

KTest

更に上のクオリティ 更に上のサービス



問題集

<http://www.ktest.jp>

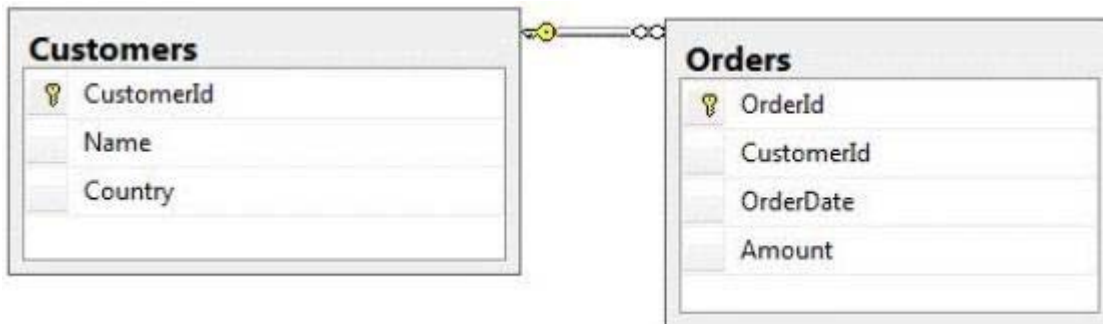
1年で無料進級することに提供する

Exam : **70-461J**

Title : Querying Microsoft SQL
Server 2012

Version : DEMO

1. あなたが ContosoDb 付きの Microsoft SQL Server 2012 のデータベースを管理します。展示に示すように、テーブルが定義されています。(図表ボタンをクリックします。)



あなたは、顧客の注文テーブルから行を表示する必要があると、次の XML 形式で 1 に設定 CustomerId の値を持つ行:

```
<row OrderId="1" OrderDate="2000-01-01T00:00:00" Amount="3400.00"
Name="Customer A" Country="Australia" />
```

```
<row OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00"
Name="Customer A" Country="Australia" />
```

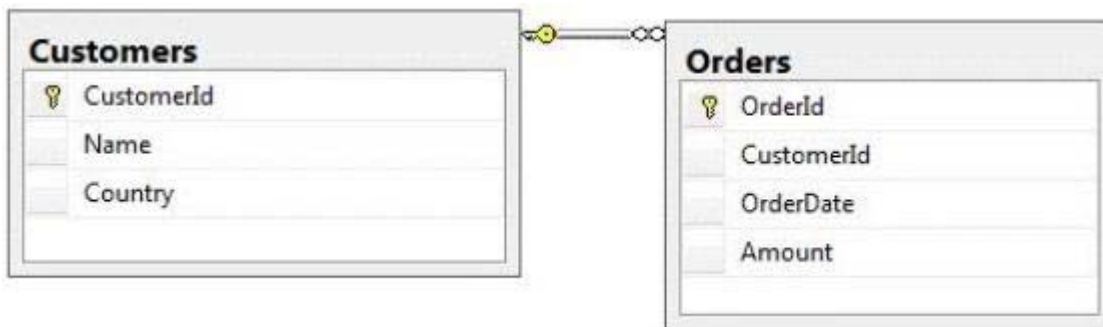
あなたはどちらの Transact-SQL クエリを使うべきでしょうか?

- A. SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.
CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW
- B. SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.
CustomerId
WHERE Customers.CustomerId = 1
FOR XML RAW, ELEMENTS
- C. SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.
CustomerId
WHERE Customers.CustomerId = 1
FOR XML AUTO
- D. SELECT OrderId, OrderDate, Amount, Name, Country
FROM Orders INNER JOIN Customers ON Orders.CustomerId - Customers.
CustomerId
WHERE Customers.CustomerId= 1
FOR XML AUTO, ELEMENTS
- E. SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
WHERE Customers.CustomerId- 1
FOR XML AUTO
- F. SELECT Name, Country, OrderId, OrderDate, Amount
FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId

