

目录

1	实验目的	2
2	实验内容	2
2.1	创建视图	2
2.2	创建存储过程	2
2.3	练习使用游标	2
2.4	创建触发器	3
2.5	创建索引	3
3	实验步骤及结果	3
3.1	创建视图	3
3.2	创建存储过程	4
3.3	练习使用游标	5
3.4	创建触发器	8
3.5	创建索引	10
4	实验总结	12

1 实验目的

1. 熟悉大型数据库实验环境，以 MS SQL SERVER 为例
2. 掌握视图
3. 掌握存储过程与触发器
4. 掌握 MS SQL SERVER 的导入和导出
5. 掌握 MS SQL SERVER 的索引

2 实验内容

2.1 创建视图

使用“实验一”中的数据库“abc”，创建一个视图，生产厂家为“北京”且价格低于北京生产的产品的平均价格，输出产品的名称、价格和生产厂家

2.2 创建存储过程

使用“实验一”中的数据库“abc”，创建一个带有输入参数的存储过程 proc_abc, 查询指定职工的销售记录，用户输入职工编号，存储过程返回职工名称、产品名称、销售日期、销售数量，假如执行存储过程时所提供的“职工编号”不存在，存储过程应给予一定的提示

2.3 练习使用游标

使用“实验一”中的数据库“abc”，练习使用游标，写出按如下报表形式显示结果的 SQL 语句，该报表查询每年每种产品总销售金额，（总销售金额=价格销量），报表显示格式如下所示：

年	产品号	产品名	销售总量	总销售金额（万元）
2001 年	2	AAA	590	3.2
2002 年	5	BBB	644	23.3
2002 年	1	CCC	32	0.2

2.4 创建触发器

使用“实验一”中的数据库“abc”，练习使用触发器，在销售表上创建触发器 tr_updateprice，每次新增销售记录时，自动更新产品表的单价，更新方法是：每增加一笔销售记录，就将该产品的单价减去 1 块钱。

2.5 创建索引

将 100 万行网络连接监控数据 Netflow 导入数据库，创建多个索引，观察创建索引对数据库文件大小的影响；并设计不同的查询语句来观察索引对查询效率的影响；可以尝试将 100 万行记录扩展为 1000 万行，然后再做索引和查询的实验？文件见附件。

3 实验步骤及结果

3.1 创建视图

使用“实验一”中的数据库“abc”，创建一个视图，生产厂家为“北京”且价格低于北京生产的产品的平均价格，输出产品的名称、价格和生产厂家。

使用 create view 语句创建视图，代码如下：

```

1 create view view1
  as
3 select CPM as '产品名',JG as '价格',SCCJ as '生产厂家'
  from CPB

```

```

5 where SCCJ='北京' and JG<(select AVG(JG) from CPB where SCCJ
   ='北京')
7
select *
from view1

```

create view

使用视图查询的结果如下：

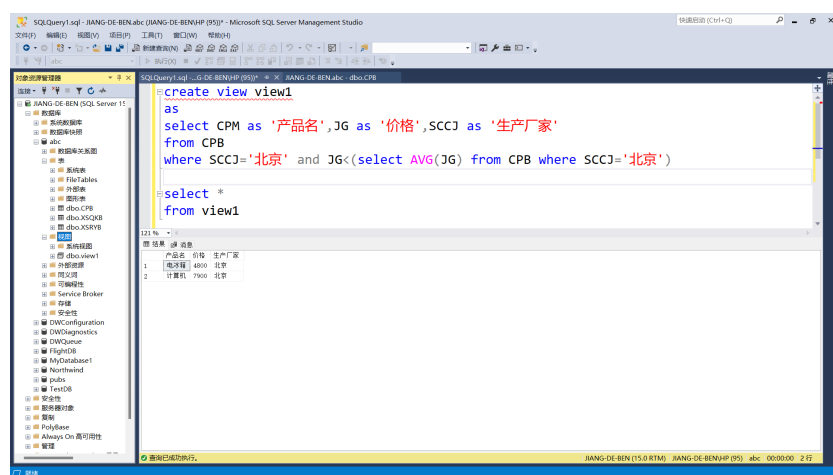


图 1: 视图查询结果

3.2 创建存储过程

使用“实验一”中的数据库“abc”，创建一个带有输入参数的存储过程 proc_abc, 查询指定职工的销售记录，用户输入职工编号，存储过程返回职工名称、产品名称、销售日期、销售数量，假如执行存储过程时所提供的“职工编号”不存在，存储过程应给予一定的提示。

