

```

-----TRIGGER-----
----Viết trigger khi xóa một OrderId thì xóa luôn các thông tin của Order đó trong bảng OrderItem.
----Nếu có Foreign Key Constraint xảy ra không cho xóa thì hãy xóa Foreign Key Constraint đó đi rồi thực thi.
Create trigger [dbo].[Trigger_OrderItemDelete]
on [dbo].[Order]
For delete
as

declare @DeletedOrderId int
select @DeletedOrderId = Id From deleted

Delete From [Order] Where Id = @DeletedOrderId
Delete From OrderItem Where OrderId = @DeletedOrderId

Print 'Order, OrderItem with OrderId = '+LTRIM(RTRIM(@DeletedOrderId))+ ' have been deleted'

```

Commands completed successfully.

Completion time: 2022-06-16T20:03:17.0738536+07:00

100 %

Query executed successfully.

DESKTOP-DJT45HV\SQLEXPRESS ... | DESKTOP-DJT45HV\My Com... | Northwind | 00:00:00 | 0 rows

```
--Delete foregin key--
```

```
Alter table OrderItem Drop constraint FK_ORDERITE_REFERENCE_ORDER
```

```
Delete From [Order] where Id = 1
```

```
Delete From [Order] where Id = 2
```

```
Select * From [Order]
```

```
Order by Id
```

```
Select * From [OrderItem]
```

```
order by OrderId
```

	Id	OrderDate	OrderNumber	CustomerId	TotalAmount
1	3	2012-07-08 00:00:00.000	542380	34	1813.00
2	4	2012-07-08 00:00:00.000	542381	84	670.80
3	5	2012-07-09 00:00:00.000	542382	76	3730.00
4	6	2012-07-10 00:00:00.000	542383	34	1444.80
5	7	2012-07-11 00:00:00.000	542384	14	625.20
6	8	2012-07-12 00:00:00.000	542385	68	2490.50
7	9	2012-07-15 00:00:00.000	542386	88	517.80
8	10	2012-07-16 00:00:00.000	542387	35	1119.90

	Id	Orderid	ProductId	UnitPrice	Quantity
1	6	3	41	7.70	10
2	7	3	51	42.40	35
3	8	3	65	16.80	15
4	9	4	22	16.80	6
5	1...	4	57	15.60	15
6	1...	4	65	16.80	20
7	1...	5	20	64.80	40
8	1...	5	33	2.00	25
9	1...	5	60	27.20	40
10	1...	6	31	10.00	20
11	1...	6	39	14.40	42
12	1...	6	49	16.00	40
13	1...	7	24	3.60	15

```
--Viết trigger khi xóa hóa đơn của khách hàng Id = 1 thì báo lỗi không cho xóa sau đó ROLL BACK lại. --
--Lưu ý: Đưa trigger này lên làm Trigger đầu tiên thực thi xóa dữ liệu trên bảng Order--

Create Trigger [dbo].[Trigger_CustomerIdDelete]
on [dbo].[Order]
for delete
as
    declare @DeleteCustomerId int
    select @DeleteCustomerId = CustomerId From deleted

    If(@DeleteCustomerId=1)
    Begin
        Raiserror('CustomerId = 1 cant be deleted',16,1);
        Rollback transaction
    End

    Delete From [Order] Where CustomerId=1
```

100 %

Messages

Commands completed successfully.

Completion time: 2022-06-16T20:06:49.3551899+07:00

100 %

```
--Viết trigger không cho phép cập nhật Phone là NULL hay trong Phone có chữ cái ở bảng Supplier. Nếu có thì báo lỗi và ROLL BACK lại--
Create trigger [dbo].[SupplierPhoneUpdate]
on [dbo].[Supplier]
for update
as
    declare @UpdatePhone nvarchar(30)
    if update(Phone)
    Begin
        Select @UpdatePhone = Phone From inserted
        If @UpdatePhone=null or @UpdatePhone = 'NULL'
        Begin
            Raiserror ('Phone must be a phone number',16,1);
            Rollback transaction
        End
    End
End

Update Supplier Set Phone = 'NULL' Where Id = 1
```

100 %

Messages

Commands completed successfully.

Completion time: 2022-06-16T20:07:31.6021323+07:00

100 %

Query executed successfully. | DESKTOP-DJT45HV\SQLEXPRESS ... | DESKTOP-DJT45HV\My Com... | Northwind | 00:00:00 | 0 rows

```
--CURSOR--

Create Function dbo.ufn_ListCompanyByCountry (@CountryDescr nvarchar(MAX))
Returns nvarchar(MAX)
As
Begin
    Declare @CompanyList nvarchar(MAX) = 'Companies in ' + @CountryDescr + ' are: ';
    Declare @Id int;
    Declare @CompanyName nvarchar(MAX);

    Declare CompanyCursor Cursor READ_ONLY
    For
        Select Id, CompanyName
        From Supplier
        Where Lower(Country) = RTRIM(LTRIM(LOWER(@CountryDescr)))

    Open CompanyCursor

    Fetch next From CompanyCursor into @Id, @CompanyName

    While @@FETCH_STATUS=0
    Begin
        Set @CompanyList = @CompanyList + @CompanyName + '(ID: ' + LTRIM(STR(@Id)) + ') ' + '; ';
        FETCH NEXT FROM CompanyCursor into @Id, @CompanyName
    End

    Close CompanyCursor
    Deallocate CompanyCursor

    Return @CompanyList
End
```

Commands completed successfully.

