كلاس حل تمرين

99/+1/19

آشنایی با View

۱- یک View تعریف کنید که تاریخ معاملات مربوط به دوچرخه را در خروجی نمایش دهد. پیوند جداول را طوری انجام دهید که فیلدهای زیر را در View داشته باشیم:

ProductName,

ProductNumber,

ProductCategory,

ProductSubCategory,

TransactionID,

ReferenceOrderID,

ReferenceOrderLineID,

TransactionDate,

TransactionType,

Quantity,

ActualCost

```
CREATE VIEW v Product TransactionHistory AS
       SELECT p.Name AS ProductName,
              p.ProductNumber,
              pc.Name AS ProductCategory,
              ps.Name AS ProductSubCategory,
              th.TransactionID,
              th.ReferenceOrderID,
              th.ReferenceOrderLineID,
              th.TransactionDate,
              th.TransactionType,
              th.Quantity,
              th.ActualCost
       FROM Production. Transaction History th
            INNER JOIN Production. Product p
             ON th.ProductID = p.ProductID
            INNER JOIN Production. Product Subcategory ps
             ON ps.ProductSubcategoryID = p.ProductSubcategoryID
            INNER JOIN Production. ProductCategory pc
             ON pc.ProductCategoryID = ps.ProductCategoryID
       WHERE pc.Name = 'Bikes';
```

نمونه خروجي

4	productname character varying (50)	productnumber character varying (25)	productcategory character varying (50)	productsubcategory character varying (50)	transactionid integer	referenceorderid integer	referenceorderlineid integer	transactiondate timestamp without time zone	transactiontype character (1)	quantity integer
1	Mountain-200 Black, 46	BK-M68B-46	Bikes	Mountain Bikes	100000	41590	0	2013-07-31 00:00:00	W	
2	Road-250 Black, 48	BK-R89B-48	Bikes	Road Bikes	100001	41591	0	2013-07-31 00:00:00	W	
3	Road-550-W Yellow, 38	BK-R64Y-38	Bikes	Road Bikes	100002	41592	0	2013-07-31 00:00:00	W	
4	Road-550-W Yellow, 40	BK-R64Y-40	Bikes	Road Bikes	100003	41593	0	2013-07-31 00:00:00	W	
5	Road-550-W Yellow, 42	BK-R64Y-42	Bikes	Road Bikes	100004	41594	0	2013-07-31 00:00:00	W	
6	Road-550-W Yellow, 44	BK-R64Y-44	Bikes	Road Bikes	100005	41595	0	2013-07-31 00:00:00	W	
7	Road-550-W Yellow, 48	BK-R64Y-48	Bikes	Road Bikes	100006	41596	0	2013-07-31 00:00:00	W	
8	Touring-1000 Yellow, 46	BK-T79Y-46	Bikes	Touring Bikes	100007	41597	0	2013-07-31 00:00:00	W	
9	Touring-1000 Yellow, 50	BK-T79Y-50	Bikes	Touring Bikes	100008	41598	0	2013-07-31 00:00:00	W	
10	Touring-1000 Blue, 46	BK-T79U-46	Bikes	Touring Bikes	100009	41599	0	2013-07-31 00:00:00	W	
11	Touring-1000 Blue, 54	BK-T79U-54	Bikes	Touring Bikes	100010	41600	0	2013-07-31 00:00:00	W	
12	Touring-2000 Blue, 54	BK-T44U-54	Bikes	Touring Bikes	100011	41601	0	2013-07-31 00:00:00	W	
13	Road-350-W Yellow, 40	BK-R79Y-40	Bikes	Road Bikes	100012	41602	0	2013-07-31 00:00:00	W	

