课程名称：数据库原理

计算机科学与技术及相关专业 2018 级，2020 年春季

实验任务书（实验一）

实验报告要求：

1. 列出所有的 SQL 语句和源代码；
2. 程序要求有适当的注释；
3. 实验报告提交电子档。

实验内容：

1、用 SQL 语句创建数据库 CAP，数据文件名为 CAPData.mdf，数据文件的初始存储空间大小为 50M，最大存储空间为 500M，存储空间自动增长量为 10M。

2、在 CAP 数据库中用 SQL 语句创建下面的 4 张表，合理设计每个字段的数据类型，建立主键与外键约束。表 Products 中的 Price 字段不允许为空。表 Customers 的 discnt 字段取值范围在[0,30]之间。利用 SQL 语句向表中添加如下数据：

Customers

cid	cname	city	discnt
C001	TipTop	Duluth	10.00
C002	Basics	Dallas	12.00
C003	Allied	Dallas	8.00
C004	ACME	Duluth	8.00
C005	Oriental	Kyoto	6.00
C006	ACME	Kyoto	0.00

Products

Pid	pname	city	quantity	price
P01	comb	Dallas	111400	0.50
P02	brush	Newark	203000	0.50
P03	razor	Duluth	150600	1.00
P04	Pen	Duluth	125300	1.00
P05	pencil	Dallas	221400	1.00
P06	folder	Dallas	123100	2.00
P07	case	Newark	100500	1.00

Agents

Aid	aname	city	percent

A01	smith	New York	6
A02	Jones	Newark	6
A03	Brown	Tokyo	7
A04	Gray	New York	6
A05	Otasi	Duluth	5
A06	Smith	Dallas	5

Orders

Ordno	month	cid	aid	pid	qty	dollars
1011	Jan	C001	A01	P01	1000	450.00
1012	Jan	C001	A01	P01	1000	450.00
1019	Feb	C001	A02	P02	400	180.00
1017	Feb	C001	A06	P03	600	540.00
1018	Feb	C001	A03	P04	600	540.00
1023	Mar	C001	A04	P05	500	450.00
1022	Mar	C001	A05	P06	400	720.00
1025	Apr	C001	A05	P07	800	720.00
1013	Jan	C002	A03	P03	1000	880.00
1026	May	C002	A05	P03	800	704.00
1015	Jan	C003	A03	P05	1200	1104.00
1014	Jan	C003	A03	P05	1200	1104.00
1021	Feb	C004	A06	P01	1000	460.00
1016	Jan	C004	A01	P01	1000	500.00
1020	Feb	C005	A03	P07	600	600.00
1024	Mar	C006	A06	P01	800	400.00

3、 利用系统预定义的存储过程 sp_helpdb 查看数据库的相关信息，例如所有者、大小、创建日期等。

4、 利用系统预定义的存储过程 sp_helpconstraint 查看表中出现的约束（包括 Primary key, Foreign key, check constraint, default, unique）。

5、 创建一张表 Orders_Jan，表的结构与 Orders 相同，将 Orders 表中 month 为‘Jan’的订单记录复制到表 Orders_Jan 中。

