**, the revenue generated by each item as **ItemRevenue**, and the number of customers as **ItemCustomers** who bought the items at any of the Arlington Herbal

Shop.

Write an SQL query to display the contents of the view **ItemView**.

CREATE VIEW ItemView AS

SELECT

I.iId AS iId,

I.Iname AS ItemName,

SUM(OI.Icount) AS NoOfBoxes,

I.Sprice AS ItemPrice,

SUM(OI.Icount \* I.Sprice) AS ItemRevenue,

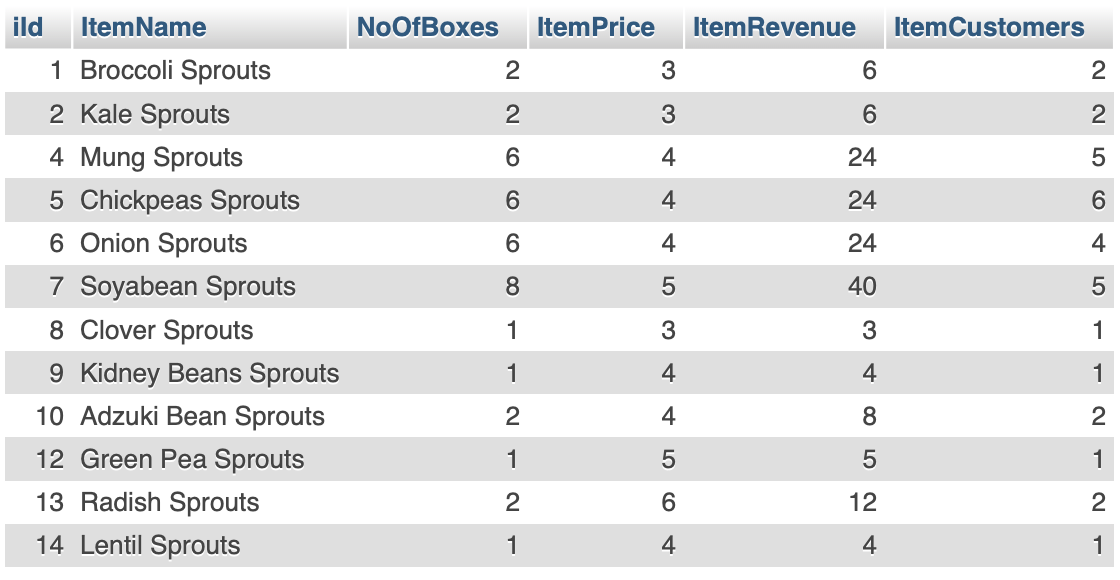
COUNT(DISTINCT O.cId) AS ItemCustomers

FROM ITEM I, ORDER\_ITEM OI, ORDERS O

WHERE I.iId = OI.iId AND OI.oId = O.oId

GROUP BY I.iId, I.Iname, I.Sprice;

SELECT \* FROM ItemView;



QV1: Use the view ItemView to retrieve a list of records where each record is comprised of item Id, item name, the number of boxes of items sold, and the price of each box of the item for all items that cost more than $3.00 and that have been bought by customers.

SELECT

iId,

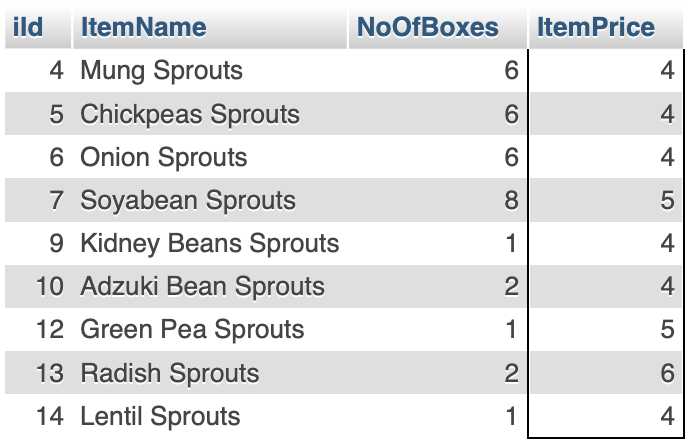
ItemName,

NoOfBoxes,

ItemPrice

FROM ItemView

WHERE ItemPrice > 3.00 AND ItemCustomers > 0;



QV2: Use the view ItemView to retrieve a list of records where each record is comprised of the item Name and the ItemRevenue for the item(s) that generated the minimum revenue in the database.

SELECT

ItemName,

ItemRevenue AS MinItemRevenue

FROM ItemView

WHERE ItemRevenue = (SELECT MIN(ItemRevenue) FROM ItemView);



QV3: Use the view ItemView to generate the min, max and average revenue generated by all the items in the ItemView.

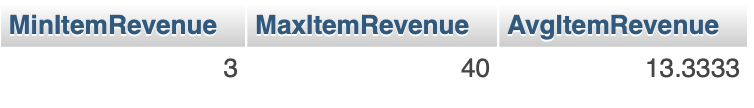
SELECT

MIN(ItemRevenue) AS MinItemRevenue,

MAX(ItemRevenue) AS MaxItemRevenue,

AVG(ItemRevenue) AS AvgItemRevenue

FROM ItemView;

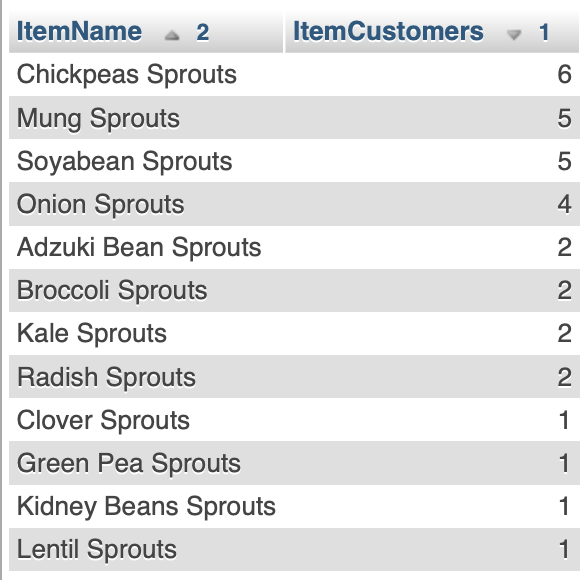


QV4: Use the view ItemView to retrieve a list of records where each record is comprised of an item name along with the number of customers who bought it. Sort the list by the number of customers in descending order followed by item names in an ascending order.

SELECT ItemName, ItemCustomers

FROM ItemView

ORDER BY ItemCustomers DESC, ItemName ASC;



QV5: Use the view ItemView to retrieve the total revenue earned, the total number of boxes sold and the average revenue per box sold by Arlington Herbal Shop as stored in the database.

SELECT

SUM(ItemRevenue) AS TotalRevenue,

SUM(NoOfBoxes) AS TotalNoOfBoxes,

AVG(ItemRevenue / NoOfBoxes) AS 'AvgRevenue/Box'

FROM ItemView;