آشنایی با View

2- (سوال تشویقی اختیاری) یک View طراحی کنید که آمار خرید مشتری را به تفکیک سال و ماه نشان دهد، که شامل تعداد کالا subcategory مرزش آنها است. همچنین در هر سطر نشان دهد این مشتری در آن سال و ماه از کدام و از کدام و category و از کدام این مشتری در آن سال و ماه از کدام و category و از کدام این مشتری در آن سال و ماه از کدام و از کدام و از کدام عداد کالای بیشتری خریده است، همچنین در هر سطر نشان دهد برای این مشتری در آن سال و ماه از کدام و از کدام عداد و از کدام عداد و این مشتری در آن سال و ماه از کدام و از کدام و از کدام عداد و این مشتری در آن سال و ماه از کدام و از کدام و از کدام و از کدام عداد و این کالاهای خریداری شده بیشتر بوده است. یک کد مشتری دلخواه را در متن دستور خود بگذارید طوری که ستونهای View مقدار داشته باشند. خروجی شامل فیلدهای و فیلدهایی با نام MaxProdCount و فیلدهایی با نام MaxCatByCount, است که مقادیر تجمیعی فوق را نشان دهند. نام فیلدهای بعدی را نیز Category یا Category که داخل هر کدام نام Category یا Category که حائز بیشترین تعداد یا ارزش برای آن ماه و آن مشتری شده ثبت می شود.

```
CREATE VIEW Q2 AS (
    with all details (customerID, productID, quantity, value, prodCatId, prodSubCatID, year, month) as
             select customer.customerid,
                    details.productid,
                    details.orderqty,
                    (details.unitprice * details.orderqty) as value,
                    productCat.productcategoryid,
                    productSubCat.productsubcategoryid,
                    extract (year from header.orderdate),
                    extract (month from header.orderdate)
             from sales.salesorderheader as header
                      inner join sales.customer customer using (customerid)
                      inner join sales.salesorderdetail details using (salesorderid)
                      inner join production.product product using (productid)
                      inner join production.productsubcategory productSubCat using (productsubcategoryid)
                      inner join production.productcategory productCat using (productcategoryid)
         ),
         customer_category_details(customerID, categoryID, sumQty, sumValue, year, month) as
             select customerID, prodCatId, sum(quantity), sum(value), year, month
             from all details
             group by customerID, prodCatId, year, month
        ),
```

```
customer category max count(customerID, maxCount, year, month) as
             select customerID,
                    max(sumQty),
                    year,
                    month
             from customer category details
             group by customerID, year, month
         ),
     customer category max count result (customerID, categoryID, year, month) as
             select customerID,
                    min(categoryID),
                    year,
                    month
             from customer category details
                      inner join customer category max count using (customerID, year, month)
             where customer category details.sumQty = customer category max count.maxCount
             group by customerID, year, month
         ),
     customer category max value (customerID, maxValue, year, month) as
             select customerID,
                    max(sumValue),
                    year,
                    month
             from customer category details
             group by customerID, year, month
         ),
```

```
customer category max value result(customerID, categoryID, year, month) as
             select customerID,
                    min(categoryID),
                    year,
                    month
             from customer category details
                      inner join customer category max value using (customerID, year, month)
             where customer category_details.sumValue = customer_category_max_value.maxValue
             group by customerID, year, month
        ),
     customer subcat details (customerID, subCatID, sumQty, sumValue, year, month) as
             select customerID, prodSubCatID, sum(quantity), sum(value), year, month
             from all details
             group by customerID, prodSubCatID, year, month
        ),
     customer subcategory max count (customerID, maxCount, year, month) as
             select customerID,
                    max(sumQty),
                    year,
                    month
             from customer subcat details
             group by customerID, year, month
        ),
```

```
customer subcategory max value (customerID, maxValue, year, month) as
             select customerID,
                    max(sumValue),
                    year,
                    month
             from customer subcat details
             group by customerID, year, month
         ),
     customer subcategory max value result(customerID, subCategoryID, year, month) as
             select customerID,
                    min(subCatID),
                    year,
                    month
             from customer subcat details
                      inner join customer subcategory max value using (customerID, year, month)
             where customer subcat details.sumValue = customer subcategory max value.maxValue
             group by customerID, year, month
         ),
     customer product details(customerID, sumProductQty, sumProductValues, year, month) as (
         select customerID,
                sum (quantity),
                sum (value),
                year,
                month
         from all details
         group by customerID, year, month
     ),
```

```
report (customerID, MaxProdCount, maxProdValue, year, month, MaxCatCountID, MaxCatValueID, MaxSubCatCountID,
            MaxSubCatValueID) as
             select customer product details.customerID
                                                                        as customerID,
                    customer product details.sumProductQty
                                                                        as MaxProcCount,
                    customer product details.sumProductValues
                                                                        as maxProdValue,
                    customer product details.year
                                                                        as year,
                    customer product details.month
                                                                        as month,
                    customer category max count result.categoryID
                                                                        as MaxCatCountID,
                    customer category max value result.categoryID
                                                                        as MaxCatValueID,
                    customer subcategory max count result.subCategoryID as MaxSubCatCountID,
                    customer subcategory max value result.subCategoryID as MaxSubCatValueID
             from customer product details
                      inner join customer category max count result using (customerID, year, month)
                      inner join customer category max value result using (customerID, year, month)
                      inner join customer subcategory max count result using (customerID, year, month)
                      inner join customer subcategory max value result using (customerID, year, month)
select customerID,
       year,
       month,
       MaxProdCount,
       maxProdValue,
       cat1.name as MaxCatByCount,
       cat2.name
                    as MaxCatByValue,
       subcat1.name as MaxSubCatByCount,
       subcat2.name as MaxSubCatByValue
from report
         inner join production.productcategory cat1 on cat1.productcategoryid = MaxCatCountID
         inner join production.productcategory cat2 on cat2.productcategoryid = MaxCatValueID
         inner join production.productsubcategory subcat1 on subcat1.productsubcategoryid = MaxSubCatCountID
         inner join production.productsubcategory subcat2 on subcat2.productsubcategoryid = MaxSubCatValueID
where customerID = 29825
                                                                                                         10 / 26
order by year, month
);
```