6、 将 Orders 表中 month 为‘Jan’的订单记录全部删掉。

7、 对曾经下过金额（dollars）大于 500 的订单的客户，将其 discnt 值增加 2 个百分点（+2）。

8、 写一段 TSQL 程序，向表 Orders 中增加 5000 条记录，要求订单尽可能均匀地分布在 12 个月中。

9、 在表 Orders 的’month’字段上建立索引。

10、 创建一个视图 order_month_summary，视图中的字段包括月份、该月的订单总量和该月的订单总金额。基于视图 order_month_summary，查询第一季度各个月份的订单总量和订单总金额。

```
一
create database CAP
on(
name=CAPData,
filename='E:\数据库原理\CAPData.mdf',
size=50,
maxsize=500,
filegrowth=10
)
log on(
name=CAP_log,
filename='E:\数据库原理\CAPlog.ldf',
size=5,
maxsize=50,
filegrowth=5
)
CREATE TABLE [dbo].[Customers](
    [cid] [varchar](12) NOT NULL,
    [cname] [varchar](12) NULL,
    [city] [varchar](12) NULL,
    [discnt] [numeric](4, 2) NULL,
PRIMARY KEY CLUSTERED
(
    [cid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO

SET ANSI_PADDING OFF
GO

ALTER TABLE [dbo].[Customers]  WITH CHECK ADD CHECK  (([discnt]>=(0) AND
[discnt]<=(30)))
GO



CREATE TABLE [dbo].[products](
    [pid] [varchar](12) NOT NULL,
    [pname] [varchar](12) NULL,
    [city] [varchar](12) NULL,
    [quantity] [numeric](10, 0) NULL,
```

```
    [price] [numeric](3, 2) NOT NULL,
PRIMARY KEY CLUSTERED
(
    [pid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO

SET ANSI_PADDING OFF
GO
CREATE TABLE [dbo].[agents](
    [aid] [varchar](12) NOT NULL,
    [aname] [varchar](12) NULL,
    [city] [varchar](12) NULL,
    [percent1] [numeric](2, 0) NULL,
PRIMARY KEY CLUSTERED
(
    [aid] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO

SET ANSI_PADDING OFF
GO

CREATE TABLE [dbo].[orders](
    [ordno] [varchar](12) NULL,
    [month1] [varchar](12) NULL,
    [cid] [varchar](12) NULL,
    [aid] [varchar](12) NULL,
    [pid] [varchar](12) NULL,
    [qty] [numeric](5, 0) NULL,
    [dollars] [numeric](6, 2) NULL
) ON [PRIMARY]

GO

SET ANSI_PADDING OFF
GO
```

```
ALTER TABLE [dbo].[orders]  WITH CHECK ADD FOREIGN KEY([aid])
REFERENCES [dbo].[agents] ([aid])
GO

ALTER TABLE [dbo].[orders]  WITH CHECK ADD FOREIGN KEY([cid])
REFERENCES [dbo].[Customers] ([cid])
GO

ALTER TABLE [dbo].[orders]  WITH CHECK ADD FOREIGN KEY([pid])
REFERENCES [dbo].[products] ([pid])
GO
```

二

```
exec sp_helpdb CAP
use CAP
EXEC sp_helpconstraint Customers
EXEC sp_helpconstraint products
EXEC sp_helpconstraint agents
EXEC sp_helpconstraint orders
```

三

```
select *  into orders_jan
from orders
where 1=2

insert into orders_jan
select *
from orders
where month1='Jan'
select * from orders_jan
```

四

```
update customers
set discnt+=2
where cid in(select distinct customers.cid from customers,orders where
customers.cid=orders.cid and dollars>500)
select * from Customers
```

五

```
declare @a int
declare @b int

set @b=1
while @b<=5000
begin
```

```
 set @a=ceiling(rand()*12)
 begin
 if @a=1
 insert into orders(month1) values('Jan')
 else if @a=2
 insert into orders(month1) values('Feb')
 else if @a=3
 insert into orders(month1) values('Mar')
 else if @a=4
 insert into orders(month1) values('Apr')
 else if @a=5
 insert into orders(month1) values('May')
 else if @a=6
 insert into orders(month1) values('Jun')
 else if @a=7
 insert into orders(month1) values('Jul')
 else if @a=8
 insert into orders(month1) values('Aug')
 else if @a=9
 insert into orders(month1) values('Sept')
 else if @a=10
 insert into orders(month1) values('Oct')
 else if @a=11
 insert into orders(month1) values('Nov')
 else if @a=12
 insert into orders(month1) values('Dec')
 end
set @b+=1
end
create clustered index o1 on orders(month1)
go
create view order_month_summary
as select month1,qty,dollars
from orders
go
select sum(qty),sum(dollars)
from order_month_summary
group by(month1)
```

