Data Training
Email: marcela.filip@datatraining.ro
Telefon: +40 721 61 27 85
Website: www.datatraining.ro

PROIECT DE BUSINESS SI IMPLEMENTARE in Data Warehouse

I. Cerinta de business:

Se doreste un raport la nivel de cu vanzarea la nivel de tara client astfel:

- O coloana country
- O coloana cu vanzarile pe primul semestru dintr-un an denumita Sales Previous Sem
- O coloana cu vanzarile pe al doilea semestru dintr-un an denumita Sales Current Sem

Business-ul doreste stocarea informatiilor intr-o tabela permanenta:

II. Rolul analistului

Analistul discuta cu business-ul si rezulta urmatorul tabel permanent si metoda de calcul:

Nume coloana	**Data Type**	**Coloana din tabela sursa sau formula de calcul**
Country	Nvarchar(100)	Este coloana companyname din tabela Sales.Customers
Sales Previous Sem	Decimal(19,4)	Sum(case when o.orderdate>=start date pentru previous sem and o.orderdate<= end date pentru previous sem then sum(sales.salesfunction) else 0 end)
Sales Current Sem	Decimal(19,4)	Sum(case when o.orderdate>=start date pentru current sem and o.orderdate<= end date pentru current sem then sum(sales.salesfunction) else 0 end)

In plus, de fiecare data cand este rulat raportul se va goli mai intai tabela pentru a se insera noile date.

III. Rolul developerului:

1. Creaza tabela permanenta in baza de date:

```
create table Sales.Report_Sem
([Country] nvarchar(100),
[Sales Previous Sem] decimal(19,4),
[Sales Current Sem] decimal(19,4)
)

)
```

Data Training
Email: marcela.filip@datatraining.ro
Telefon: +40 721 61 27 85
Website: www.datatraining.ro

2. Pregateste select-ul cu care extrage datele conform cerintei de business (fara sa insereze in tabela) si isi ia ca referinta anul 2007 pe care il imparte in Previous Sem (1 ian 2007 – 30 iun 2007) si Current Sem (1 iul 2007 – 31 dec 2007):

```
select
    c.country as Country,
    sum(case when o.orderdate>='2007-01-01' and o.orderdate<='2007-06-30' then
od.qty*od.unitprice else 0 end) as [Sales Previous Sem],
    sum(case when o.orderdate>='2007-07-01' and o.orderdate<='2007-12-31' then
od.qty*od.unitprice else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>='2007-01-01' and o.orderdate<='2007-12-31'
group by
    c.country
```

3. Insereaza datele pentru prima data in tabela ca sa testeze ca pe langa select, functioneaza si insertul.

```
insert into Sales.Report_Sem (Country,[Sales Previous Sem],[Sales Current Sem])
select
    c.country as Country,
    sum(case when o.orderdate>='2007-01-01' and o.orderdate<='2007-06-30' then
od.qty*od.unitprice else 0 end) as [Sales Previous Sem],
    sum(case when o.orderdate>='2007-07-01' and o.orderdate<='2007-12-31' then
od.qty*od.unitprice else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>='2007-01-01' and o.orderdate<='2007-12-31'
group by
    c.country
```

4. Verifica sa vada daca sunt date in tabela:

```
select * from  Sales.Report_Sem
```

Results | Messages

	Country	Sales Previous Sem	Sales Current Sem
1	Finland	9473.2000	4807.4500
2	USA	41327.5900	79710.1100
3	Italy	3232.6000	5216.3500
4	Brazil	15213.2000	29337.3100
5	Germany	57875.4000	66294.9300
6	Switzerland	6387.8000	12314.7000
7	Mexico	12173.6500	2667.0000
8	Sweden	11699.4000	16325.3000
9	Argentina	1098.1000	718.5000
10	Austria	25911.8000	37240.1800

(sunt 21 de randuri)

5. Parametrizeaza select-ul astfel incat sa rezulte 2 perioade si anul intreg si are in minte ca raportul se va rula tot timpul cu 1 parametru: start date previous si restul perioadelor se vor calcula pornind de la acest parametru.

```
declare @startdate_prev datetime,
          @enddate_prev datetime,
          @startdate_curr datetime,
          @enddate_curr datetime

set @startdate_prev = '2007-01-01'
set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
set @enddate_curr=eomonth(@startdate_curr,5)

select @startdate_prev, @enddate_prev, @startdate_curr, @enddate_curr
```

	(No column name)	(No column name)	(No column name)	(No column name)
1	2007-01-01 00:00:00.000	2007-06-30 00:00:00.000	2007-07-01 00:00:00.000	2007-12-31 00:00:00.000