نمونه خروجي

Data	Output Explai	n Messages Noti	fications						
4	customerid integer	year double precision	month double precision	maxprodcount bigint	maxprodvalue numeric	maxcatbycount character varying (50)	maxcatbyvalue character varying (50)	maxsubcatbycount character varying (50)	maxsubcatbyvalue character varying (50)
1	29825	2011	5	26	20565.6206	Clothing	Bikes	Mountain Bikes	Mountain Bikes
2	29825	2011	8	39	18881.9746	Clothing	Bikes	Socks	Mountain Bikes
3	29825	2011	12	44	73261.1513	Bikes	Bikes	Mountain Bikes	Mountain Bikes
4	29825	2012	2	27	18031.6005	Clothing	Bikes	Mountain Bikes	Mountain Bikes
5	29825	2012	5	149	37182.6841	Clothing	Bikes	Gloves	Mountain Bikes
6	29825	2012	8	108	33134.0092	Clothing	Bikes	Mountain Bikes	Mountain Bikes
7	29825	2012	11	79	35989.7381	Clothing	Bikes	Mountain Bikes	Mountain Bikes
8	29825	2013	2	119	32230.2682	Clothing	Bikes	Mountain Bikes	Mountain Bikes
9	29825	2013	5	202	36828.4322	Clothing	Bikes	Jerseys	Mountain Bikes
10	29825	2013	8	155	48197.584	Clothing	Bikes	Mountain Bikes	Mountain Bikes
11	29825	2013	11	93	29140.800	Bikes	Bikes	Mountain Bikes	Mountain Bikes
12	29825	2014	3	166	39459.624	Clothing	Bikes	Mountain Bikes	Mountain Bikes

۳- دستوری بنویسید که تمام محصولات را به همراه SalesOrderID نمایش دهد حتی اگر هرگز سفارشی برای آن محصول ثبت نشده باشد.

SELECT SalesOrderID, P.ProductID, P.Name
FROM Production.Product AS P
LEFT OUTER JOIN Sales.SalesOrderDetail AS SOD
ON P.ProductID = SOD.ProductID

Data Outp	out Explain M	essages Not	ifications		
	salesorderid integer	productid integer	name character varying (50)		
1	43659	776	Mountain-100 Black, 42		
2	43659	777	Mountain-100 Black, 44		
3	43659	778	Mountain-100 Black, 48		
4	43659	771	Mountain-100 Silver, 38		
5	43659	772	Mountain-100 Silver, 42		
6	43659	773	Mountain-100 Silver, 44		
7	43659	774	Mountain-100 Silver, 48		
8	43659	714	Long-Sleeve Logo Jersey, M		
9	43659	716	Long-Sleeve Logo Jersey, XL		
10	43659	709	Mountain Bike Socks, M		
11	43659	712	AWC Logo Cap		
12	43659	711	Sport-100 Helmet, Blue		
13	43660	762	Road-650 Red, 44		
14	43660	758	Road-450 Red, 52		

✓ Successfully run. Total query runtime: 219 msec. 121555 rows affected.

۴- دستور نوشته شده در قسمت اول را تغییر دهید تا فقط کالاهایی که سفارش داده نشده اند ، نمایش داده شوند.

