Yongeun Kwon (Jen) (A01263922)

Section 1: 15 Marks Answer all questions.

1. ALTER command is used for – Write all that apply. (2 marks) => a, b, c, e, f

a. Drop a Column

b. Add a Column

c. Change datatype of existing column

e. Add Primary Key

f. Add Foreign Key

d. Create INDEX

Wrong answer because the content of the option is Create INDEX, not Add INDEX. Alter command could be used to add Index if the table is optimized table. Because the operation 'ALTER TABLE ADD INDEX' is supported only with memory optimized tables.

2. Based on the Consumer Table script, what does this error indicate in the below snippet? Name the column and the value that is causing the error. (2 marks)

=> The error message indicates that there is an attempt to insert data into the “LastName” column of the “Consumer” table, but the inserted value is too long for the column’s defined length (Varchar (20)). The “LastName” column can hold strings with a maximum length of 20 characters.

“Kaycee DeVere Hunt” causes error because it has 21 characters, which exceeds the defined length of 20 characters. To fix this error, there are 2 ways. At First, truncate the insert value to fit within the defined length. Secondly, increase the size of the “LastName” column to accommodate the longer value.

For example, to alter the length of the column longer.

ALTER TABLE Consumer ALTER COLUMN LastName varchar(30);

3. Based on the Order Table script, what does this error indicate in the below snippet? Name the column and the value that is causing the error. (2 marks)

The error message indicates that there is an attempt to insert an explicit value into the identity column “OrderId” of the “Order” table. The identity column is defined to generate its values automatically using the IDENTITY property (such as AUTO\_INCREMENT). It has an OrderId that starts from 1000 and increases by 3. Since the identity column is set to auto-increment, explicit values cannot be inserted into it.

To fix the error,

1. We should remove the explicit value for the “OrderId” column from the INSERT statement, and the value will be automatically generated by the IDENTITY property:

INSERT INTO [ORDER] (CustomerId, OrderDate) Values (50, '2023-07-22 10:15:00');

1. Use the SET IDENTITY\_INSERT [Table] ON statement before inserting operation if we need to insert specific values into the 'OrderId' column.

SET IDENTITY\_INSERT [ORDER] ON;

INSERT INTO [ORDER] (OrderId, CustomerId, OrderDate) VALUES (1000, 50, '2023-07-22 10:15:00');

SET IDENTITY\_INSERT [ORDER] OFF;

1. If we do not use IDENTITY property when creating the table, we can insert explicit value.

CREATE TABLE [ORDER](

OrderId BIGINT PRIMARY KEY,

CustomerId INT NOT NULL CONSTRAINT FK\_ORDER\_CustomerId FOREIGN KEY REFERENCES Customer (CustomerId),

OrderDate DATETIME NOT NULL

);

4. Based on the Customer and Order table script, a SQL query is written to join both the tables. There is no error in the SQL query. But the query is not producing any results although data exists. Identify the issue in the code in the SQL JOIN query. Write correct SQL query to fix it.

CustomerId and OrderId are not the corresponding fields between the two tables. To fix this error, we should join CustomerId from Customer table and CustomerId from Order table.

SELECT \*

FROM Customer C

INNER JOIN [ORDER] O

ON C.CustomerId = O.CustomerId

5. Based on the below snippet, what can you interpret from the error? How do you fix it ? (1 mark)

Ambiguous column name “Name” indicates that the column “Name” is present in both tables involved in the JOIN operation. With this query, SQL cannot determine which one we are referring to in SELECT statement.

To fix this error, we need to specify which table’s ‘Name’ column we want to select. There are two ways to address this, specify with A (ProductCategory) or B (ProductSubcategory) table.

