

Assignment 10 - Views (50 points)
Due Date: Monday, April 6th, 11:59 PM

Objectives

This assignment focuses on how to create and use views and indexes.

This assignment uses the tables associated with the *bookstore* database. Write SQL statements to perform the following queries:

Query 1:	<p>Create a view named <i>customer_address</i> that shows the shipping for each customer.</p> <p>This view should return these columns from the CUSTOMER table: customer_id, email_address, last_name, and first_name.</p> <p>This view should return these columns from the ADDRESS table: ship_line1, ship_line2, ship_city, ship_state, and ship_zip.</p> <p>Write a SELECT statement that returns these columns from the <i>customer_address</i> view: customer_id, last_name, first_name, ship_city and ship_state.</p>
Query 2:	<p>Write an SQL statement that updates the customer table using the <i>customer_address</i> view you created in query 1. Set the first line of the shipping address to “1990 Westwood Blvd.” for the customer with an ID of 11119.</p>
Query 3:	<p>Create a view named <i>product_summary</i>. This view should return summary information about each product.</p> <p>Each row should include product_id, order_count (the number of times the product has been ordered) and order_total (the total sales for the product).</p> <p>Write a SELECT statement that returns all the columns from the <i>product_summary</i> view.</p>
Query 4:	<p>Creates a view named <i>cheap_products</i> whose subquery retrieves products only where the price is less than \$50. Add a CHECK OPTION constraint.</p> <p>Write a SELECT statement that returns all the columns from the <i>cheap_products</i>.</p>
Query 5:	<p>Write an INSERT statement that adds this row to the <i>cheap_products</i> view</p> <p style="margin-left: 40px;">Product ID: 17888 Category ID: 41 product_code: book db Product Name: Fundamental of Database Systems List_price: 45.99 Description: Fundamental of Database Systems Elmasri discount_percent: 20.00 date_added : 2015-06-01 11:12:59 vendor_id : 2</p>

CS 150A | Assignment #10
Spring 2020

Query 6:	<p>Create a view that lists the first name and phone number of the contact person at each vendor. Don't include the vendor's ID in the view. Name the view <i>contact</i>.</p> <p>Change the <i>contact</i> view so that no users can accidentally perform DML operations on the view.</p> <p>Write a SELECT statement that returns all the columns from the <i>contact</i> view.</p>																																																
Query 7:	<p>Create a view named <i>order_items</i> that returns columns from the ORDERS, ORDERITEMS, and PRODUCT tables.</p> <p>This view should return these columns from the ORDERS table: order_id, order_date, tax_amount, and ship_date.</p> <p>This view should return these columns from the ORDERITEMS table: item_price, discount_amount, final_price (the discount amount subtracted from the item price), quantity, and item_total (the calculated total for the item).</p> <p>This view should return the product_name column from the PRODUCT Table.</p> <p>Write a SELECT statement that returns all the columns from the <i>order_items</i> view.</p>																																																
Query 8:	<p>Create a view that lists the product ID and product name for each product along with the first name and phone number of the person to contact if the product needs to be reordered. Name the view <i>reorder_info</i>.</p> <p>Write a SELECT statement that returns all the columns from the <i>reorder_info</i> view.</p>																																																
Query 9:	<p>Write a SELECT statement that returns these columns from the ORDERS table:</p> <ul style="list-style-type: none">▪ The card_number column▪ The length of the card_number column▪ The last four digits of the card_number column <p>When you get that working right, add the columns that follow to the result set. This is more difficult because these columns require the use of functions within functions.</p> <p>A column that displays the last four digits of the card_number column in this format: XXXX-XXXX-XXXX-1234. In other words, use Xs for the first 12 digits of the card number and actual numbers for the last four digits of the number.</p> <p>Sample run:</p> <table><tr><th>card_number</th><th>card_number_length</th><th>last_four_digits</th><th>formatted_number</th></tr><tr><td>4111111111111111</td><td>16</td><td>1111</td><td>XXXX-XXXX-XXXX-1111</td></tr><tr><td>4012888888881881</td><td>16</td><td>1881</td><td>XXXX-XXXX-XXXX-1881</td></tr><tr><td>4111111111111111</td><td>16</td><td>1111</td><td>XXXX-XXXX-XXXX-1111</td></tr><tr><td>378282246310005</td><td>15</td><td>0005</td><td>XXXX-XXXX-XXXX-0005</td></tr><tr><td>4111111111111333</td><td>16</td><td>1333</td><td>XXXX-XXXX-XXXX-1333</td></tr><tr><td>6011111111111117</td><td>16</td><td>1117</td><td>XXXX-XXXX-XXXX-1117</td></tr><tr><td>5555555555554444</td><td>16</td><td>4444</td><td>XXXX-XXXX-XXXX-4444</td></tr><tr><td>4012888888881991</td><td>16</td><td>1991</td><td>XXXX-XXXX-XXXX-1991</td></tr><tr><td>4111111111111333</td><td>16</td><td>1333</td><td>XXXX-XXXX-XXXX-1333</td></tr><tr><td>4012888888881881</td><td>16</td><td>1881</td><td>XXXX-XXXX-XXXX-1881</td></tr><tr><td>4012888888881881</td><td>16</td><td>1881</td><td>XXXX-XXXX-XXXX-1881</td></tr></table>	card_number	card_number_length	last_four_digits	formatted_number	4111111111111111	16	1111	XXXX-XXXX-XXXX-1111	4012888888881881	16	1881	XXXX-XXXX-XXXX-1881	4111111111111111	16	1111	XXXX-XXXX-XXXX-1111	378282246310005	15	0005	XXXX-XXXX-XXXX-0005	4111111111111333	16	1333	XXXX-XXXX-XXXX-1333	6011111111111117	16	1117	XXXX-XXXX-XXXX-1117	5555555555554444	16	4444	XXXX-XXXX-XXXX-4444	4012888888881991	16	1991	XXXX-XXXX-XXXX-1991	4111111111111333	16	1333	XXXX-XXXX-XXXX-1333	4012888888881881	16	1881	XXXX-XXXX-XXXX-1881	4012888888881881	16	1881	XXXX-XXXX-XXXX-1881
card_number	card_number_length	last_four_digits	formatted_number																																														
4111111111111111	16	1111	XXXX-XXXX-XXXX-1111																																														
4012888888881881	16	1881	XXXX-XXXX-XXXX-1881																																														
4111111111111111	16	1111	XXXX-XXXX-XXXX-1111																																														
378282246310005	15	0005	XXXX-XXXX-XXXX-0005																																														
4111111111111333	16	1333	XXXX-XXXX-XXXX-1333																																														
6011111111111117	16	1117	XXXX-XXXX-XXXX-1117																																														
5555555555554444	16	4444	XXXX-XXXX-XXXX-4444																																														
4012888888881991	16	1991	XXXX-XXXX-XXXX-1991																																														
4111111111111333	16	1333	XXXX-XXXX-XXXX-1333																																														
4012888888881881	16	1881	XXXX-XXXX-XXXX-1881																																														
4012888888881881	16	1881	XXXX-XXXX-XXXX-1881																																														

Submission Instructions:

- For each of the queries above, submit the query and the result from running the query. Please use the provided SQL file to write your submissions.
- You will need to label your assignment with your first initial, last name, and the name of the assignment.
Example: *hibrahim_assignment10.sql* and *hibrahim_assignment10.txt*
- Zip the two files together to create one compressed file. **Example:** *hibrahim_assignment10.zip*
- Upload the compressed file into Canvas.
- Remember to include the query number as a comment at each step.
- Read your output TXT file before you submit it.