使用 create procedure 语句创建存储过程，代码如下：

```

2 create procedure proc_abc
  @myzgh char(6)

```

```

as declare @count int
4  set @count=
(
6  select count(*)
from XSRYB
8  where ZGH=@myzgh
)
10 if @count>0
begin
12     select XSRYB.XM as '姓名',CPB.CPM as '产品名',XSQKB.XSRQ
        as '销售日期',XSQKB.XSSL as '销售数量'
14     from XSRYB inner join XSQKB on XSRYB.ZGH=XSQKB.ZGH
        inner join CPB on CPB.CPH=XSQKB.CPH
        where XSRYB.ZGH=@myzgh
16 end
else
18 begin
    print '查询职工号不存在！'
20 end

```

create procedure

使用存储过程查询的结果如下：

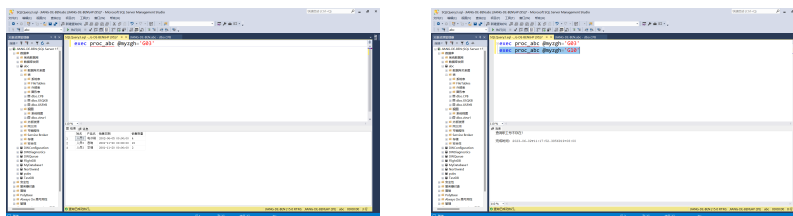


图 2: 存储过程查询结果

3.3 练习使用游标

创建游标的代码如下：

```

declare
2  abc_cursor2 cursor for
   select year(XSRQ),CPH
4  from XSQKB
   group by year(XSRQ),CPH

```

创建游标

练习使用游标，写出按照上述报表形式显示结果的 SQL 语句，该报表查询每年每种产品总销售金额，代码如下：

```

1  open abc_cursor2

3  declare
    @YEAR int,
5    @CPH char(6),
    @CPM varchar(20),
7    @XSZL int,
    @ZXSJE float

9

11  create table #temp_cursor2
   (
13    [year] [int],
    [cph] [char](6),
15    [cpm] [varchar](20),
    [xszl] [int],
    [zxsje] [float]
17  )on [primary]

19  fetch next from abc_cursor2
   into @YEAR,@CPH

21

23  while @@FETCH_STATUS=0
   begin
       set @XSZL=
25    (

```

```

27         select sum(XSSL)
28         from XSQKB
29         where CPH=@CPH and year(XSRQ)=@YEAR
30     )
31
32     set @CPM=
33     (
34         select CPM
35         from CPB
36         where CPH=@CPH
37     )
38
39     set @ZXSJE=
40     (
41         select JG*@XSZL
42         from CPB
43         where CPH=@CPH
44     )
45     insert into #temp_cursor2
46     values (@YEAR,@CPH,@CPM,@XSZL, cast ((@ZXSJE/10000) as
47         varchar(10)))
48     fetch next from abc_cursor2 into @YEAR,@CPH
49
50 end
51
52 close abc_cursor2
53 deallocate abc_cursor2
54
55 select year as '年',cph as '产品号',cpm as '产品名',xszl as
56     '销售总量',zxsje as '总销售金额（万元）'
57 from #temp_cursor2
58
59 drop table #temp_cursor2

```

练习使用游标

使用游标查询的结果如下：

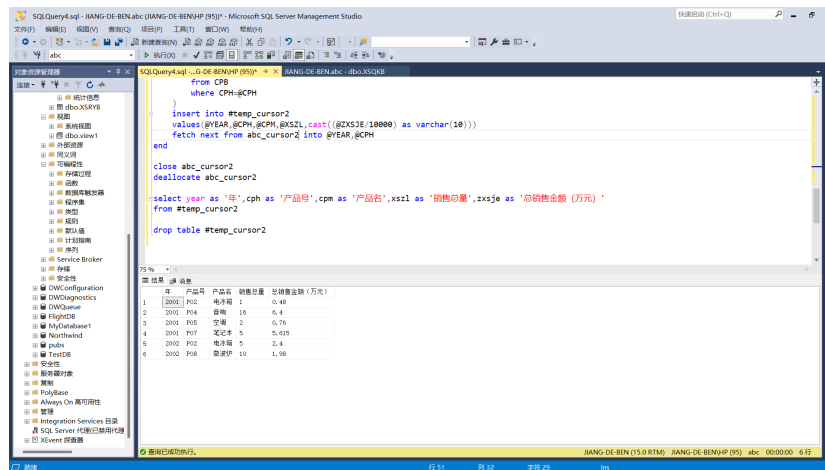


图 3: 游标查询结果

3.4 创建触发器

使用“实验一”中的数据库“abc”，练习使用触发器，在销售表上创建触发器 tr_updateprice，每次新增销售记录时，自动更新产品表的单价，更新方法是：每增加一笔销售记录，就将该产品的单价减去 1 块钱。