SELECT SalesOrderID, P.ProductID, P.Name
FROM Production.Product AS P
LEFT OUTER JOIN Sales.SalesOrderDetail AS SOD
ON P.ProductID = SOD.ProductID
WHERE SalesOrderID IS NULL

Data (Output	Explain	Messages		Notifications
4	salesore integer	derid 🛕	productid integer	0	name character varying (50)
1		[null]	4	97	Pinch Bolt
2		[null]	4	52	Lock Nut 2
3		[null]	4	96	Paint - Yellow
4		[null]	4	55	Lock Nut 3
5		[null]	3	84	Hex Nut 12
6		[null]	3	99	Head Tube
7		[null]	4:	37	Thin-Jam Lock Nut 11
8		[null]	5	18	ML Road Seat Assembly
9		[null]	3	97	Hex Nut 19
10		[null]	4	86	Metal Sheet 5
11		[null]	3:	30	Touring End Caps
12	12 [null]		8-	847 Headlights - Dual-Be	
13		[null]	3	59	Thin-Jam Hex Nut 9
14		[null]	4	05	External Lock Washer 9

✓ Successfully run. Total query runtime: 94 msec. 238 rows affected.

۵- دستوری بنویسید که تمام ردیف های جدول SalesOrderID به جدول SalesSalesOrderHeader را با بر اساس SalesOrderID و SalesYTD و SalesYTD را در نتایج بیاورید.

SELECT SalesOrderID, SalesPersonID, SalesYTD
FROM Sales.SalesPerson AS SP
 LEFT OUTER JOIN Sales.SalesOrderHeader AS SOH
 ON SP.BusinessEntityID = SOH.SalesPersonID

Data Output		utput Explain		Messages	Notifications		
4	saleso		<u></u>	salespersonid integer	<u></u>	salesytd numeric	
1		43	3659		279	2315185.611	
2		43	3660		279	2315185.611	
3		43	3661		282	2604540.7172	
4		43	3662		282	2604540.7172	
5		43	3663		276	4251368.5497	
6		43	3664		280	1352577.1325	
7		43	3665		283	1573012.9383	
8		43	3666		276	4251368.5497	
9		43	3667		277	3189418.3662	
10		43	3668		282	2604540.7172	
11		43	3669		283	1573012.9383	
12		43	3670		275	3763178.1787	
13		43	3671		283	1573012.9383	
14		43	3672		282	2604540.7172	

✓ Successfully run. Total query runtime: 1 secs 344 msec. 3806 rows affected.

۶- دستور نوشته شده قسمت ۵ را تغییر دهید تا نام فروشنده از جدول Person.Person نیز نشان داده شود.

SELECT SalesOrderID,
SalesPersonID,
SalesYTD,
FirstName,
MiddleName,
LastName

FROM Sales.SalesPerson AS SP

LEFT OUTER JOIN Sales.SalesOrderHeader AS SOH
ON SP.BusinessEntityID = SOH.SalesPersonID
LEFT OUTER JOIN Person.Person AS P
ON P.BusinessEntityID = SP.BusinessEntityID;

Data O	utput Explain	Messages Notifi	cations			
4	salesorderid integer	salespersonid integer	salesytd numeric	firstname character varying (50)	middlename character varying (50)	lastname character varying (50)
1	43659	279	2315185.611	[null]	[null]	[null]
2	43660	279	2315185.611	[null]	[null]	[null]
3	43661	282	2604540.7172	[null]	[null]	[null]
4	43662	282	2604540.7172	[null]	[null]	[null]
5	43663	276	4251368.5497	[null]	[null]	[null]
6	43664	280	1352577.1325	[null]	[null]	[null]
7	43665	283	1573012.9383	[null]	[null]	[null]
8	43666	276	4251368.5497	[null]	[null]	[null]
9	43667	277	3189418.3662	[null]	[null]	[null]
10	43668	282	2604540.7172	[null]	[null]	[null]
11	43669	283	1573012.9383	[null]	[null]	[null]
12	43670	275	3763178.1787	[null]	[null]	[null]
13	43671	283	1573012.9383	[null]	[null]	[null]
14	43672	282	2604540.7172	[null]	[null]	[null]

۷- جدول Sales.SalesOrderHeader شامل کلیدهای خارجی به جداول Sales.CurrencyRate است .برای Sales.SalesOrderHeader شامل کلیدهای خارجی به جداول Sales.SalesOrderHeader باشد حتی اگر منطبق پیوند سه جدول، یک دستور بنویسید و مطمئن شوید که شامل همه ردیف های CurrencyRateID ، AverageRate ، SalesOrderID قسمتی از خروجی است.)