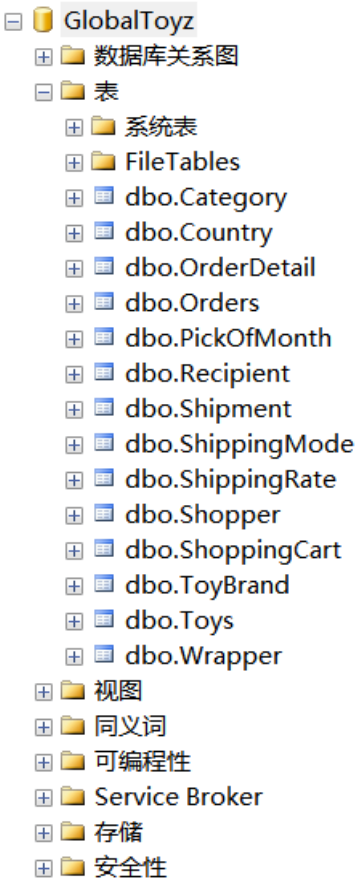
实验任务书（实验二）

实验报告要求：

1. 列出所有的 SQL 语句和源代码；
2. 程序要求有适当的注释；
3. 实验报告提交电子档。

实验内容：

一、熟悉示例数据库



运行给定的 SQL Script，建立数据库 Glo

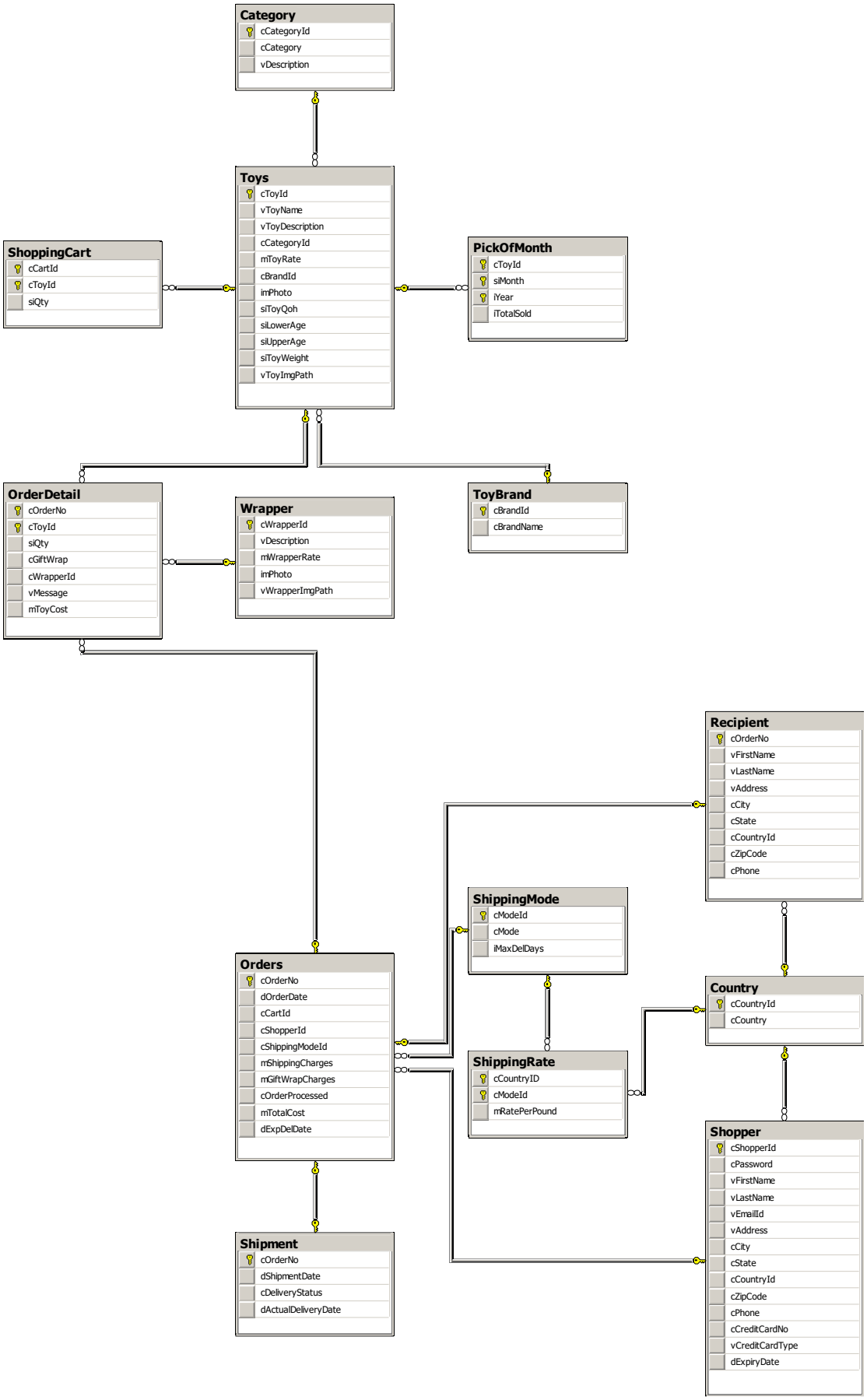
1．balToyz。
2．创建数据库关系图，了解表的结构。
3．在 Orders 表中增加 1000 笔订单数据，注意与其它表之间的关系。
4．在 Orders 表中新增数据之后，更新 PickofMonth 表。

二、查询、更新数据库

1. 查找属于 California 和 Florida 州的顾客的名、姓和 emailID。
2. 查找定单号码、顾客 ID，定单的总价值，并以定单的总价值的升序排列。
3. 查找在 orderDetail 表中 vMessage 为空值的行。
4. 查找玩具名字中有“Racer”字样的所有玩具的基本资料。
5. 根据 2016 年的玩具销售总数，查找“Pick of the Month”玩具的前五名玩具的

