

Лабораторна робота 3

Знайомство з основними командами маніпулювання даними мови SQL

Підготовка до виконання роботи

1. Перемістити файл **SK.mdb** створений під час виконання попередньої лабораторної роботи до нового каталогу (наприклад, D:\ACC_LAB_3).
2. Запустити СУБД Access.
3. Відкрити раніше створену базу даних.

Виконання роботи

I. Команда SELECT SQL. Обробка даних з використанням запитів SELECT.

Для виконання наступних запитів SELECT SQL необхідно відкрити вкладку “Create” у вікні Access.

1. Вивести на екран список товарів, поставлених постачальником 1 (Ivanov I.I. PE) за договором 1 (рисунок 3.1).

```
SELECT
    Contracts.ContractNumber,
    Supplied.Product,
    Supplied.Amount,
    Supplied.PricePerItem,
    Suppliers.SupplierName,
    Suppliers.SupplierID
FROM
    Suppliers INNER JOIN
```

```

(Contracts INNER JOIN Supplied ON Contracts.ContractNumber =
Supplied.ContractNumber)
ON Suppliers.SupplierID = Contracts.SupplierID
WHERE
(((Contracts.ContractNumber)=1) AND ((Suppliers.SupplierID)=1));

```

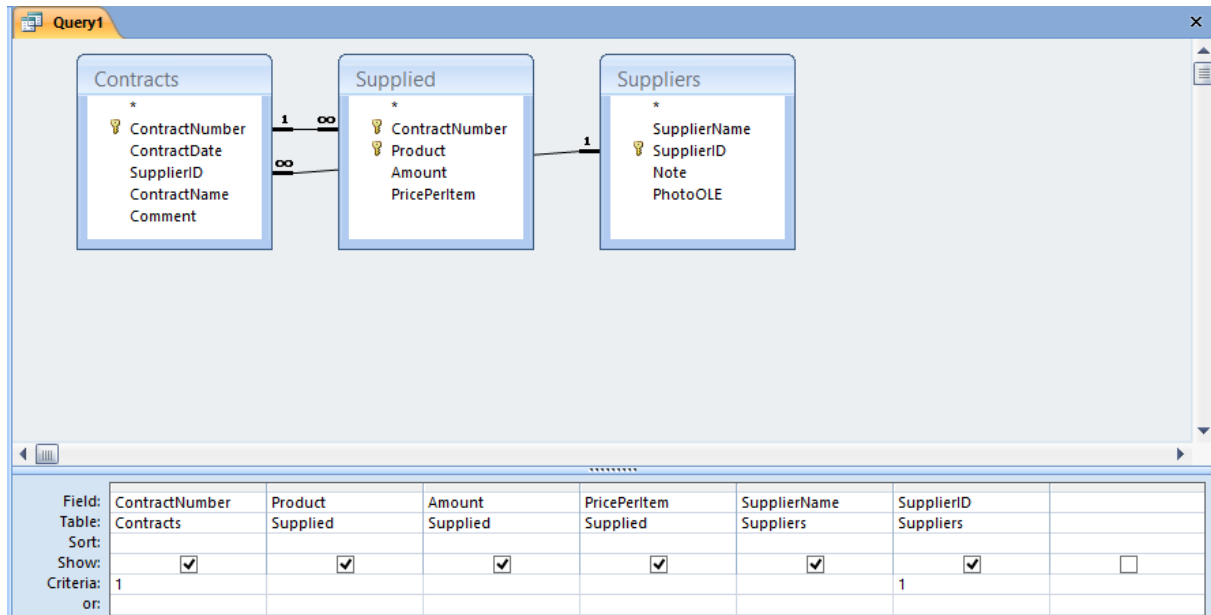


Рисунок 3.1

2. Вивести на екран список товарів, поставлених постачальником 1 (Іванов І.І. РЕ) в період з 9/1/1999 по 9/12/1999.

Для цього необхідно використовувати наступний SQL запит:

```

SELECT
    Supplied.ContractNumber,
    Contracts.ContractDate,
    Supplied.Product,
    Supplied.Amount,
    Supplied.PricePerItem,
    Suppliers.SupplierName
FROM
    Suppliers INNER JOIN

```

```

        (Contracts INNER JOIN Supplied ON Contracts.ContractNumber =
        Supplied.ContractNumber)
    ON Suppliers.SupplierID = Contracts.SupplierID
WHERE
    (((Contracts.ContractDate) Between #9/5/1999# And #9/12/1999#) AND
    ((Suppliers.SupplierID)=1));

```

Також можна використовувати дизайнер запитів (рисунок 3.3), але введення команд SQL є більш прийнятним та допомагає запобігти виникненню помилок.

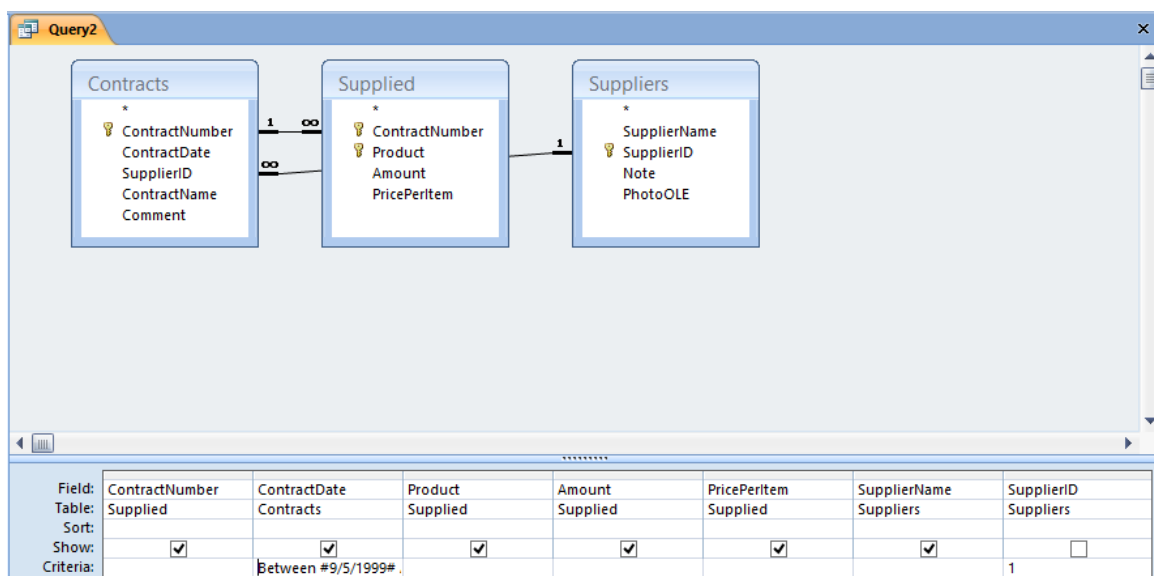


Рисунок 3.2

3. Вивести на екран список товарів, поставлених в 9 місяці 1999 з виводом найменування постачальника і дати поставки.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.3):

```

SELECT
    Contracts.ContractNumber,
    Contracts.ContractDate,

```

```

Supplied.Product,
Supplied.Amount,
Supplied.PricePerItem,
(Supplied.Amount * Supplied.PricePerItem) AS Total,
Suppliers.SupplierName
FROM
Suppliers INNER JOIN
    (Contracts INNER JOIN Supplied ON Contracts.ContractNumber =
    Supplied.ContractNumber)
    ON Suppliers.SupplierID = Contracts.SupplierID
WHERE
    Month(Contracts.ContractDate) = 9 AND Year(Contracts.ContractDate) =
1999