SELECT CR.CurrencyRateID, CR.AverageRate, SM.ShipBase, SalesOrderID

FROM Sales.SalesOrderHeader AS SOH

LEFT OUTER JOIN Sales.CurrencyRate AS CR

ON SOH.CurrencyRateID = CR.CurrencyRateID

LEFT OUTER JOIN Purchasing.ShipMethod AS SM

ON SOH.ShipMethodID = SM.ShipMethodID

Data Ou	tput Explain Me	essages Notific	ations	
4	currencyrateid integer	averagerate numeric	shipbase numeric	salesorderid integer
1	[null]	[null]	8.99	43659
2	[null]	[null]	8.99	43660
3	4	1.4641	8.99	43661
4	4	1.4641	8.99	43662
5	[null]	[null]	8.99	43663
6	[null]	[null]	8.99	43664
7	[null]	[null]	8.99	43665
8	[null]	[null]	8.99	43666
9	[null]	[null]	8.99	43667
10	4	1.4641	8.99	43668
11	[null]	[null]	8.99	43669
12	[null]	[null]	8.99	43670
13	[null]	[null]	8.99	43671
14	4	1.4641	8.99	43672

Successfully run. Total query runtime: 730 msec. 31465 rows affected.

۸- دستوری بنویسید که ستون BusinessEntityID را از جدول Sales.SalesPerson از جدول - A ProductID به همراه هر ProductID از جدول - ProductID برگرداند.

SELECT SP.BusinessEntityID, P.ProductID FROM Sales.SalesPerson AS SP CROSS JOIN Production.Product AS P;

OR

SELECT SP.BusinessEntityID, P.ProductID FROM Sales.SalesPerson, Production.Product;

4	businessentityid integer	productid integer
1	274	1
2	275	1
3	276	1
4	277	1
5	278	1
6	279	1
7	280	1
8	281	1
9	282	1
10	283	1
11	284	1
12	285	1
13	286	1
14	287	1

✓ Successfully run. Total query runtime: 91 msec. 8568 rows affected.

۹- دستوری بنویسید که نام مشتریان به همراه نام کالاهای خریداری شده توسط آنها را نمایش دهد. (نکته: برای نوشتن این پرسش چهار جدول لازم است!)

SELECT soh.CustomerID, Prod.Name
FROM Sales.Customer AS C
INNER JOIN Sales.SalesOrderHeader AS SOH
ON C.CustomerID = SOH.CustomerID
INNER JOIN Sales.SalesOrderDetail AS SOD
ON SOH.SalesOrderID = SOD.SalesOrderID
INNER JOIN Production.Product AS Prod
ON SOD.ProductID = Prod.ProductID

4	customerid integer	name character varying (50)
1	29825	Mountain-100 Black, 42
2	29825	Mountain-100 Black, 44
3	29825	Mountain-100 Black, 48
4	29825	Mountain-100 Silver, 38
5	29825	Mountain-100 Silver, 42
6	29825	Mountain-100 Silver, 44
7	29825	Mountain-100 Silver, 48
8	29825	Long-Sleeve Logo Jersey, M
9	29825	Long-Sleeve Logo Jersey, XL
10	29825	Mountain Bike Socks, M
11	29825	AWC Logo Cap
12	29825	Sport-100 Helmet, Blue
13	29672	Road-650 Red, 44
14	29672	Road-450 Red, 52

Successfully run. Total query runtime: 241 msec. 121317 rows affected.

۱۰- با جستجوی اینترنتی، دستوری بنویسید که آمار تعداد رکورد کلیه جداول AW را نشان دهد.

SELECT schemaname, relname, n_live_tup FROM pg_stat_user_tables ORDER BY n_live_tup DESC;

4	schemaname name	relname name	n_live_tup bigint
1	sales	salesorderdetail	121317
2	production	transactionhistory	113443
3	production	transactionhistoryarchive	89253
4	production	workorder	72591
5	production	workorderrouting	67131
6	sales	salesorderheader	31465
7	sales	salesorderheadersalesreason	27647
8	sales	customer	19820
9	person	address	19614
10	sales	creditcard	19118
11	sales	personcreditcard	19118
12	sales	currencyrate	13532
13	purchasing	purchaseorderdetail	8845
14	purchasing	purchaseorderheader	4012
15	production	billofmaterials	2679

 $\checkmark\,$ Successfully run. Total query runtime: 123 msec. 70 rows affected.

* برای مراحل بعد، بزرگترین جدول از نظر تعداد رکورد را انتخاب کنید.