ID。

6. 根据 OrderDetail 表，查找玩具总价值大于￥50 的定单的号码和玩具总价值。
7. 查找一份包含所有装运信息的报表，包括：Order Number, Shipment Date, Actual Delivery Date, Days in Transit. (提示：Days in Transit = Actual Delivery Date – Shipment Date)
8. 查找所有玩具的名称、商标和种类（Toy Name, Brand, Category）。
9. 查找玩具的名称和所有玩具的购物车 ID。如果玩具不在购物车中，也需在结果中出现。
10. 以下列格式查找所有购物者的名字和他们的简称：（Initials, vFirstName, vLastName）,例如 Angela Smith 的 Initials 为 A.S。
11. 查找所有玩具的平均价格，并舍入到整数。
12. 查找所有购买者和收货人的名、姓、地址和所在城市，要求保留结果中的重复记录。
13. 查找没有包装的所有玩具的名称。（要求用子查询实现）
14. 查找已收货定单的定单号码以及下定单的时间。（要求用子查询实现）
15. 查找一份基于 Orderdetail 的报表，包括 cOrderNo,cToyId 和 mToyCost，记录以 cOrderNo 升序排列，并计算每一笔定单的玩具总价值。
16. 查找从来没有下过订单的顾客。
17. 删除“Largo”牌的所有玩具。


Category
cCategoryId
cCategory
vDescription
Toys
cToyId
vToyName
vToyDescription
cCategoryId
mToyRate
cBrandId
imPhoto
siToyQoh
siLowerAge
siUpperAge
siToyWeight
vToyImgPath
ShoppingCart
cCartId
cToyId
siQty
PickOfMonth
cToyId
siMonth
iYear
iTotalSold
OrderDetail
cOrderNo
cToyId
siQty
cGiftWrap
cWrapperId
vMessage
mToyCost
Wrapper
cWrapperId
vDescription
mWrapperRate
imPhoto
vWrapperImgPath
ToyBrand
cBrandId
cBrandName
Recipient
cOrderNo
vFirstName
vLastName
vAddress
cCity
cState
cCountryId
cZipCode
cPhone
ShippingMode
cModeId
cMode
iMaxDelDays
Orders
cOrderNo
dOrderDate
cCartId
cShopperId
cShippingModeId
mShippingCharges
mGiftWrapCharges
cOrderProcessed
mTotalCost
dExpDelDate
Country
cCountryId
cCountry
ShippingRate
cCountryID
cModeId
mRatePerPound
Shopper
cShopperId
cPassword
vFirstName
vLastName
vEmailId
vAddress
cCity
cState
cCountryId
cZipCode
cPhone
cCreditCardNo
vCreditCardType
dExpiryDate
Shipment
cOrderNo
dShipmentDate
cDeliveryStatus
dActualDeliveryDate

一

```
declare @m int
set @m = 0

while @m <= 1000
begin
set @m  += 1
insert into Orders(cOrderNo,dOrderDate,cCartId,cShopperId)
values('000011',2020-6-7,'000011','000011')
end
```

二

```
select vLastName as 姓,vFirstName as 名,vemailid as emailID
from Shopper
where cState = 'California' or cState = 'Florida'
```

	姓	名	emailID
1	Johnson	Barbara	barbaraj@speedmail.com
2	Roberts	Catherine	catheriner@qmail.com
3	Brown	Charles	charlesb@speedmail.com
4	Miller	Cynthia	cynthiam@qmailcom
5	Moore	David	davidm@qmail.com
6	Martinez	Joseph	josephm@qmail.com
7	Wright	Patricia	patreciaw@speedmail.com
8	Lopez	Paul	paull@qmail.com
9	Scott	Robert	Roberts@speedmail.com
10	Adams	Sandra	Sandra@qmail.com
11	Baker	Sarah	sarahb@qmail.com
12	Cooper	David	davidc@speedmail.com

三

```
select cOrderNo,cShopperId,mTotalCost
from Orders
order by mTotalCost ASC
```

	cOrderNo	cShopperId	mTotalCost
1	000007	000008	16.99
2	000009	000010	26.99
3	000004	000006	40.99
4	000008	000009	53.98
5	000001	000002	62.22
6	000010	000003	67.97
7	000003	000007	83.97
8	000002	000005	96.50
9	000006	000012	97.97
10	000005	000002	231.68