```

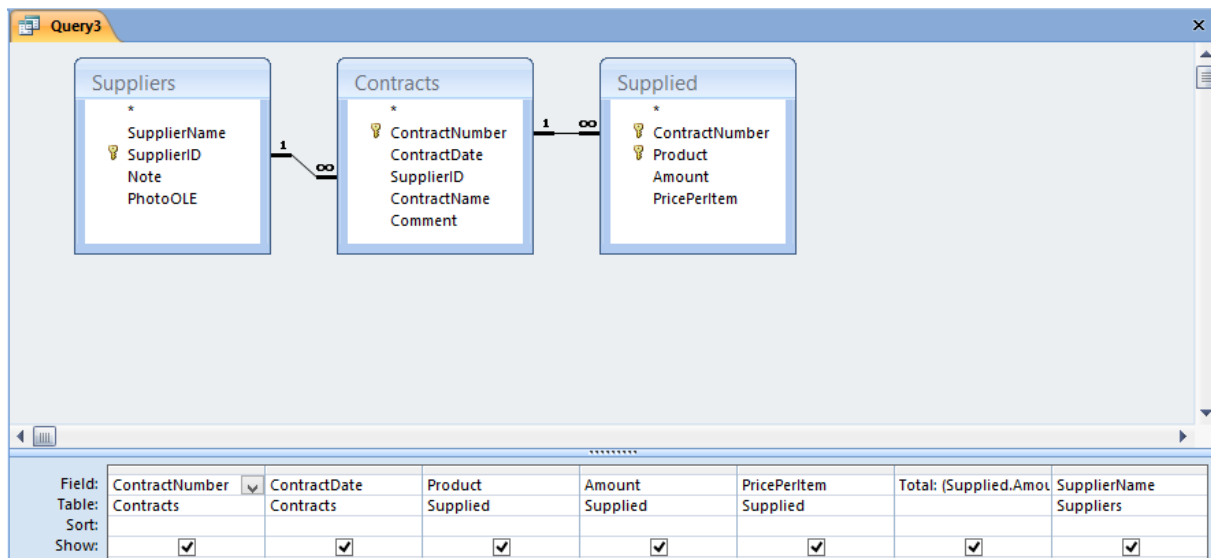


Рисунок 3.3

4. Вивести на екран список договорів (номер, дата), загальна кількість поставлених товарів і загальну суму за кожним договором (розмір партії помножити на ціну за штуку і підсумувати за договором). Список повинен бути відсортований в порядку зростання номерів договорів.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.4):

```
SELECT
    Contracts.ContractNumber,
    Contracts.ContractDate,
    Sum(Supplied.Amount) AS [TotalAmount],
    Sum(Supplied.Amount * Supplied.PricePerItem) AS [TotalPrice]
FROM
    Contracts INNER JOIN
        Supplied ON Contracts.ContractNumber = Supplied.ContractNumber
GROUP BY Contracts.ContractNumber, Contracts.ContractDate
ORDER BY Contracts.ContractNumber
```

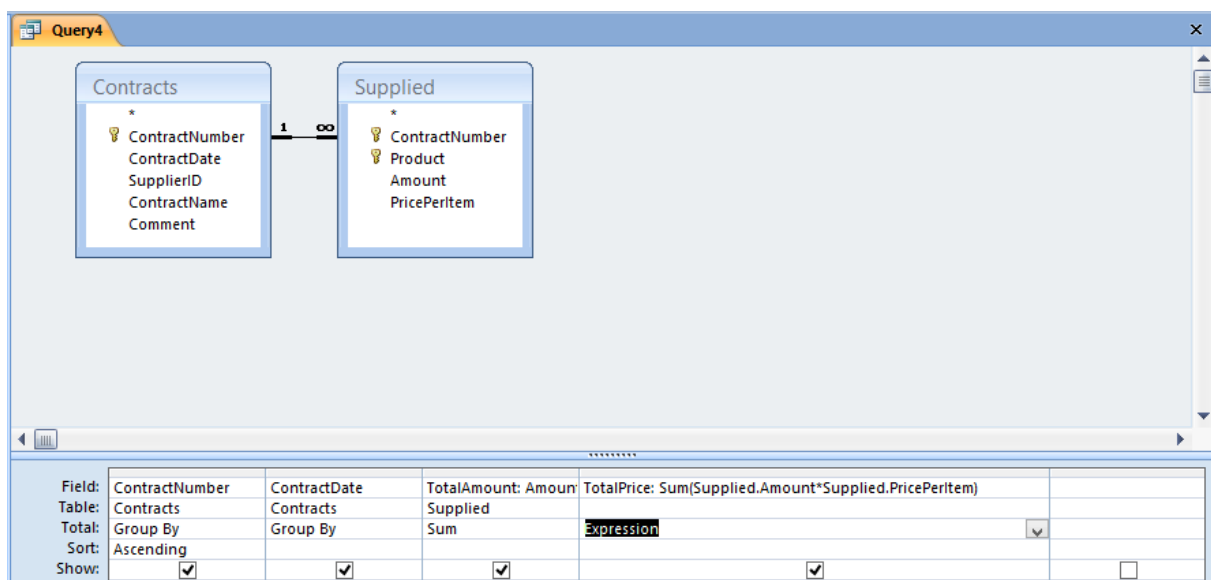


Рисунок 3.4

5. Вивести на екран список договорів (номер, дата) і загальну суму по кожному договору (розмір партії помножити на ціну за штуку і підсумувати за договором). Список повинен бути відсортований в порядку зростання загальних сум по кожному договору. Крім того, на список має бути накладено умова фільтрації, що складається у ви-

ключенні з результату запиту записів, для яких номер договору більше 3.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.5):

```
SELECT
    Contracts.ContractNumber,
    Contracts.ContractDate,
    Sum(Supplied.Amount * Supplied.PricePerItem) AS [TotalPrice]
FROM
    Contracts INNER JOIN
        Supplied ON Contracts.ContractNumber = Supplied.ContractNumber
GROUP BY Contracts.ContractNumber, Contracts.ContractDate
HAVING Contracts.ContractNumber <= 3
ORDER BY Sum(Supplied.Amount * Supplied.PricePerItem)
```