创建触发器的代码如下：

```

1      create trigger tr_updateprice
2      on XSQKB
3      for insert ,update
4      as
5      declare
6          @newXSSL int ,
7          @newJGCPH char(6)
8
9      set @newXSSL=(select XSSL from inserted)
10     set @newJGCPH=(select CPH from inserted)
11
12     update CPB
13     set JG=JG-@newXSSL
14     where CPH=@newJGCPH

```


创建触发器

使用触发器的代码如下：

```
2 select * from XSQKB  
select * from CPB  
  
4 insert into XSQKB  
values  
6 ('G01','P02','2023-5-30',10)  
  
8 select * from XSQKB  
select * from CPB
```

使用触发器

使用触发器查询的结果如下：

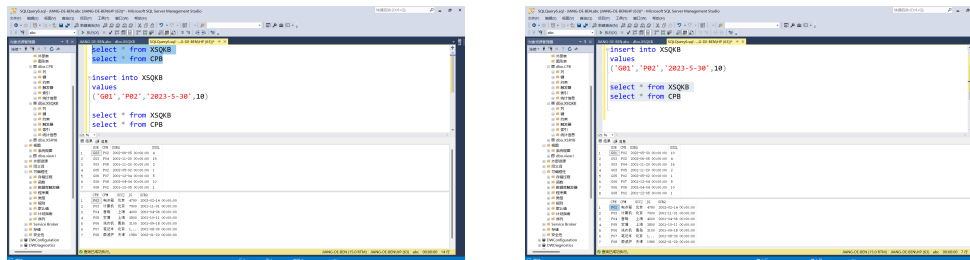


图 4: 使用触发器查询结果

3.5 创建索引

将 100 万行数据导入数据库，创建索引，比较引入索引前后的查询速度

将 txt 文件导入数据库如图所示



图 5: 导入数据

使用 select 语句查询数据结果如下

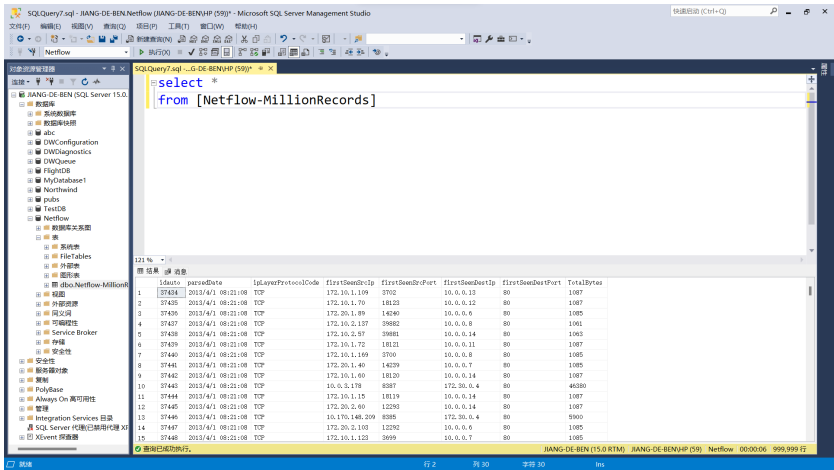


图 6: 查询数据

使用 create index 语句创建索引，代码如下：

```
1 create clustered index netflow_s1
on [Netflow-MillionRecords] (idauto)

3

5 create index netflow_s2
on [Netflow-MillionRecords] (ipLayerProtocolCode)

7

create index netflow_s3
on [Netflow-MillionRecords] (parsedDate)
```