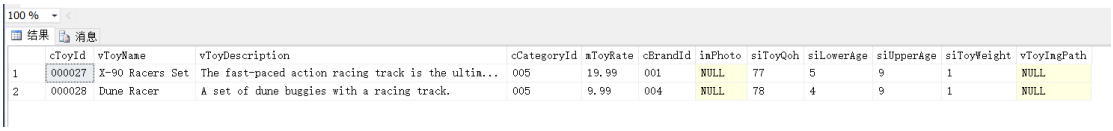
四

```
select *
from OrderDetail
where vMessage is null
```

	cOrderNo	cToyId	siQty	cGiftWrap	cWrapperId	vMessage	mToyCost
1	000001	000007	2	N	NULL	NULL	39.98
2	000003	000017	3	N	NULL	NULL	71.97
3	000007	000006	1	N	NULL	NULL	12.99

五

```
select *
from Toys
where vToyName like '%Racer%'
```



	cToyId	vToyName	vToyDescription	cCategoryId	mToyRate	cBrandId	imPhoto	siToyQoh	siLowerAge	siUpperAge	siToyWeight	vToyImgPath
1	000027	X-90 Racers Set	The fast-paced action racing track is the ultim...	005	19.99	001	NULL	77	5	9	1	NULL
2	000028	Dune Racer	A set of dune buggies with a racing track.	005	9.99	004	NULL	78	4	9	1	NULL

六

```
select top 5 *
from PickOfMonth
where iYear = 2016
order by iTotalSold DESC
```

结果 消息

	cToyId	siMonth	iYear	iTotalSold
1	000026	12	2016	6500
2	000003	7	2016	5670
3	000011	9	2016	5600
4	000007	4	2016	5000
5	000021	11	2016	4500

七

```
select cToyId,mToyCost
from OrderDetail
where mToyCost > 50
```

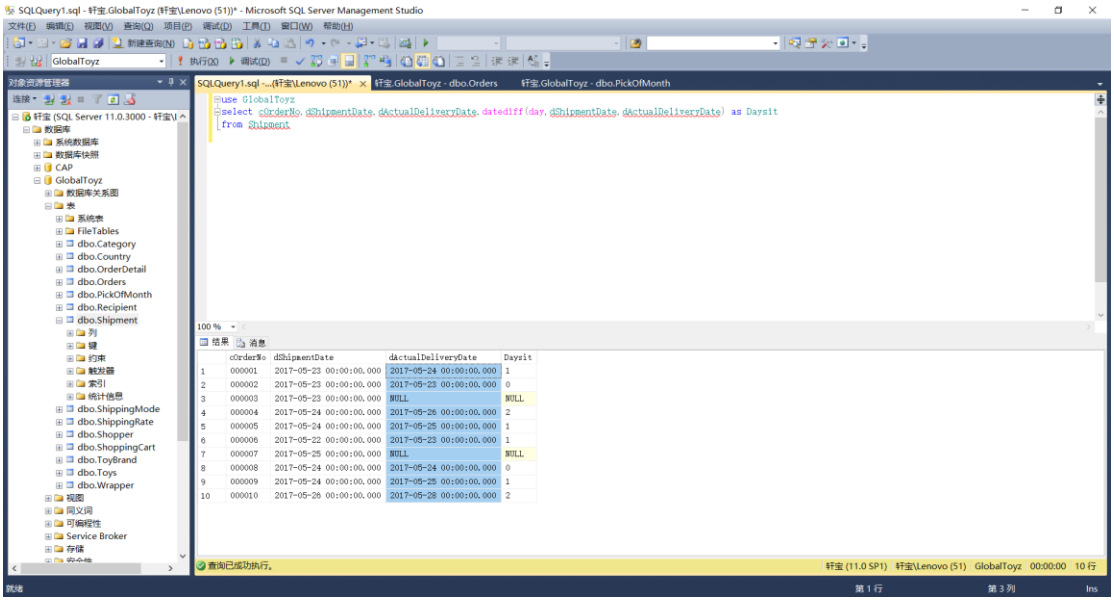
结果 消息

	cToyId	mToyCost
1	000016	86.50
2	000017	71.97
3	000030	71.98

八

```
select
cOrderNo,dShipmentDate,dActualDeliveryDate,datediff(day,dShipmentDate
,dActualDeliveryDate) as Daysit
```

```
from Shipment
```



九

```
select vToyName,cBrandName,cCategory
from Toys,ToyBrand,Category
where Toys.cBrandId = ToyBrand.cBrandId and Toys.cCategoryId =
Category.cCategoryId
```