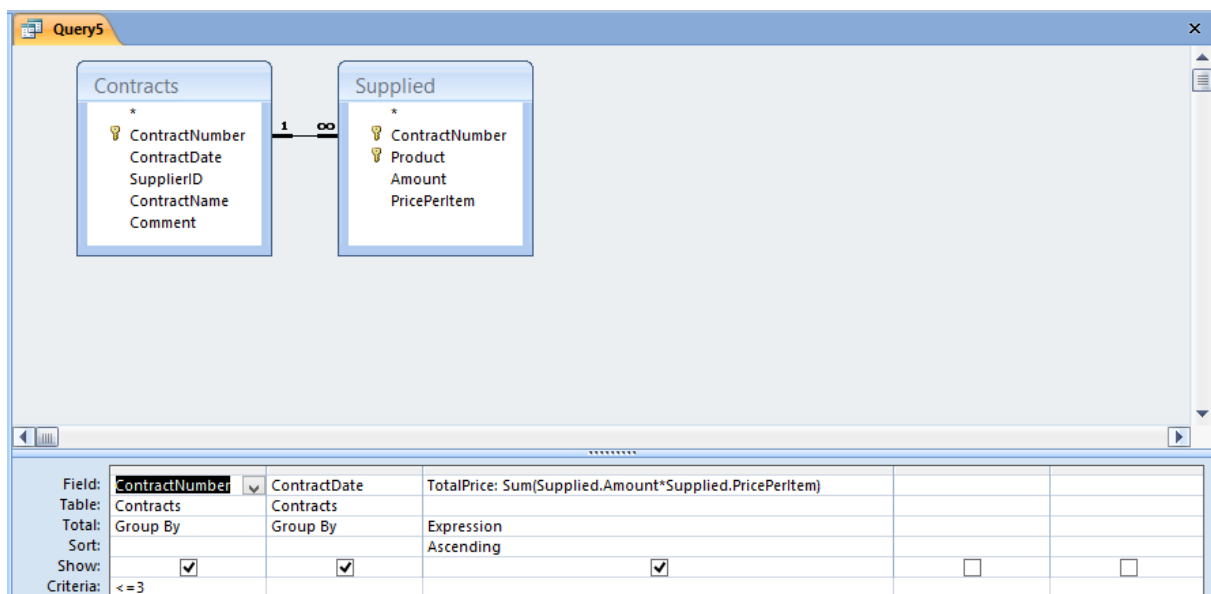


Рисунок 3.5

6. Вивести на екран відомості про найбільшу за розміром комплекти з тих договорів із зазначенням постачальника, а також номери і дати договору.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.6):

```
SELECT
    Contracts.ContractNumber,
    Contracts.ContractDate,
    Supplied.Product,
    Supplied.Amount,
    Suppliers.SupplierName
FROM
    Suppliers INNER JOIN
        (Contracts INNER JOIN Supplied ON Contracts.ContractNumber =
            Supplied.ContractNumber)
        ON Suppliers.SupplierID = Contracts.SupplierID
WHERE
    Supplied.Amount = (SELECT Max(Amount) FROM Supplied)
```

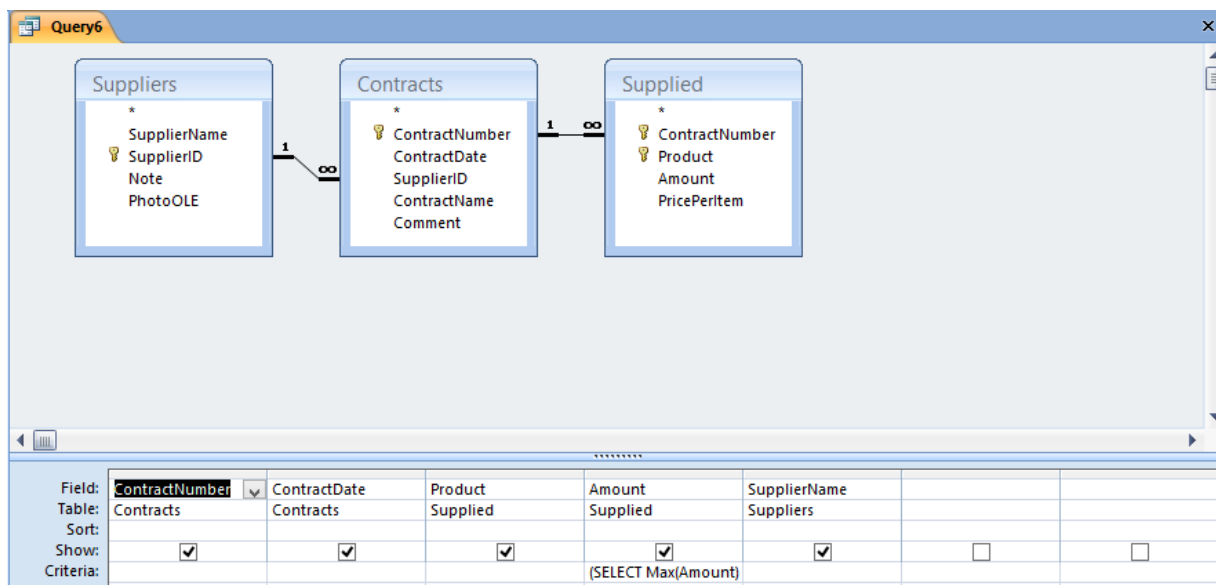


Рисунок 3.6

7. Вивести на екран список постачальників (найменування та код), з якими не було укладено жодного договору.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.7):

Варіант 1

```
SELECT
    Suppliers.SupplierName,
    Suppliers.SupplierID
FROM
    Suppliers
WHERE
    Suppliers.SupplierID NOT IN (SELECT SupplierID FROM Contracts)
```

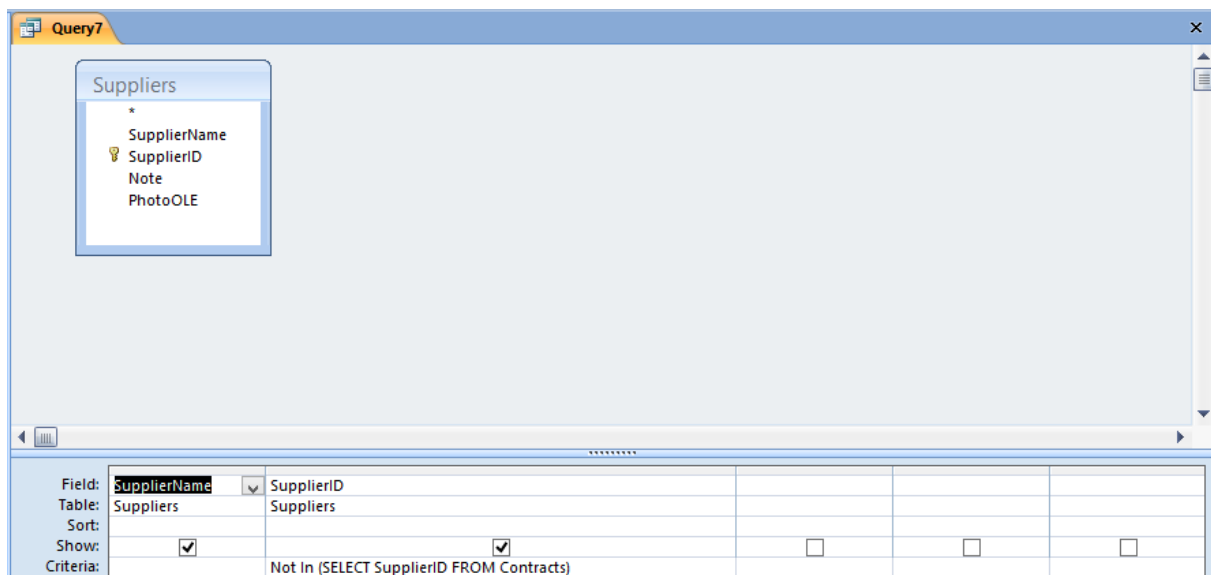


Рисунок 3.7а

Варіант 2

```
SELECT
    Suppliers.SupplierName,
    Suppliers.SupplierID
FROM
    Suppliers
WHERE
    Suppliers.SupplierID <> ANY(SELECT SupplierID FROM Contracts)
```