WHERE Customers.CustomerId= 1
 FOR XML AUTO, ELEMENTS
 G. SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount
 FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
 WHERE Customers.CustomerId= 1
 FOR XML PATH ('Customers')
 H. SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId,
 OrderDate, Amount
 FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId
 WHERE Customers.CustomerId= 1
 FOR XML PATH ('Customers')

Answer: A

2. あなたが ContosoDb 付きの Microsoft SQL Server 2012 のデータベースを管理します。展示に示すように、テーブルが定義されています。(図表ボタンをクリックします。)



あなたは、顧客の注文テーブルから行を表示する必要があると、次の XML 形式で 1 に CustomerIdvalue セットを持つ行。

```

  <Orders OrderId="1" OrderDate="2000-01-01T00:00:00" Amount="3400.00">
    <Customers Name="Customer A" Country="Australia" />
  </Orders>
  <Orders OrderId="2" OrderDate="2001-01-01T00:00:00" Amount="4300.00">
    <Customers Name="Customer A" Country="Australia" />
  </Orders>
  
```

あなたはどちらの Transact-SQL クエリを使うべきでしょうか？

- A. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW
- B. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers=CustomerId = 1 FOR XML RAW, ELEMENTS
- C. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO
- D. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS
- E. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON

Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1

FOR XML AUTO

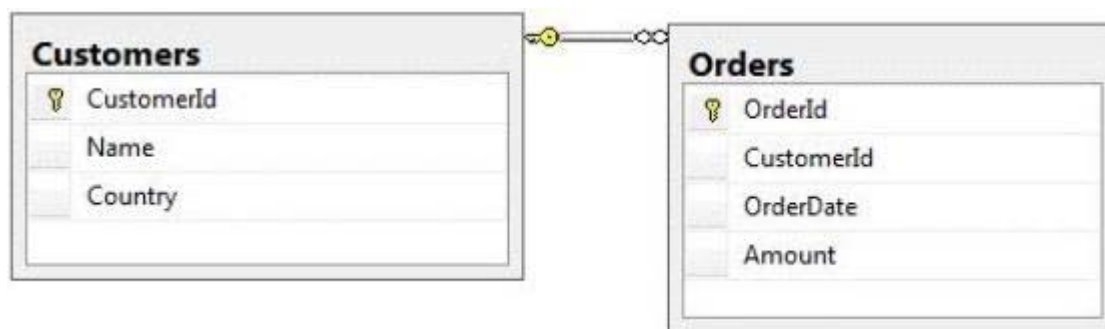
F. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS

G. SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers')

H. SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers')

Answer: C

3. あなたが ContosoDb 付きの Microsoft SQL Server 2012 のデータベースを管理します。展示に示すように、テーブルが定義されています。(図表ボタンをクリックします。)



あなたは、顧客の注文テーブルから行を表示する必要があると、次の XML 形式で 1 に設定 CustomerId の値を持つ行。

```
<CUSTOMERS Name="Customer A" Country="Australia">
  <ORDERS OrderID="1" OrderDate="2001-01-01" Amount="3400.00" />
  <ORDERS OrderID="2" OrderDate="2002-01-01" Amount="4300.00" />
</CUSTOMERS>
```

あなたはどちらの Transact-SQL クエリを使うべきでしょうか?

A. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW

B. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW, ELEMENTS

C. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers.CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO

D. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId - Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS

E. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO

F. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO,

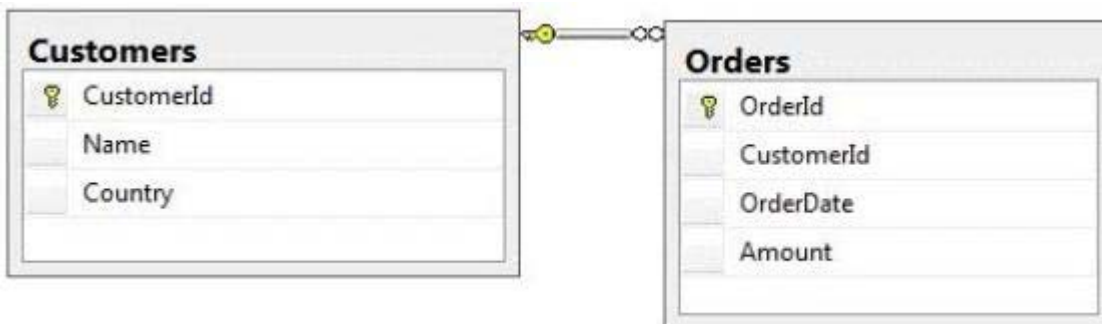
ELEMENTS

G. SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers')

H. SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers')

Answer: E

4. あなたが ContosoDb 付きの Microsoft SQL Server 2012 のデータベースを管理します。展示に示すように、テーブルが定義されています。(図表ボタンをクリックします。)



あなたは、顧客の注文テーブルから行を表示する必要があると、次の XML 形式で 1 に設定 CustomerId の値を持つ行。

```
<Orders>
  <OrderId>1</OrderId>
  <OrderDate>2000-01-01T00:00:00</OrderDate>
  <Amount>3400.00</Amount>
  <Customers>
    <Name>Customer A</Name>
    <Country>Australia</Country>
  </Customers>
</Orders>
<Orders>
  <OrderId>2</OrderId>
  <OrderDate>2001-01-01T00:00:00</OrderDate>
  <Amount>4300.00</Amount>
  <Customers>
    <Name>Customer A</Name>
    <Country>Australia</Country>
  </Customers>
</Orders>
```

あなたはどちらの Transact-SQL クエリを使うべきでしょうか？

- A. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW
 B. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML RAW,

ELEMENTS

C. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId = 1 FOR XML AUTO

D. SELECT OrderId, OrderDate, Amount, Name, Country FROM Orders INNER JOIN Customers ON Orders.CustomerId = Customers. CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS

E. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId- 1 FOR XML AUTO

F. SELECT Name, Country, OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML AUTO, ELEMENTS

G. SELECT Name AS '@Name', Country AS '@Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers')

H. SELECT Name AS 'Customers/Name', Country AS 'Customers/Country', OrderId, OrderDate, Amount FROM Orders INNER JOIN Customers ON Orders.CustomerId= Customers.CustomerId WHERE Customers.CustomerId= 1 FOR XML PATH ('Customers')

Answer: D

5. あなたは、アプリケーションをサポートする Microsoft SQL Server 2012 のサーバーのデータベースを開発します。

アプリケーションは、次の定義を持つ表が含まれている：

```
CREATE TABLE Inventory (  
ItemID int NOT NULL PRIMARY KEY,  
ItemsInStore int NOT NULL,  
ItemsInWarehouse int NOT NULL)
```

あなたは、行ごとに ItemsInStore と ItemsInWarehouse 値の総和を返す計算列を作成する必要があります。新しい列が重く照会することが期待、そしてあなたは、インデックス列にできるようにする必要があります。

あなたはどちらの Transact-SQL ステートメントを使用する必要がありますか？

A. ALTER TABLE Inventory ADD TotalItems AS ItemsInStore + ItemsInWarehouse

B. ALTER TABLE Inventory ADD TotalItems AS ItemsInStore + ItemsInWarehouse PERSISTED

C. ALTER TABLE Inventory ADD TotalItems AS SUM(ItemsInStore, ItemsInWarehouse) PERSISTED

D. ALTER TABLE Inventory ADD TotalItems AS SUM(ItemsInStore, ItemsInWarehouse)

Answer: B

6. あなたは、Customers という名前のテーブルを含む Microsoft SQL Server 2012 のデータベースを開発します。

Customers テーブルには、次の定義が含まれている：

```

CREATE TABLE [dbo].[Customers] (
  [CustomerId] [bigint] NOT NULL,
  [MobileNumber] [nvarchar] (25) NOT NULL,
  [HomeNumber] [nvarchar] (25) NULL,
  [Name] [nvarchar] (50) NOT NULL,
  [Country] [nvarchar] (25) NOT NULL,
  CONSTRAINT [PK_Customers] PRIMARY KEY CLUSTERED
(
  [CustomerId] ASC
) ON [PRIMARY]
) ON [PRIMARY]