结果 消息

	vToyName	cBrandName	cCategory
1	Robby the Whale	Bobby	Activity
2	Water Channel System	Bobby	Activity
3	Parachute and Rocket	The Bernie Kids	Activity
4	Super Deluge	LAMOBIL	Activity
5	Light Show Lamp	Bobby	Dolls
6	Glass Decoration	Largo	Dolls
7	Tie Dye Kit	Frances-Price	Dolls
8	Alice in Wonderland	Bobby	Dolls
9	Glamorous Doll	Bobby	Dolls
10	Sleeping Beauty Doll	LAMOBIL	Dolls
11	Pet Loving Doll	Bobby	Dolls

十

```
select vToyName,cCartId
from Toys,ShoppingCart
where Toys.cToyId = ShoppingCart.cToyId
```

结果 消息

	vToyName	cCartId
1	Robby the Whale	000001
2	Tie Dye Kit	000001
3	Alice in Wonderland	000001
4	Glamorous Doll	000002
5	Victorian Dollhouse	000002
6	Kitchen Set	000003
7	Super Deluge	000004
8	Racing Truck	000004
9	Robby the Whale	000005
10	Childrens Bedroom	000005
11	Key Boom Guitar	000005

十一

```
select left(vFirstName,1)+'.'+left(vLastName,1) as
Initials,vFirstName,vLastName
from Shopper
```

结果 消息

	Initials	vFirstName	vLastName
1	A.S	Angela	Smith
2	B.J	Barbara	Johnson
3	B.W	Betty	Williams
4	C.J	Carol	Jones
5	C.R	Catherine	Roberts
6	C.B	Charles	Brown
7	C.D	Christopher	Davis
8	C.M	Cynthia	Miller
9	D.W	Daniel	Wilson
10	D.M	David	Moore
11	D.T	Deborah	Taylor

十二

```
select cast(round(avg(mToyRate),0) as int) as 平均价格
from Toys
```

结果 消息

	平均价格
1	20

十三

```
select vFirstName,vLastName,vAddress,cCity
from Shopper
```

结果 消息

	vFirstName	vLastName	vAddress	cCity
1	Angela	Smith	16223 Radiance Court	Woodbridge
2	Barbara	Johnson	227 Beach Ave.	Sunnyvale
3	Betty	Williams	1 Tread Road	Virginia Beach
4	Carol	Jones	765 - Furling Road Apt 112	Boone
5	Catherine	Roberts	5508 Aquiline Court	San Jose
6	Charles	Brown	7822 S. Glitzy Avenue	Maitland
7	Christopher	Davis	4896 11th ST	Hill Avenue
8	Cynthia	Miller	98066 Weary Storm Street	Moon Park
9	Daniel	Wilson	4642 Peripheral Drive	Brecksville
10	David	Moore	8808 Joviality Drive	San Ramon
11	Deborah	Taylor	2199- Fairfax Drive	Libertyville

```
select vFirstName,vLastName,vAddress,cCity
from Recipient
```

结果 消息

	vFirstName	vLastName	vAddress	cCity
1	Barbara	Johnson	227 Beach Ave.	Sunnyvale
2	Catherine	Roberts	5508 Aquiline Court	San Jose
3	Christopher	Davis	4896 11th ST	Hill Avenue
4	Jennifer	Martin	9812 76th Street	Brooklyn
5	Barbara	Johnson	227 Beach Ave.	Sunnyvale
6	Donna	Anderson	7930 Orange St.	Las Vegas
7	Laura	Rodriguez	3242 Limestone	WayMarietta
8	Michelle	Hernandez	1353 Realm Lakes	Naperville
9	David	Moore	8808 Joviality Drive	San Ramon
10	Betty	Williams	1 Tread Road	Virginia Beach

十四

```
select vToyName
from Toys
where cToyId not in (select cToyId
                     from OrderDetail
                     where cWrapperId in (select cWrapperId
                                          from Wrapper))
```

结果 消息

	vToyName
1	Water Channel System
2	Parachute and Rocket
3	Super Deluge
4	Light Show Lamp
5	Glass Decoration
6	Tie Dye Kit
7	Glamorous Doll
8	Sleeping Beauty Doll
9	Pet Loving Doll
10	Flower Loving Doll
11	Nursery

十五

```
select cOrderNo,cToyId,mToyCost,mToyCost*siQty as 玩具总价值
from OrderDetail
order by cOrderNo
```

结果 消息

	cOrderNo	cToyId	mToyCost	玩具总价值
1	000001	000007	39.98	79.96
2	000001	000008	14.99	14.99
3	000002	000016	86.50	173.00
4	000003	000017	71.97	215.91
5	000004	000030	35.99	35.99
6	000005	000001	35.96	143.84
7	000005	000024	25.99	25.99
8	000005	000030	71.98	143.96
9	000006	000013	23.99	47.98
10	000006	000017	29.98	29.98
11	000007	000006	12.99	12.99