1. Specifying with A table.

SELECT A.Name

FROM Production.ProductCategory A

INNER JOIN Production.ProductSubcategory B

ON A.ProductCategoryId = B.ProductCategoryID

2. Specifying with B table.

SELECT A.Name

FROM Production.ProductCategory A

INNER JOIN Production.ProductSubcategory B

ON A.ProductCategoryId = B.ProductCategoryID

6. Assume there is a stored procedure sp\_GetEmployeeSalary in the database. The stored procedure accepts one parameter which is EmployeeId and is defined as Varchar(15) datatype. Write the command to execute the procedure for EmployeeId = A5754 (1 mark)

EXEC sp\_GetEmployeeSalary 'A5754';

7. As a database developer you were asked to develop a solution to capture any insert, update or delete changes on the data in BUDGET\_ALLOCATION table for audit purposes. What is your solution? Explain in 2-3 sentences. (1 mark)

To capture changes on a table for auditing purposes is to create a trigger on the table. A Trigger is a special type of stored procedure that automatically runs when an event occurs in the database server. In this case, we could create triggers on the BUDGET\_ALLOCATION table for INSERT, UPDATE, and DELETE operations.

8. As a database developer you were asked to remove all the data in TRANSACTON\_LOG table. You observed log\_data column stores the date. Write SQL query to satisfy the requirement. (2 marks)

1. DELETE: To remove existing data

DELETE FROM TRANSACTION\_LOG;

2. TRUNCATE: To remove existing data in a table

TRUNCATE TABLE TRANSACTON\_LOG;

9. Identify the problem in the below snippet. The code when executed takes time and server is not responsive after few mins. How do you fix the code ? ( 1 mark)The problem of this code is that it creates an infinite loop within while statement. The variavle @i is not being incremented within the WHILE loop. The condition will always be true because @i is always 0, and the loop will continue indefinitely, consuming server resources.

To address this, we need to increment @i within the loop. I would increment @i by 1.

DECLARE @i INT

SET @i = 0

WHILE @i <= 10

BEGIN

SELECT @i

SET @i = @i + 1

END

10. Based on the ERD below, what would be the datatype of iso\_country\_code column in Region table? Provide explanation. (2 marks)

The datatype of the iso\_country\_code column in the Region table would be VARCHAR (10). This is because it is a FOREIGN KEY (FK) references the iso\_country\_code PRIMARY KEY (PK) of the Country table. For Foreign key, the data types of both the PRIMARY KEY column and the FOREIGN KEY column need to match.

Section 2: (15 Marks) Answer all questions based on the below ERD and data. Table script and data is provided in case you prefer to restore. FileName: Section2Dataset2.sql

11) Write a query to count total records in TransactionDetails. Rename the column as TotalTransactions. Expected result is shown in the below snippet. (3 marks)

SELECT COUNT(\*) AS TotalTransactions

FROM TransactionDetails;

12) Write a query to find total amount by paymentTypeId for each store. Rename the columns as shown in the below snippet (5 marks)

SELECT SUM(Amount) AS TotalAmount, s.Name AS StoreName, pt.Name AS PaymentMethod

FROM TransactionDetails td

INNER JOIN Store s ON s.StoreID = td.StoreId

INNER JOIN PaymentType pt ON pt.PaymentTypeId = td.PaymentTypeId

GROUP BY s.Name, pt.Name;

13) Write a query to display “Name” from Customer table and rename the column as “CustomerName, “Name” from Store table and rename the column as “StoreName”, PurchaseDateTime. Sort the results by PurchaseDateTime ascending order. Expected Result is shown in the below Snippet. (4 marks)

SELECT c.Name AS CustomerName, s.Name AS StoreName, td.PurchaseDateTime

FROM TransactionDetails td

INNER JOIN Customer c ON c.CustomerId = td.CustomerId

INNER JOIN Store s ON s.StoreID = td.StoreId

ORDER BY td.PurchaseDateTime ASC;

14) Write a SQL query to find Total amount for each store. Sort the results by TotalAmount in ascending order. Rename the columns as shown in the below snippet (3 marks) Expected Result is shown in the below snippet.

SELECT SUM(Amount) AS TotalAmount, s.Name AS StoreName

FROM TransactionDetails td

INNER JOIN Store s ON s.StoreID = td.StoreId

GROUP BY s.Name

ORDER BY TotalAmount ASC;