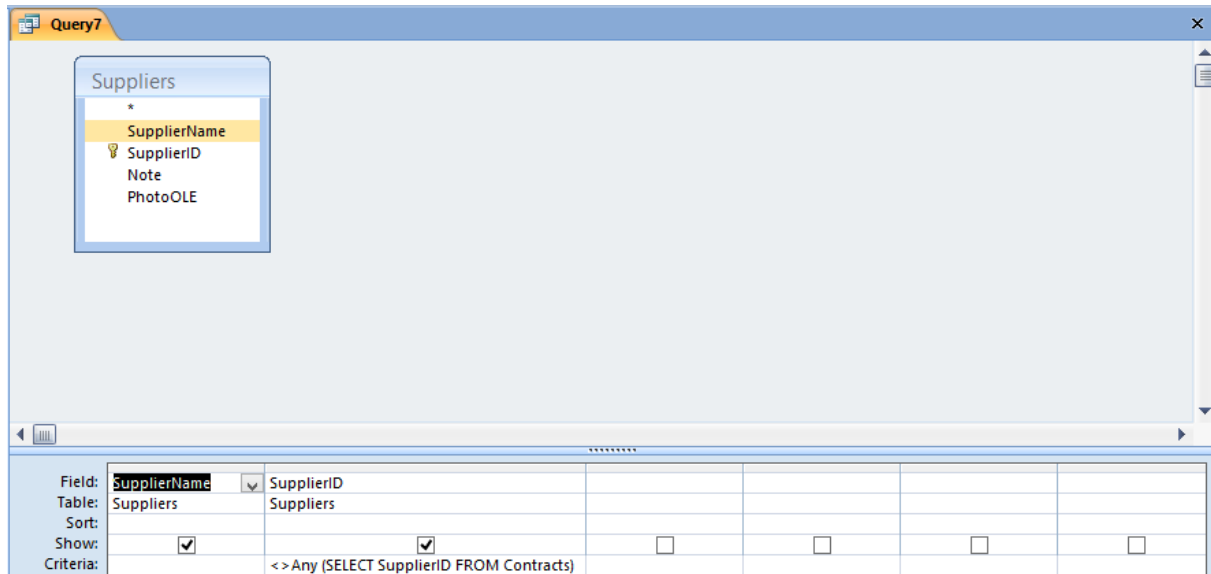



Рисунок 3.76

8. Вивести на екран список найменувань поставлених товарів із зазначенням середньої ціни поставки за одиницю (незалежно від поставачальника).

Використати наступну SQL команду або дизайнер запитів (рисунок 3.8):

```
SELECT
    Supplied.Product,
    Avg(Supplied.PricePerItem) AS [AvgPricePerItem]
FROM
    Supplied
GROUP BY Supplied.Product
```

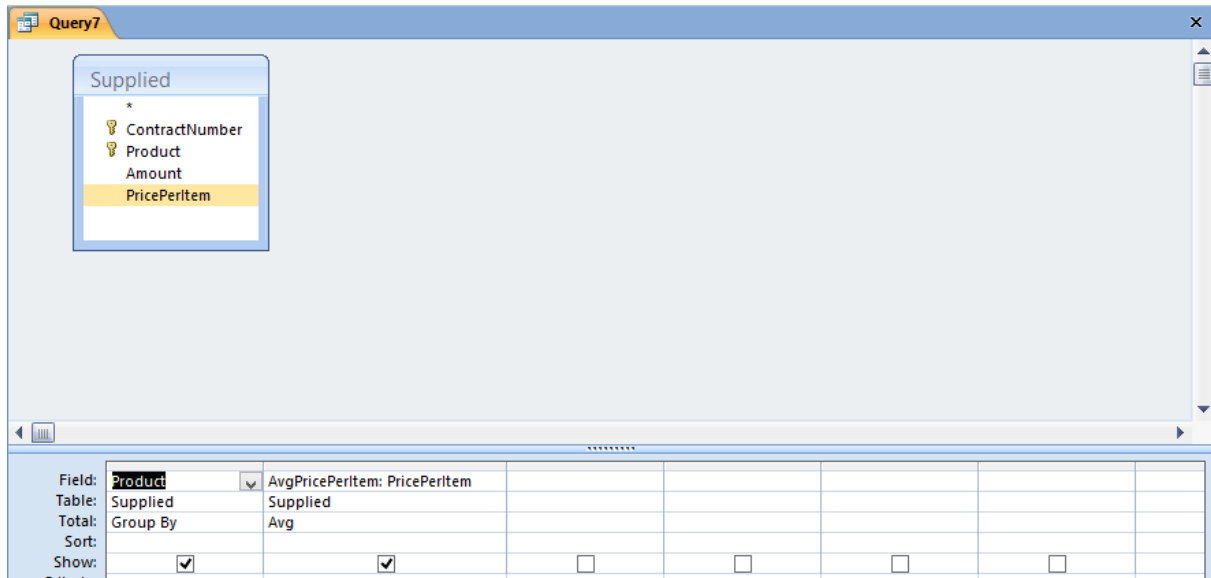


Рисунок 3.8

9. Вивести на екран список товарів (найменування, кількість і ціна, постачальник), для яких ціна за одиницю більше середньої.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.9):

```
SELECT
    Supplied.Product,
    Supplied.Amount,
    Supplied.PricePerItem,
    Suppliers.SupplierName
FROM
    Suppliers INNER JOIN
        (Contracts INNER JOIN Supplied ON Contracts.ContractNumber =
            Supplied.ContractNumber)
    ON Suppliers.SupplierID = Contracts.SupplierID
WHERE
    Supplied.PricePerItem > (SELECT Avg(PricePerItem) FROM Supplied)
```

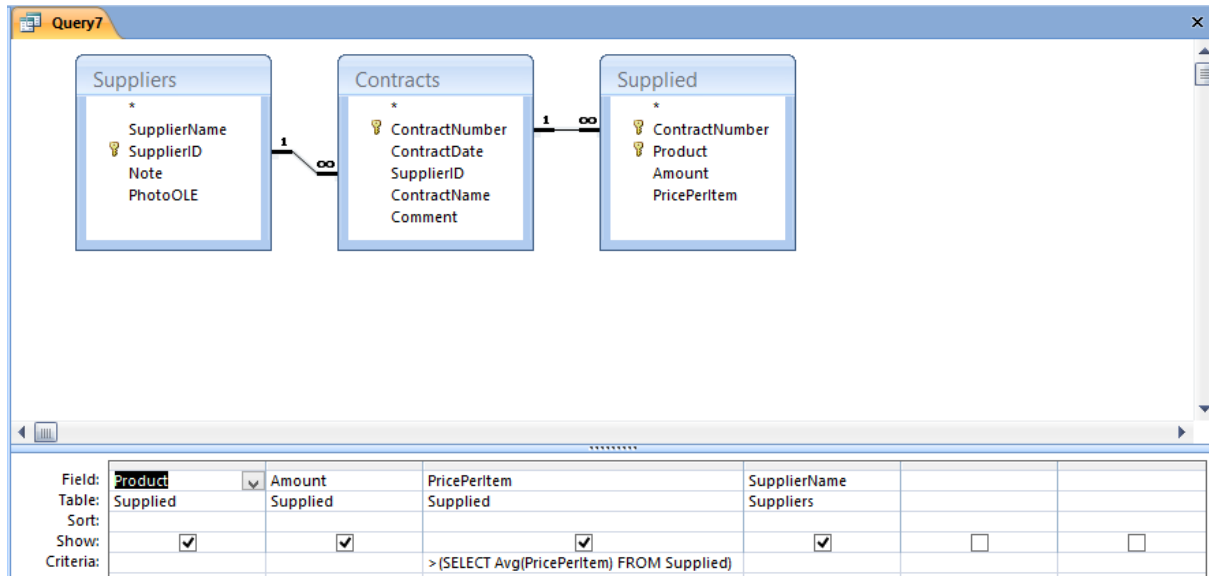


Рисунок 3.9

10. Вивести на екран відомості про п'ять найдорожчих товари (найменування, ціна за одиницю, постачальник).

