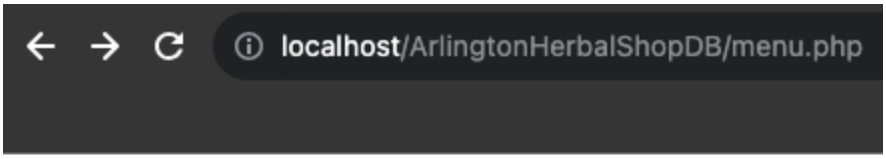
CSE 3330: Database and File Structures

Team Members:
Zahraa Hasan 1001550871
Faris Soepangat 1001374988

PROJECT PHASE 3 FALL 2023
Due Date: 23rd Nov 2023

Part 1: Create a web interface

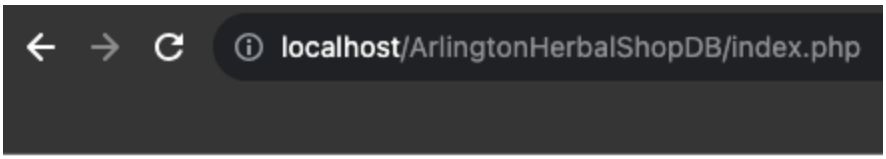
Q1. Display the ITEM details based on any one of the following: Item name or Item Id




1. Select
2. Display
3. Insert
4. Update
5. Delete
6. Exit

Enter your choice: 2

SUBMIT

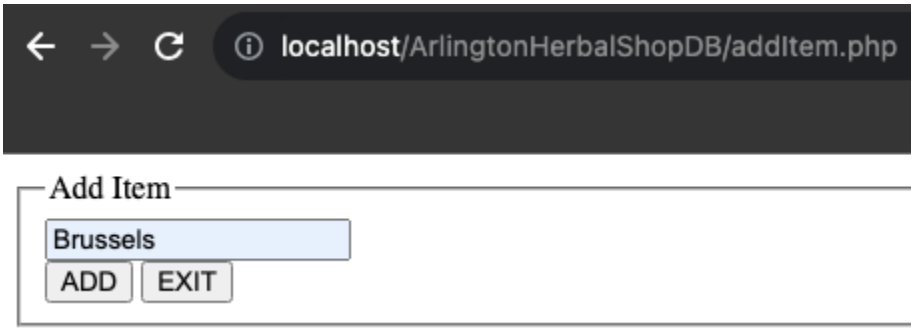



ITEMS

EXIT

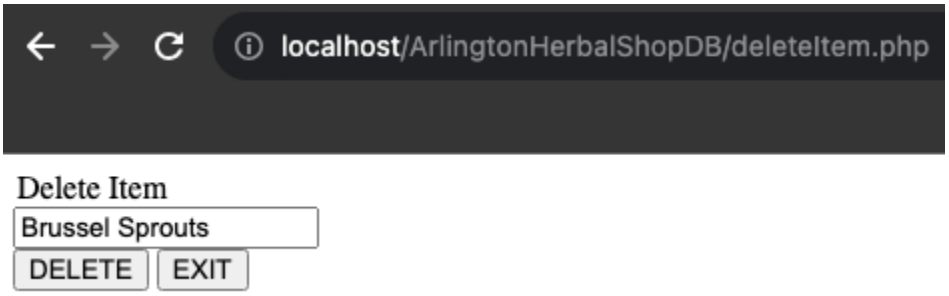
| iId | Iname | Sprice |
|---|---|---|
| 1 | Broccoli Sprouts | 3 |
| 2 | Kale Sprouts | 3 |
| 3 | Alfalfa Sprouts | 3 |
| 4 | Mung Sprouts | 4 |
| 5 | Chickpeas Sprouts | 4 |
| 6 | Onion Sprouts | 4 |
| 7 | Soyabean Sprouts | 5 |
| 8 | Clover Sprouts | 3 |
| 9 | Kidney Beans Sprouts | 4 |
| 10 | Adzuki Bean Sprouts | 4 |
| 11 | Beet Sprouts | 5 |
| 12 | Green Pea Sprouts | 5 |
| 13 | Radish Sprouts | 6 |
| 14 | Lentil Sprouts | 4 |

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;
```

| iId | ItemName | NoOfBoxes | ItemPrice | ItemRevenue | ItemCustomers |
|---|---|---|---|---|---|
| 1 | Broccoli Sprouts | 2 | 3 | 6 | 2 |
| 2 | Kale Sprouts | 2 | 3 | 6 | 2 |
| 4 | Mung Sprouts | 6 | 4 | 24 | 5 |
| 5 | Chickpeas Sprouts | 6 | 4 | 24 | 6 |
| 6 | Onion Sprouts | 6 | 4 | 24 | 4 |
| 7 | Soyabean Sprouts | 8 | 5 | 40 | 5 |
| 8 | Clover Sprouts | 1 | 3 | 3 | 1 |
| 9 | Kidney Beans Sprouts | 1 | 4 | 4 | 1 |
| 10 | Adzuki Bean Sprouts | 2 | 4 | 8 | 2 |
| 12 | Green Pea Sprouts | 1 | 5 | 5 | 1 |
| 13 | Radish Sprouts | 2 | 6 | 12 | 2 |
| 14 | Lentil Sprouts | 1 | 4 | 4 | 1 |

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;
```

| iId | ItemName | NoOfBoxes | ItemPrice |
|---|---|---|---|
| 4 | Mung Sprouts | 6 | 4 |
| 5 | Chickpeas Sprouts | 6 | 4 |
| 6 | Onion Sprouts | 6 | 4 |
| 7 | Soyabean Sprouts | 8 | 5 |
| 9 | Kidney Beans Sprouts | 1 | 4 |
| 10 | Adzuki Bean Sprouts | 2 | 4 |
| 12 | Green Pea Sprouts | 1 | 5 |
| 13 | Radish Sprouts | 2 | 6 |
| 14 | Lentil Sprouts | 1 | 4 |

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);
```

| ItemName | MinItemRevenue |
|---|---|
| Clover Sprouts | 3 |

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;
```

| MinItemRevenue | MaxItemRevenue | AvgItemRevenue |
|---|---|---|
| 3 | 40 | 13.3333 |

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;
```

| ItemName ▲ 2 | ItemCustomers ▼ 1 |
|---|---|
| Chickpeas Sprouts | 6 |
| Mung Sprouts | 5 |
| Soyabean Sprouts | 5 |
| Onion Sprouts | 4 |
| Adzuki Bean Sprouts | 2 |
| Broccoli Sprouts | 2 |
| Kale Sprouts | 2 |
| Radish Sprouts | 2 |
| Clover Sprouts | 1 |
| Green Pea Sprouts | 1 |
| Kidney Beans Sprouts | 1 |
| Lentil Sprouts | 1 |

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;
```

| TotalRevenue | TotalNoOfBoxes | AvgRevenue/Box |
|---|---|---|
| 160 | 38 | 4.08333333 |