Completion time: 2022-06-16T20:10:30.4579104+07:00

100 %

Query executed successfully. | DESKTOP-DJT45HV\SQLEXPRESS ... | DESKTOP-DJT45HV\My Com... | Northwind | 00:00:00 | 0 rows

```
--S TRANSACTION--

Begin try
    Begin Transaction UpdateQuantityTrans

        Set nocount on;

        Declare @NumOfUpdateRecords int =0;
        Declare @DFactor int;
        Set @DFactor=2;

        Update oi Set oi.Quantity = oi.Quantity / @DFactor
        From OrderItem oi
        inner join [Order] o on oi.OrderId = o.Id inner join Customer as c on o.CustomerId = c.Id
        Where c.Country like '%USA%'

        Set @NumOfUpdateRecords = @@ROWCOUNT
        Print 'Update successful. There are ' + LTRIM(RTRIM(@NumOfUpdateRecords)) + ' rows in OrderItem table';

        COMMIT TRANSACTION UpdateQuantityTrans
    End try
Begin catch
    Rollback tran UpdateQuantityTran
    Print 'Update failed. See detail: ';
    Print ERROR_MESSAGE();
End catch
```

100 %

Messages

Update successful. There are 352 rows in OrderItem table

Completion time: 2022-06-16T20:11:27.5668256+07:00

100 %

Query executed successfully. | DESKTOP-DJT45HV\SQLEXPRESS ... | DESKTOP-DJT45HV\My Com... | Northwind | 00:00:00 | 0 rows

```
--TEMP TABLE--

Begin try
    Begin transaction CompareTwoCountriesTrans

        Set nocount on
        declare @Country1 nvarchar(MAX)
        declare @Country2 nvarchar(MAX)

        set @Country1 = 'USA';
        set @Country2 = 'UK';

        Create Table #CountryInfo1 --Create a physical table
        (
            Id int,
            ProductName nvarchar(50),
            SupplierId int
        )

        Declare @CountryInfo2 Table
        (
            Id int,
            ProductName nvarchar(50),
            SupplierId int
        )

        Insert into #CountryInfo1
        Select p.Id, p.ProductName, p.SupplierId
        From Product p inner join Supplier s on p.SupplierId = s.Id
        Where Lower(s.Country) = LTRIM(RTRIM(LOWER(@Country1)))

        Insert into @CountryInfo2
        Select p.Id, p.ProductName, p.SupplierId
```

```

Insert into @CountryInfo2
Select p.Id, p.ProductName, p.SupplierId
From Product p inner join Supplier s on p.SupplierId = s.Id
Where Lower(s.Country) = LTRIM(RTRIM(LOWER(@Country2)))

Declare @NumOfProduct1 int
Set @NumOfProduct1 = (Select Count(*) From #CountryInfo1);
Declare @NumOfProduct2 int
Set @NumOfProduct2 = (Select Count(*) From @CountryInfo2);

Print 'A quantity of product came from ' + LTRIM(@Country1) + ': ' + Convert(nvarchar,@NumOfProduct1)
Print 'A quantity of product came from ' + LTRIM(@Country2) + ': ' + Convert(nvarchar,@NumOfProduct2)

Print
Case
    When @NumOfProduct1 = @NumOfProduct2
        Then 'A quantity of product came from ' + LTRIM(STR(Convert(nvarchar,@Country1))) + ' and ' + LTRIM(STR(Convert(nvarchar,@Country2))) + ' are equal'
    When @NumOfProduct1 > @NumOfProduct2
        Then 'A quantity of product came from ' + LTRIM(@Country1) + ' is bigger than ' + LTRIM(@Country2)
    Else 'A quantity of product came from ' + LTRIM(@Country2) + ' is bigger than ' + LTRIM(@Country1)
End

Drop table #CountryInfo1
Commit transaction CompareTwoCountriesTrans
End Try
Begin Catch
    Rollback tran CompareTwoCountriesTrans
    Print 'There are some problems. See detail'
    Print ERROR_MESSAGE();
End catch

```

```

A quantity of product came from USA: 12
A quantity of product came from UK: 7
A quantity of product came from USA is bigger than UK

Completion time: 2022-06-16T20:12:41.4940651+07:00

```

100 %

✓ Query executed successfully.

DESKTOP-DJT45HV\SQLEXPRESS ...

DESKTOP-DJT45HV\My Com...

Northwind

00:00:00 0 rows