Acum goleste tabela cu scopul de a testa selectul dupa adaugarea variabilor:

```
delete from Sales.Report_Sem
```

Acum parametrizeaza select-ul:

```
declare @startdate_prev datetime,
          @enddate_prev datetime,
          @startdate_curr datetime,
          @enddate_curr datetime

set @startdate_prev = '2007-01-01'
set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
```

```
set @enddate_curr=eomonth(@startdate_curr,5)

insert into Sales.Report_Sem (Country,[Sales Previous Sem],[Sales Current Sem])
select
	c.country as Country,
	sum(case when o.orderdate>=@startdate_prev and o.orderdate<=@enddate_prev then
od.qty*od.unitprice else 0 end) as [Sales Previous Sem],
	sum(case when o.orderdate>=@startdate_curr and o.orderdate<=@enddate_curr then
od.qty*od.unitprice else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>=@startdate_prev and o.orderdate<=@enddate_curr
group by
	c.country
```

Acum, developerul stie ca de fiecare data cand se va rula raportul, mai intai se va goli tabela, deci va adauga si delete-ul in script.

```
delete from Sales.Report_Sem

declare @startdate_prev datetime,
		@enddate_prev datetime,
		@startdate_curr datetime,
		@enddate_curr datetime

set @startdate_prev = '2007-01-01'
set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
set @enddate_curr=eomonth(@startdate_curr,5)

insert into Sales.Report_Sem (Country,[Sales Previous Sem],[Sales Current Sem])
select
	c.country as Country,
	sum(case when o.orderdate>=@startdate_prev and o.orderdate<=@enddate_prev then
od.qty*od.unitprice else 0 end) as [Sales Previous Sem],
	sum(case when o.orderdate>=@startdate_curr and o.orderdate<=@enddate_curr then
od.qty*od.unitprice else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>=@startdate_prev and o.orderdate<=@enddate_curr
group by
	c.country
```

6. Transforma select-ul in procedura stocata cu scopul de a incapsula intreg codul si de a fi rulat totul compact.
   - Parametrul @startdate_prev nu va mai fi in zona de declare, ci devine singurul paramtru al procedurii)

```
create procedure Sales.Sp_Report_Sem
(@startdate_prev datetime)
as
begin
set nocount on;
delete from Sales.Report_Sem

declare
		@enddate_prev datetime,
		@startdate_curr datetime,
		@enddate_curr datetime


set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
set @enddate_curr=eomonth(@startdate_curr,5)

insert into Sales.Report_Sem (Country,[Sales Previous Sem],[Sales Current Sem])
select
	c.country as Country,
	sum(case when o.orderdate>=@startdate_prev and o.orderdate<=@enddate_prev
then od.qty*od.unitprice else 0 end) as [Sales Previous Sem],
	sum(case when o.orderdate>=@startdate_curr and o.orderdate<=@enddate_curr
then od.qty*od.unitprice else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>=@startdate_prev and o.orderdate<=@enddate_curr
group by
	c.country

end
```

7. Developerul isi da seama ca inmultirea dintre od.qty si od.unitprice ar putea fi facuta cu o functie si apoi functia inserata in procedura.
   Developerul creeaza functia:

```
create function Sales.Udf_Report_Sem
(@cantitate as int,
@pret as decimal(19,4)
)
returns decimal(19,4)
as
begin
	return @cantitate*@pret
end
```

8. Modifica procedura stocata ca sa insereze functia in locul inmultirii.

Data Training
Email: marcela.filip@datatraining.ro
Telefon: +40 721 61 27 85
Website: www.datatraining.ro

```
alter procedure Sales.Sp_Report_Sem
(@startdate_prev datetime)
as
begin
set nocount on;
delete from Sales.Report_Sem

declare
              @enddate_prev datetime,
              @startdate_curr datetime,
              @enddate_curr datetime


set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
set @enddate_curr=eomonth(@startdate_curr,5)

insert into Sales.Report_Sem (Country,[Sales Previous Sem],[Sales Current Sem])
select
       c.country as Country,
       sum(case when o.orderdate>=@startdate_prev and o.orderdate<=@enddate_prev then
Sales.Udf_Report_Sem(od.qty,od.unitprice) else 0 end) as [Sales Previous Sem],
       sum(case when o.orderdate>=@startdate_curr and o.orderdate<=@enddate_curr then
Sales.Udf_Report_Sem(od.qty,od.unitprice) else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>=@startdate_prev and o.orderdate<=@enddate_curr
group by
       c.country

end
```