```

あなたは、どちら MobileNumber または HomeNumber 列が更新されている場合にのみ監査レコードを作成する必要があります。

あなたはどちらの Transact-SQL クエリを使うべきでしょうか？

A. CREATE TRIGGER TrgPhoneNumberChange

ON Customers FOR UPDATE

AS

IF COLUMNS_UPDATED (HomeNumber, MobileNumber)

-- Create Audit Records

B. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF EXISTS(SELECT HomeNumber FROM inserted) OR EXISTS (SELECT MobileNumber FROM inserted)

-- Create Audit Records

C. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF COLUMNS_CHANGED (HomeNumber, MobileNumber)

-- Create Audit Records

D. CREATE TRIGGER TrgPhoneNumberChange ON Customers FOR UPDATE AS IF UPDATE (HomeNumber) OR UPDATE (MobileNumber)

-- Create Audit Records

Answer: D

7. あなたが SavingAccounts と LoanAccounts という名前の二つのテーブルを持つ Microsoft SQL Server 2012 のデータベースを開発しています。両方のテーブルのデータ型は nvarchar データ型の

AccountNumber をという名前の列があります。あなたの transactionID の AccountNumber を、金額、および TransactionDate という名前の列があるトランザクションの名前の 3 番目のテーブルを使用しています。あなたは、複数のレコードがトランザクションテーブルに挿入されたとき、または SavingAccounts LoanAccounts で有効 AccountNumber をを持っている唯一のレコードが挿入されていることを確認する必要があります。

あなたはどちらの Transact-SQL ステートメントを使用する必要がありますか？

A. CREATE TRIGGER TrgValidateAccountNumber ON Transactions INSTEAD OF INSERT AS BEGIN

INSERT INTO Transactions

SELECT TransactionID,AccountNumber,Amount,TransactionDate FROM inserted

WHERE AccountNumber IN

(SELECT AccountNumber FROM LoanAccounts

UNION SELECT AccountNumber FROM SavingAccounts))

END

```
B. CREATE TRIGGER TrgValidateAccountNumber ON Transactions FOR INSERT AS BEGIN
INSERT INTO Transactions
SELECT TransactionID,AccountNumber,Amount,TransactionDate FROM inserted
WHERE AccountNumber IN
(SELECT AccountNumber FROM LoanAccounts
UNION SELECT AccountNumber FROM SavingAccounts))
```

END

```
C. CREATE TRIGGER TrgValidateAccountNumber ON Transactions
INSTEAD OF INSERT
```

```
AS
```

```
BEGIN
```

```
IF EXISTS (
SELECT AccountNumber FROM inserted EXCEPT
(SELECT AccountNumber FROM LoanAccounts
UNION SELECT AccountNumber FROM SavingAccounts))
```

```
BEGIN
```

```
ROLLBACK TRAN
```

```
END
```

END

```
D. CREATE TRIGGER TrgValidateAccountNumber ON Transactions FOR INSERT AS BEGIN
IF EXISTS (
SELECT AccountNumber FROM inserted EXCEPT
(SELECT AccountNumber FROM LoanAccounts
UNION SELECT AccountNumber FROM SavingAccounts))
```

```
BEGIN
```

```
ROLLBACK TRAN
```

```
END
```