十六

```
select *
from Shopper
where cShopperId not in (select cShopperId
                         from Orders)
```

结果 消息

	cShopperId	cPassword	vFirstName	vLastName	vEmailId	vAddress	cCity	cState	cCountryId	cZipCode	cPhone	cCreditCardNo	vCre
1	000001	angels	Angela	Smith	angelas@qmail.com	16223 Radiance Court	Woodbridge	Virginia	001	22191	227-2344	6947343412896785	Mas
2	000004	credit	Carol	Jones	carolj@qmail.com	765 - Furling Road Apt 112	Boone	North Carolina	001	28607	678-4544	6344676854335436	Vis
3	000011	chubby	Deborah	Taylor	deboraht@qmail.com	2199- Fairfax Drive	Libertyville	Illinois	001	60048	889-2235	2345468798078563	Mas
4	000013	benhur	Dorothy	Thomas	dorthyt@speedmail.com	678 East 56th Street- #12	New York	New York	001	10009	696-2278	8765435456678754	Vis
5	000014	benjy	Elizabeth	Jackson	elizabethj@qmail.com	598 Apex Avenue #2	Saint Paul	Minnesota	001	55102	545-9078	4576544354567542	Vis
6	000015	sundance	Frances	Turner	francest@speedmail.com	2562 Eastwood	Denton	Texas	001	76205-5922	878-6670	5676879007565452	Mas
7	000016	bopeep	Helen	White	helenw@spedmail.com	Fleet Street	Point Pleasant	Pennsylvania	001	18950	585-7796	4564564564564565	Vis
8	000017	momma	James	Harris	helenw@spedmail.com	3456 Mt. Regale Drive	1509 Alexandria	Virginia	001	22303-2541	335-6678	4657567545344544	Mas
9	000018	grumpy	Jennifer	Martin	jenniferm@qmail.com	9812 76th Street	Brooklyn	Maryland	001	21225	569-7789	9775445343233443	Vis
10	000019	patch	Jessica	Thompson	jessicat@speedmail.com	565 Pebble St.	Arlington	Texas	001	76014	445-6797	9766545343233455	Mas

十七

```
delete
from Toys
where cBrandId in (select cBrandId
```

```
from ToyBrand
where cBrandName = 'Largo')
```

实验任务书（实验三、实验四）

实验报告要求：

1. 列出所有的 SQL 语句和源代码；
2. 程序要求有适当的注释；
3. 实验报告提交电子档。

实验内容（一）：存储过程与触发器

1. 创建一个称为 prcCharges 的存储过程，它返回某个定单号的装运费用和包装费用。
2. 创建一个称为 prcHandlingCharges 的过程，它接收定单号并显示经营费用。PrchandlingCharges 过程应使用 prcCharges 过程来得到装运费和礼品包装费。
   提示：经营费用=装运费+礼品包装费
3. 在 OrderDetail 上定义一个触发器，当向 OrderDetail 表中新增一条记录时，自动修改 Toys 表中玩具的库存数量（siToyQoh）。
4. Orders 表是 GlobalToyz 数据库里的一张核心的表，对这张表上做的任何更新动作（增、删、改）都需要记录下来，这是数据库审计（Audit）的基本思想。要求设计一张表存储对 Orders 表的更新操作，包括操作者、操作时间、操作类型、更新前的数据、更新后的数据。设计触发器实现对 Orders 表的审计。
5. 编写代码，分析玩具和地域的关系，例如哪个城市的购买者对哪一种、哪一类或哪一个品牌的玩具更有兴趣。这道题是个开放的题目，同学们可以按照自己的理解从不同的角度进行分析。实验报告中需给出代码、结果截图和对分析结果的文字描述。

实验内容（二）：视图、事务与游标

1. 定义一个视图，包括定单的编号、时间、金额以及收货人的姓名、国家代码和国家名称。
2. 基于（1）中定义的视图，查询所有国家代码为‘001’的收货人的姓名和他们所下定单的笔数及定单的总金额。
3. 视图定义如下：

```
CREATE VIEW vwOrderWrapper
AS
SELECT cOrderNo, cToyId, siQty, vDescription, mWrapperRate
FROM OrderDetail JOIN Wrapper
ON OrderDetail.cWrapperId = Wrapper.cWrapperId
```