Використати наступну SQL команду або дизайнер запитів (рисунок 3.10):

```
SELECT TOP 5
    Supplied.Product,
    Supplied.PricePerItem,
    Suppliers.SupplierName
FROM
    (Suppliers INNER JOIN Contracts ON Suppliers.SupplierID = Contracts.SupplierID) INNER JOIN
    Supplied ON Contracts.ContractNumber = Supplied.ContractNumber
ORDER BY Supplied.PricePerItem DESC
```

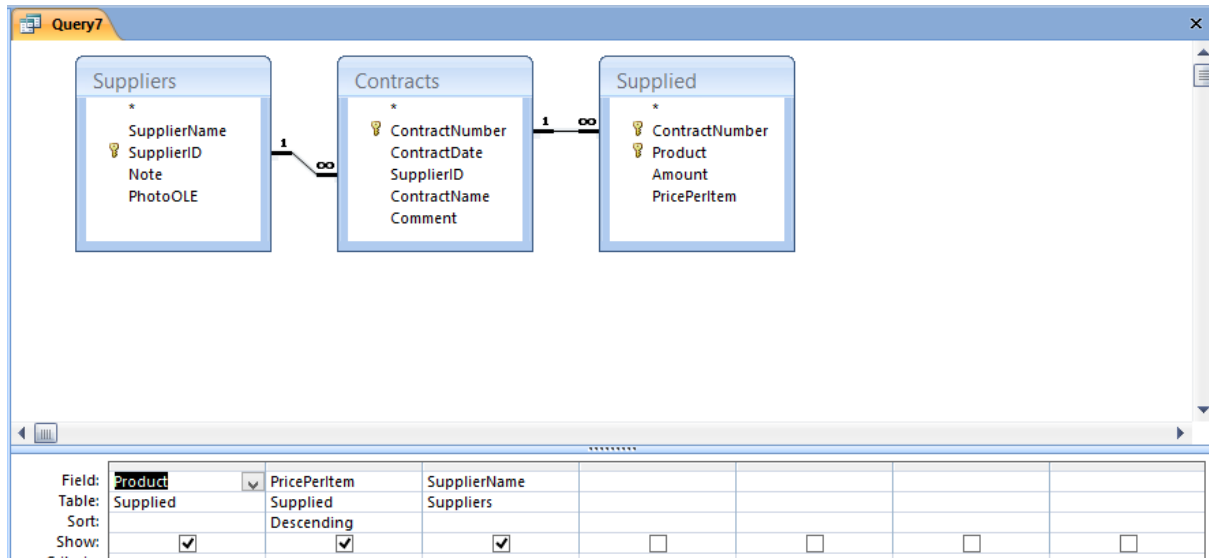


Рисунок 3.10

11. Для кожного дня 9 місяці 1999 року визначити, на яку суму були поставлені товари кожним постачальником (в результат запити включати тільки ті дні, в які були поставки товарів).

Використати наступну SQL команду або дизайнер запитів (рисунок 3.11):

```
TRANSFORM Sum(Supplied.Amount * Supplied.PricePerItem) AS [Total]
SELECT
    Suppliers.SupplierName
FROM
    Suppliers INNER JOIN
        (Contracts INNER JOIN Supplied ON Contracts.ContractNumber = Sup-
        plied.ContractNumber)
    ON Suppliers.SupplierID = Contracts.SupplierID
WHERE
    Month(Contracts.ContractDate) = 9 AND Year(Contracts.ContractDate) =
1999
GROUP BY
    Suppliers.SupplierName,
    Month(Contracts.ContractDate),
    Year(Contracts.ContractDate)
```

PIVOT Day(Contracts.ContractDate)

This is crosstab query, which defines [Suppliers.SupplierName] as row headings, Day(Contracts.ContractDate) as column headings, and [Total] as cell values.

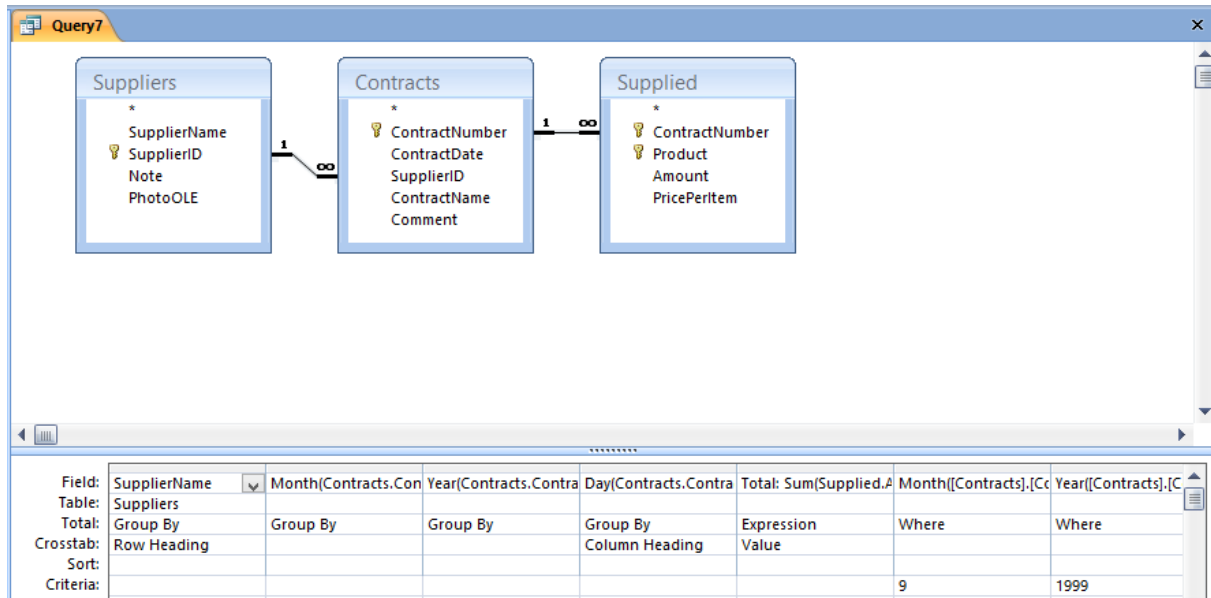


Рисунок 3.11

12. Сформувати список договорів (тільки номери), загальна кількість поставлених товарів і загальну суму за кожним договором. Для постачальників - фізичних осіб вивести прізвище, ім'я, по батькові (повністю), для постачальників - юридичних осіб - номер свідоцтва платника ПДВ.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.12):

```
SELECT
    Contracts.ContractNumber,
    Sum(Supplied.Amount) AS [TotalAmount],
    Sum(Supplied.Amount * Supplied.PricePerItem) AS [TotalPrice],
```

```

        LastName & " " & FirstName & " " & SecondName AS [SupplierFullName],
        TaxNumber
FROM
    ((Contracts LEFT JOIN IndividualEntrepreneurs ON Contracts.SupplierID
    = IndividualEntrepreneurs.SupplierID) LEFT JOIN
        LegalEntities          ON          Contracts.SupplierID          =
        LegalEntities.SupplierID) INNER JOIN
        Supplied ON Contracts.ContractNumber = Supplied.ContractNumber
GROUP BY
    Contracts.ContractNumber,
    LastName & " " & FirstName & " " & SecondName,
    TaxNumber
ORDER BY Contracts.ContractNumber