۱۱- براساس روش مطرح شده در درس، دستوری بنویسید که مشخص کند کدام فیلدهای این جدول ایندکس دارند و چه دستوری برای ساخت آن ایندکس استفاده شده است.

```
SELECT
    indexname,
    indexdef
FROM
    pg_indexes
WHERE
    tablename = 'salesorderdetail';
```

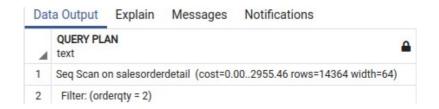
Oata Output Explain Messages Notifications		
indexname a name	<u></u>	Indexdef text
PK_SalesOrderDetail_SalesOrderID_SalesOrderDetailID		CREATE UNIQUE INDEX "PK_SalesOrderDetail_SalesOrderID_SalesOrderDetailID" ON sales.salesorderdetail USING btree (salesorderid, salesorderdetailid)

از بین فیلدهایی که روی آنها ایندکسی وجود ندارد یکی را با توجه به سوالات زیر انتخاب کنید (مثلا field1). فعلا ایندکسی نسازید (یا اگر بطور تستی ساختهاید drop کنید)، و دستوراتی بنویسید که:

۱۲ - رکورد(های) با شرط خاصی از نوع تساوی یعنی field1=myvalue1 را برگرداند.

EXPLAIN

SELECT *
FROM sales.salesorderdetail
WHERE orderqty = 2;

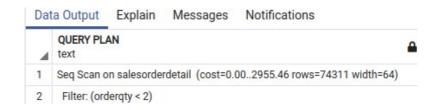


از بین فیلدهایی که روی آنها ایندکسی وجود ندارد یکی را با توجه به سوالات زیر انتخاب کنید (مثلا field1). فعلا ایندکسی نسازید (یا اگر بطور تستی ساختهاید drop کنید)، و دستوراتی بنویسید که:

۱۳- رکورد(های) با شرط خاصی از نوع کوچکتر یعنی field1<myvalue2 را برگرداند.

EXPLAIN

SELECT *
FROM sales.salesorderdetail
WHERE orderqty < 2;</pre>

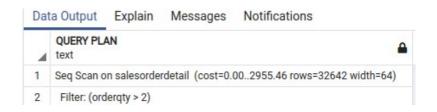


از بین فیلدهایی که روی آنها ایندکسی وجود ندارد یکی را با توجه به سوالات زیر انتخاب کنید (مثلا field1). فعلا ایندکسی نسازید (یا اگر بطور تستی ساختهاید drop کنید)، و دستوراتی بنویسید که:

۱۴- رکورد(های) با شرط خاصی از نوع برزگتر یعنی field1>myvalue2 را برگرداند.

EXPLAIN

SELECT *
FROM sales.salesorderdetail
WHERE orderqty > 2;



```
اکنون ایندکسی روی فیلد field1 بسازید و دستورات زیر را بنویسید:
```

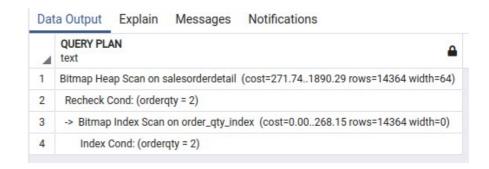
۱۵- دستور سوال (۱۲) را دوباره با استفاده از ایندکس اجرا کنید.

(۱۲ - رکورد(های) با شرط خاصی از نوع تساوی یعنی field1=myvalue1 را برگرداند.)

CREATE INDEX ORDER_QTY_INDEX on sales.salesorderdetail (orderqty);

EXPLAIN

SELECT *
FROM sales.salesorderdetail
WHERE orderqty = 2;



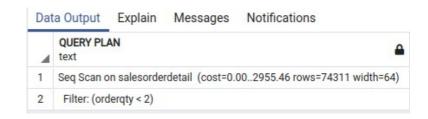
اکنون ایندکسی روی فیلد field1 بسازید و دستورات زیر را بنویسید:

۱۶- دستور سوال (۱۳) را دوباره با استفاده از ایندکس اجرا کنید.

(۱۳- رکورد(های) با شرط خاصی از نوع کوچکتر یعنی field1<myvalue2 را برگرداند.)

EXPLAIN

SELECT *
FROM sales.salesorderdetail
WHERE orderqty < 2;</pre>



اکنون ایندکسی روی فیلد field1 بسازید و دستورات زیر را بنویسید:

۱۶- دستور سوال (۱۴) را دوباره با استفاده از ایندکس اجرا کنید.

(۱۴- رکورد(های) با شرط خاصی از نوع بزرگتر یعنی field1>myvalue2 را برگرداند.)

EXPLAIN

SELECT *
FROM sales.salesorderdetail
WHERE orderqty > 2;