9. Ruleaza procedura cu parametrul 2007-01-01 si verifica tabela

```
exec Sales.Sp_Report_Sem '2007-01-01'


select * from Sales.Report_Sem
```

	Country	Sales Previous Sem	Sales Current Sem
1	Argentina	1098.1000	718.5000
2	Austria	25911.8000	37240.1800
3	Belgium	7321.1000	4766.0000
4	Brazil	15213.2000	29337.3100
5	Canada	21732.6000	13237.5000
6	Denmark	14837.4000	12355.2500
7	Finland	9473.2000	4807.4500
8	France	25078.4000	22827.4000
9	Germany	57875.4000	66294.9300
10	Ireland	9520.5000	14438.5500

10. Testeaza aleatoriu 2-3 tari (etapa de Data Testing a developer-ului)

```
declare @startdate_prev datetime,
          @enddate_prev datetime,
          @startdate_curr datetime,
          @enddate_curr datetime

set @startdate_prev='2007-01-01'
set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
set @enddate_curr=eomonth(@startdate_curr,5)


select
    c.country as Country,
    sum(case when o.orderdate>=@startdate_prev and o.orderdate<=@enddate_prev then
Sales.Udf_Report_Sem(od.qty,od.unitprice) else 0 end) as [Sales Previous Sem],
    sum(case when o.orderdate>=@startdate_curr and o.orderdate<=@enddate_curr then
Sales.Udf_Report_Sem(od.qty,od.unitprice) else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>=@startdate_prev and o.orderdate<=@enddate_curr
and c.country='Canada'
group by c.country
```

	Country	Sales Previous Sem	Sales Current Sem
1	Canada	21732.6000	13237.5000

```
select * from Sales.Report_Sem
where country='Canada'
```

	Country	Sales Previous Sem	Sales Current Sem
1	Canada	21732.6000	13237.5000

11. Pregateste pachetul de Deploy (ce trebuie sa ajunga in productie si ordinea de rulare). Va creea un singur script, astfel incat adminul de baze de date sa ruleze tot deodata.

PACHET DEPLOY (1 singur script). Atentia la “GO” dintre statement-uri.

```
Use training;
Go
```

Data Training
Email: marcela.filip@datatraining.ro
Telefon: +40 721 61 27 85
Website: www.datatraining.ro

```
if OBJECT_ID('Sales.Report_Sem','U') is not null
drop table Sales.Report_Sem

GO


create table Sales.Report_Sem
([Country] nvarchar(100),
[Sales Previous Sem] decimal(19,4),
[Sales Current Sem] decimal(19,4)
)

go

if OBJECT_ID('Sales.Udf_Report_Sem','FN') is not null
drop function Sales.Udf_Report_Sem

GO

create function Sales.Udf_Report_Sem
(@cantitate as int,
@pret as decimal(19,4)
)
returns decimal(19,4)
as
begin
	return @cantitate*@pret
end

go

if OBJECT_ID('Sales.Sp_Report_Sem ','P') is not null
drop procedure Sales.Sp_Report_Sem

GO

create procedure Sales.Sp_Report_Sem
(@startdate_prev datetime)
as
begin
set nocount on;
delete from Sales.Report_Sem

declare
		@enddate_prev datetime,
		@startdate_curr datetime,
		@enddate_curr datetime


set @enddate_prev =eomonth(@startdate_prev,5)
set @startdate_curr=dateadd(mm,6,@startdate_prev)
set @enddate_curr=eomonth(@startdate_curr,5)

insert into Sales.Report_Sem (Country,[Sales Previous Sem],[Sales Current Sem])
```

Data Training
Email: marcela.filip@datatraining.ro
Telefon: +40 721 61 27 85
Website: www.datatraining.ro

```
select
	c.country as Country,
	sum(case when o.orderdate>=@startdate_prev and o.orderdate<=@enddate_prev then
Sales.Udf_Report_Sem(od.qty,od.unitprice) else 0 end) as [Sales Previous Sem],
	sum(case when o.orderdate>=@startdate_curr and o.orderdate<=@enddate_curr then
Sales.Udf_Report_Sem(od.qty,od.unitprice) else 0 end) as [Sales Current Sem]
from Sales.Customers as c
inner join sales.Orders as o on c.custid=o.custid
inner join sales.OrderDetails as od on od.orderid=o.orderid
where o.orderdate>=@startdate_prev and o.orderdate<=@enddate_curr
group by
	c.country

end

go


exec Sales.Sp_Report_Sem '2007-01-01'
```