```

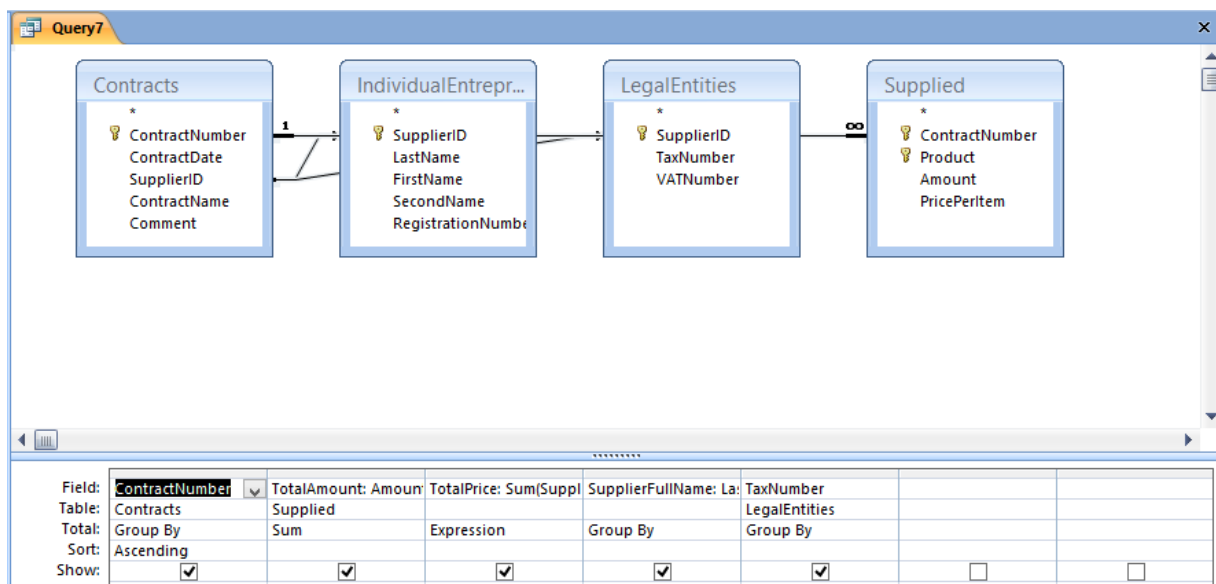


Рисунок 3.12

13. Визначити, яка кількість кожного товару було поставлено кожним постачальником.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.13):

```

TRANSFORM Sum(Supplied.Amount) AS [TotalAmount]

```

```

SELECT
    Suppliers.SupplierName
FROM
    Suppliers INNER JOIN
        (Contracts INNER JOIN Supplied ON Contracts.ContractNumber =
            Supplied.ContractNumber)
        ON Suppliers.SupplierID = Contracts.SupplierID
GROUP BY Suppliers.SupplierName
PIVOT Supplied.Product

```

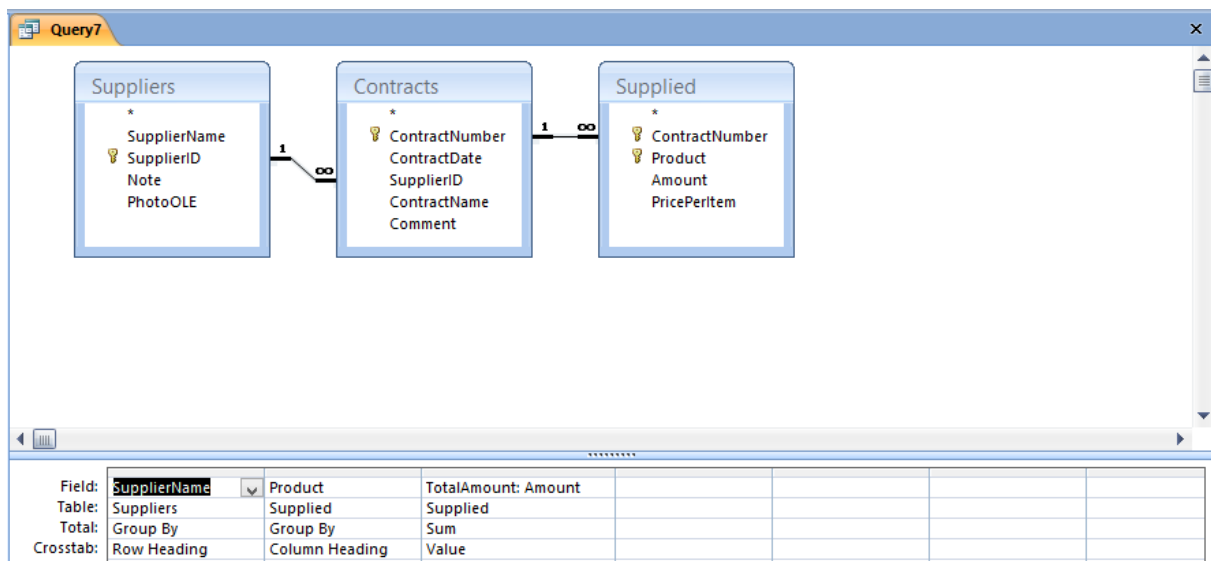


Рисунок 3.13

14. Вивести на екран список договорів (номер, дата) і загальну суму за кожним договором (розмір партії помножити на ціну за штуку і підсумувати за договором). Список повинен бути відсортований в порядку зростання загальних сум по кожному договору. Крім того, на список має бути накладено умова фільтрації, що складається у виключенні з результату запиту записів, для яких номер договору більше зазначеного значення.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.14):

```

PARAMETERS NumParam Short;
SELECT
    Contracts.ContractNumber,
    Contracts.ContractDate,
    Sum(Supplied.Amount) AS [TotalAmount],
    Sum(Supplied.Amount * Supplied.PricePerItem) AS [TotalPrice]
FROM
    Contracts INNER JOIN
    Supplied ON Contracts.ContractNumber = Supplied.ContractNumber
GROUP BY Contracts.ContractNumber, Contracts.ContractDate
HAVING Contracts.ContractNumber <= NumParam
ORDER BY Contracts.ContractNumber