END

Answer: A

8. あなたは、Microsoft SQL Server 2012 のデータベースを開発します。

あなたは、次のタスクを実行するビューを作成する：

- 50 万レコードの各まで含む 8 のテーブルを結合します。

- 5 フィールドの集計を行います。

ビューは頻繁にいくつかのレポートで使用されます。 、レポートの性能を向上させる必要があります。

あなたはどうすればいいのでしょうか？

A. テーブル値関数にビューに変換します。

B. 共通テーブル式 (CTE) にビューに変換します。

C. インデックス付きビューにビューに変換します。

D. ストアドプロシージャにビューを変換し、ストアドプロシージャからの一時テーブルに結果を取得します。

Answer: C

9. あなたは、Microsoft SQL Server 2012 のデータベースのデータベース開発者であります。データベースには、次の定義を持つ Customers という名前のテーブルが含まれている：

```
CREATE TABLE Customer
(CustomerID INT NOT NULL PRIMARY KEY,
 CustomerName VARCHAR(255) NOT NULL,
 CustomerAddress VARCHAR(1000) NOT NULL)
```

You are designing a new table named Orders that has the following definition:

```
CREATE TABLE Orders
(OrderID INT NOT NULL PRIMARY KEY,
 CustomerID INT NOT NULL,
 OrderDescription VARCHAR(2000))
```

あなたは、受注テーブルの CustomerID 列は、Customer テーブルの CustomerID 列に存在する値のみが含まれていることを確認する必要があります。

あなたはどちらの Transact-SQL ステートメントを使用する必要がありますか？

A. ALTER TABLE Orders

```
ADD CONSTRAINT FX_Orders_CustomerID FOREIGN KEY (CustomerId) REFERENCES
Customer (CustomerId)
```

B. ALTER TABLE Customer

```
ADD CONSTRAINT FK_Customer_CustomerID FOREIGN KEY {CustomerId} REFERENCES
Orders (CustomerId)
```

C. ALTER TABLE Orders

```
ADD CONSTRAINT CK_Crders_CustomerID
CHECK (CustomerId IN (SELECT CustomerId FROM Customer))
```

D. ALTER TABLE Customer

```
ADD OrderId INT NOT NULL;
```

ALTER TABLE Customer

```
ADD CONSTRAINT FK_Customer_OrderID FOREIGN KEY (CrderID) REFERENCES Orders
(CrderID);
```

E. ALTER TABLE Orders

```
ADD CONSTRAINT PK Orders CustomerId PRIMARY KEY (CustomerId)
```

Answer: A

10. あなたが歯科医、精神科医、および医師のためのデータが含まれている三つのテーブルを持っている。あなたはそれらの電子メールアドレスと電話番号をルックアップするために使用されているビューを作成します。

ビューには、次の定義が含まれている：

```
Create view apt.vwProviderList
(Specialty, CompanyID, CompanyNumber, LastName,
 FirstName, BusinessName, Email, Phone)
as

SELECT 'Dentist' as Specialty
, DentistID
, DentistNumber
, DentistLastName
, DentistFirstName
, DentistBusinessName
, Email
, Phone
FROM apt.Dentist
UNION ALL
SELECT 'Psychiatrist' as Specialty
, PsychiatristID
, PsychiatristNumber
, PsychiatristLastName
, PsychiatristFirstName
, PsychiatristBusinessName
, Email
, Phone
SELECT 'Physician' as Specialty
, PhysicianID
, PhysicianNumber
, PhysicianLastName
, PhysicianFirstName
, PhysicianBusinessName
, Email
, Phone
FROM apt.Physician
GO
```

あなたは、ユーザーがこのビューを使用して電話番号やメールアドレスだけを更新できるようにする必要があります。

あなたはどうすればいいのでしょうか？

- A. ビューを変更します。各 SELECT 文とともに VIEWS クエリヒントを EXPAND 使用します。
- B. ビューに INSTEAD OF UPDATE トリガを作成します。
- C. ビューを削除します。 SCHEMABINDING 句を使用してビューを再作成し、ビューにインデックスを作成します。
- D. ビューの AFTER UPDATE トリガーを作成します。

Answer: B