create index

创建索引前后数据库内存大小比较

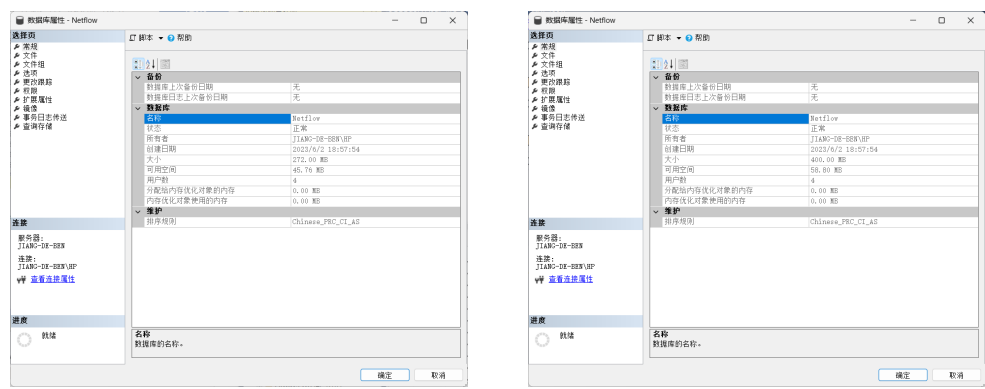


图 7: 创建索引前后数据库内存大小比较

使用 select 语句查询数据结果如下

可以看出，创建索引后，数据库内存增加较多，在数据为 100 万行的时候，查询速度并没有很大差异，推测原因是数据量较小，查询速度并没有明显差异，同时 SQL server 的版本为 2019，查询速度已经很快，所以创建索引对查询速度的影响不明显。但是当数据量增大时，查询速度会有明显差异。当数据量达到 1000 万行时，查询速度会有比较明显的差异。

Idx	parsedDate	ipLayerProtocolCode	firstIpDestIp	firstIpDestPort	firstIpDestIp	firstIpDestPort	TotalBytes
1	2013/4/1 08:05:00	TCP	172.16.1.102	2025	10.0.0.11	80	1085
2	2013/4/1 08:05:00	TCP	172.16.1.148	9647	10.0.0.14	80	1087
3	2013/4/1 08:10:00	TCP	172.16.1.232	9310	10.0.0.6	80	1085
4	2013/4/1 08:13:00	TCP	172.16.1.11	11949	10.0.0.10	80	1087
5	2013/4/1 08:42:58	TCP	172.16.2.86	64877	10.0.0.6	80	1085
6	2013/4/1 08:42:58	TCP	172.16.1.176	56321	10.0.0.8	80	1085
7	2013/4/1 08:42:58	TCP	172.16.1.120	56320	10.0.0.9	80	1085
8	2013/4/1 08:42:58	TCP	172.16.1.159	2823	10.0.0.12	80	1087
9	2013/4/1 08:42:58	TCP	172.16.1.241	32829	10.0.0.8	80	1085
10	2013/4/1 08:42:58	TCP	172.16.1.194	56324	10.0.0.6	80	1085
11	2013/4/1 08:42:58	TCP	172.16.1.199	56298	10.0.0.7	80	1085
12	2013/4/1 08:42:58	TCP	172.16.1.151	2833	10.1.0.100	80	1089
13	2013/4/1 08:42:58	TCP	172.16.1.137	56296	10.0.0.9	80	1085
14	2013/4/1 08:42:58	TCP	172.16.1.141	56314	10.0.0.9	80	1085
15	2013/4/1 08:11:00	TCP	172.16.1.179	2236	10.0.0.9	80	1085

图 8: 查询数据

4 实验总结

本次实验主要学习了视图、存储过程、游标、触发器、索引的使用方法，通过实验，我对这些知识有了更深入的理解。

在创建视图的时候，视图的创建语句中，where 子句中不能使用聚合函数，否则会报错；在创建存储过程的时候，存储过程中的变量必须先声明后使用，否则会报错；在创建游标的时候，游标的使用方法和 C 语言中的游标类似，都是先声明游标，然后打开游标，然后使用 fetch next 语句获取游标中的数据，最后关闭游标；在创建触发器的时候，触发器的创建语句中，for 子句中可以使用 insert、update、delete，表示在插入、更新、删除数据时触发器会被触发；在使用触发器的时候，触发器中的变量使用 inserted 表示插入的数据，使用 deleted 表示删除的数据，使用 update 表示更新的数据；

在创建索引的时候，使用了用空间换时间的思想，通过创建索引，可以提高查询速度，但是会增加数据库的内存占用。