```

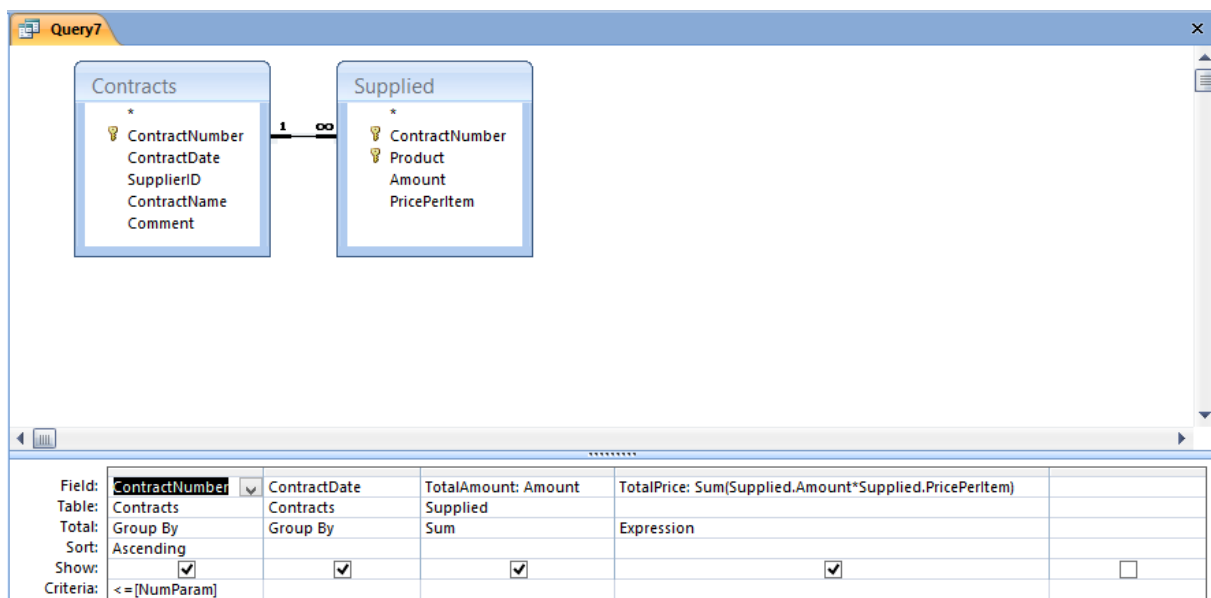


Рисунок 3.14

15. Сформувати список товарів, поставлених постачальниками 1 та 2.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.15):

```

SELECT
    Product,

```



```

        Amount,
        PricePerItem,
        (Amount * PricePerItem) AS [TotalPrice]
FROM
    Contracts, Suppliers, Supplied
WHERE
    Contracts.ContractNumber = Supplied.ContractNumber AND
    Contracts.SupplierID = Suppliers.SupplierID AND
    Suppliers.SupplierID = 1
UNION
SELECT
    Product,
    Amount,
    PricePerItem,
    (Amount * PricePerItem) AS [TotalPrice]
FROM
    Contracts, Suppliers, Supplied
WHERE
    Contracts.ContractNumber = Supplied.ContractNumber AND
    Contracts.SupplierID = Suppliers.SupplierID AND
    Suppliers.SupplierID = 2

```



Рисунок 3.15

16. Сформувати список товарів, які поставлялися більш ніж 1 раз.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.16):

Варіант 1

```
SELECT
    Supplied.Product,
    Count(Supplied.Product) AS [CountProducts]
FROM
    Supplied
GROUP BY Supplied.Product
HAVING Supplied.Product IN
    (SELECT
        Product
    FROM
        Supplied
    GROUP BY Product
    HAVING Count(Product) > 1)
```

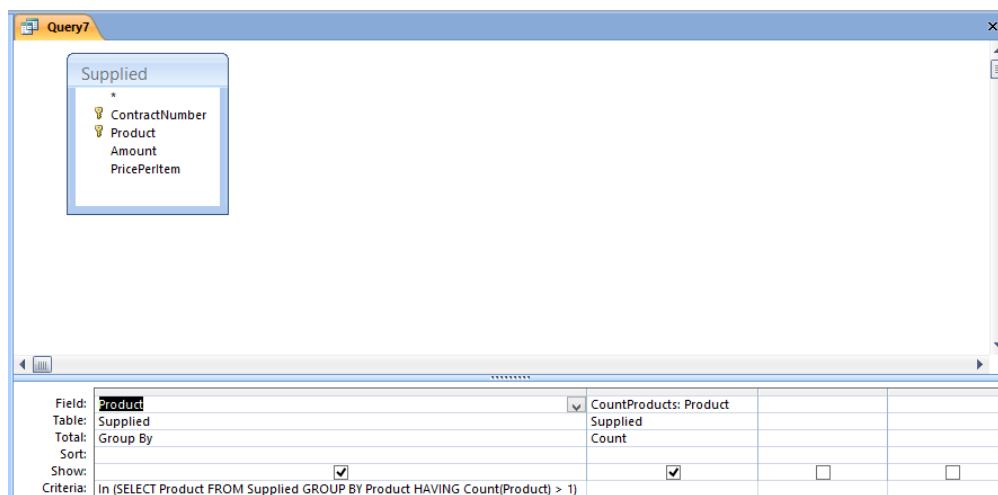


Рисунок 3.16а

Варіант 2

```
SELECT
    Supplied.Product,
    Count(Supplied.Product) AS [CountProduct]
```

```

FROM
    Supplied
GROUP BY Supplied.Product
HAVING Count(Product) > 1

```

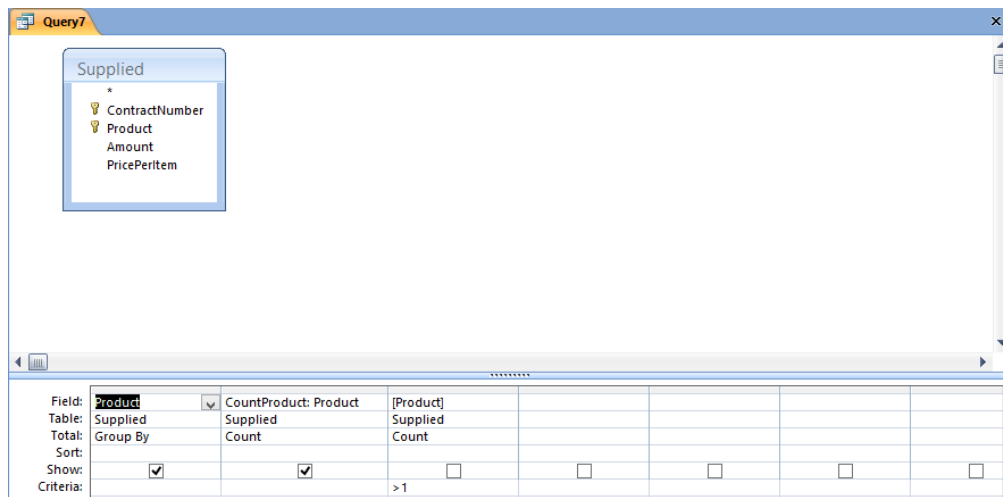


Рисунок 3.16б

II. Маніпулювання даними за допомогою команд UPDATE-SQL, DELETE-SQL.

17. Збільшити на 10 одиниць кількість кожного товару, поставленого постачальником 1.

Використати наступну SQL команду або дизайнер запитів (рисунок 3.17):

```

UPDATE
    Supplied
SET
    Supplied.Amount = Supplied.Amount + 10
WHERE
    Supplied.ContractNumber IN
        (SELECT
            ContractNumber

```

```

FROM
    Contracts
WHERE
    SupplierID = 1)

```

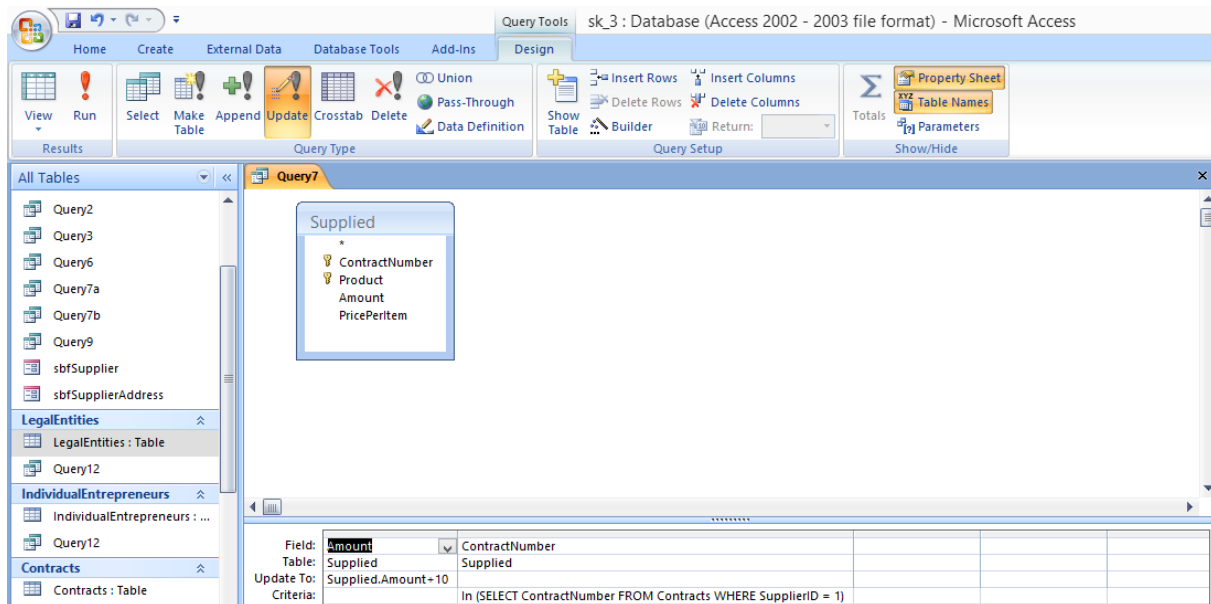


Рисунок 3.17

18. Видалити всі «порожні» договори (тобто ті договори, для яких відсутні записи про поставлені товари).

1) Створити наступний запис в таблиці «Contracts»:

ContractNumber	8
ContractDate	7/27/2002
SupplierID	3

2) Використати наступну SQL команду або дизайнер запитів (рисунок 3.18):

```

DELETE
    Contracts.ContractNumber
FROM
    Contracts

```

WHERE

Contracts.ContractNumber NOT IN (SELECT ContractNumber FROM Supplied)

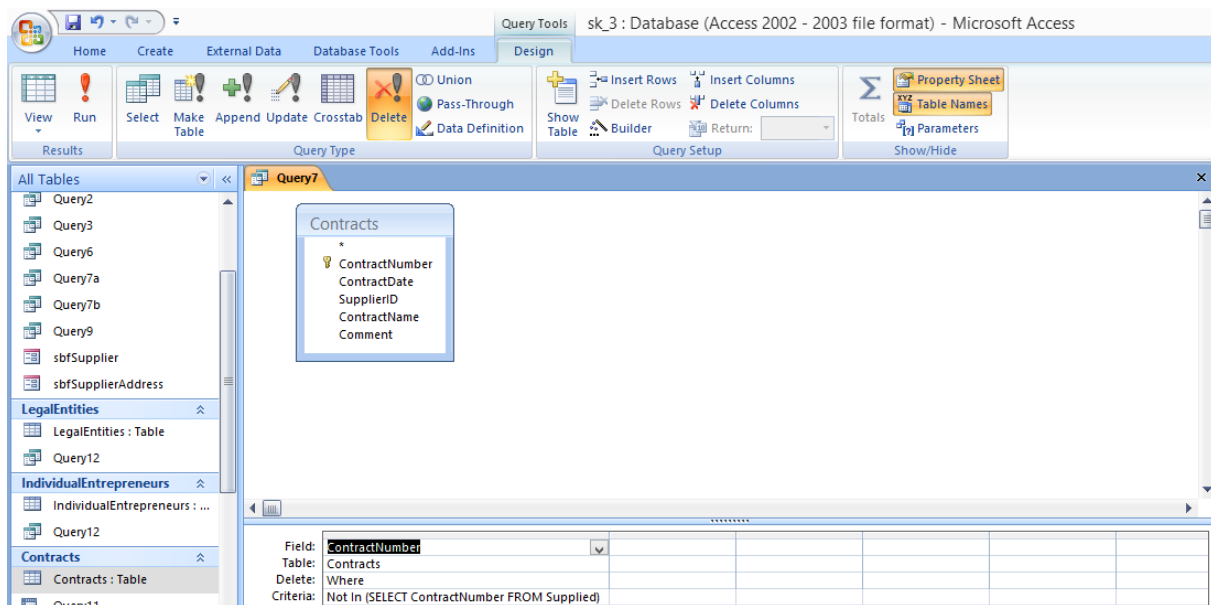


Рисунок 3.18

III. Закінчити роботу. Зберегти файл SK.mdb.

Вимоги до звіту:

- 1) коротко описати основні етапи виконання роботи;
- 2) для кожної з реалізованих команд мови SQL привести текст запиту і результат виконання запиту.

Питання за темою роботи

1. Команда SELECT-SQL. Загальна характеристика, призначення та використання.
2. Команда SELECT-SQL. Реляційні операції, які реалізуються за допомогою команди SELECT-SQL.

3. Команда SELECT-SQL. Стовпці виведення результату запиту. Використання виразів в шпальтах виведення.

4. Команда SELECT-SQL. Пропозиція FROM. Призначення і використання.

5. Команда SELECT-SQL. Пропозиція WHERE. Призначення і використання.

6. Команда SELECT-SQL. Пропозиція WHERE. FILTER-умови і їх використання.

7. Команда SELECT-SQL. Пропозиція WHERE. Формування запиту на базі декількох таблиць. JOIN-умови і їх використання.

8. Команда SELECT-SQL. З'єднання таблиць в багатотабличних запитах. Види з'єднань - INNER JOIN, LEFT JOIN, RIGHT JOIN і їх особливості.

9. Команда SELECT-SQL. Перехресні запити і їх особливості.

10. Команда SELECT-SQL. Аргумент DISTINCT. Призначення і використання.

11. Команда SELECT-SQL. Аргумент TOP. Призначення і використання.

12. Команда SELECT-SQL. Булеві оператори AND, OR, NOT та їх використання

13. Команда SELECT-SQL. Спеціальні оператори IN, BETWEEN. Призначення і використання.

14. Команда SELECT-SQL. Агрегуючі функції COUNT, SUM, AVG, MAX, MIN. Призначення і використання.

15. Команда SELECT-SQL. Пропозиція GROUP BY. Призначення і використання.

16. Команда SELECT-SQL. Пропозиція ORDER BY. Призначення і використання.

17. Команда SELECT-SQL. Пропозиція HAVING. Призначення і використання.

18. Команда SELECT-SQL. Підзапити. Призначення і використання.

19. Команда SELECT-SQL. Підзапити. Види підзапитів і їх особливості.

20. Команда SELECT-SQL. Параметри і їх використання при побудові запитів.

21. Команда INSERT-SQL. Загальна характеристика, призначення та використання.

22. Команда DELETE-SQL. Загальна характеристика, призначення та використання.

23. Команда UPDATE-SQL. Загальна характеристика, призначення та використання.

24. Як реалізувати запит 15 без використання операції UNION?