执行以下更新命令并分析该命令的执行结果。

```
UPDATE vwOrderWrapper
SET siQty = 2, mWrapperRate = mWrapperRate + 1
WHERE cOrderNo = ‘000001’
```

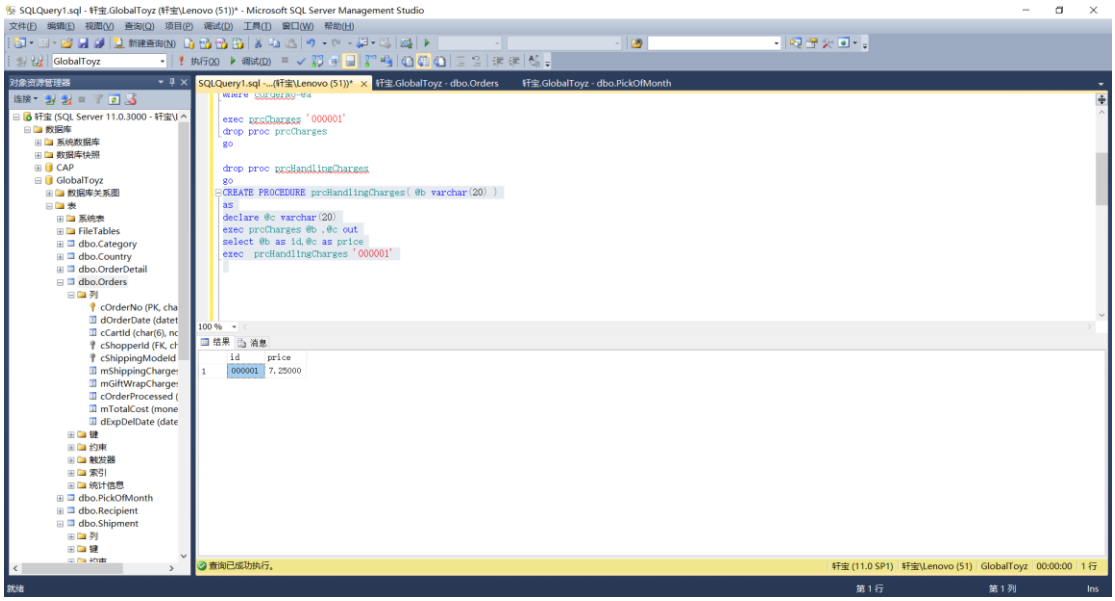
4. 在 GlobalToyz 数据库里创建一个用户，用户名为 user_xxxx（你的学号）。通过视图限制

该用户只能访问 Orders 表中 2017 年以前的数据。

5. 当购物者确认定单时，应该包含下面的步骤：
   （1）产生新的定单号（要求创建一个存储过程，用于产生新定单号）。
   （2）定单号，当前日期，购物车 ID，和购物者 ID 应该加到 Orders 表中。
   （3）定单号，玩具 ID 和数量应加到 OrderDetail 表中。
   （4）在 OrderDetail 表中更新玩具成本。（提示：Toy cost = Quantity * Toy Rate）。
   （5）从 ShoppingCart 表中将本次已购买的玩具删除。
   将上述步骤定义为一个事务。编写一个过程以购物车 ID 和购物者 ID 为参数，实现这个事务。（提示：首先需要修改表 ShoppingCart 的结构，在表中新增一个字段‘Status’。该字段取值为 1，表示该玩具为本次下订单时要购买的玩具，并产生一些模拟数据。）
6. 编写一个程序显示每天的定单状态。如果当天的定单值总合大于 150，则显示“High sales”，否则显示”Low sales”。要求列出日期、定单状态和定单总价值。（**要求用游标实现**）
7. 基于表 Orders 和 Shopper，以下列格式生成报表：

   购货人 ID　XXX　　购货人姓名　XXX
   购货人地址　XXXXXX
   定单号 XXX　定单时间 XXX　定单金额 XXX
   定单号 XXX　定单时间 XXX　定单金额 XXX

一.
```
create procedure prcCharges(@a varchar(20))
as
select mShippingCharges + mGiftWrapCharges as sum1
from Orders
where cOrderNo=@a
```
二
```
CREATE PROCEDURE prcHandlingCharges( @b varchar(20) )
as
declare @c varchar(20)
exec prcCharges @b ,@c out
select @b as id,@c as price
exec  prcHandlingCharges '000001'
```



三create trigger sitoyQohChange
on OrderDetail
for insert
as
declare @cToyid char(10)
declare @siQty int

select
@cToyid = inserted.cToyid,
@siQty = inserted.siQty
from inserted

update Toys
set siToyQoh = siToyQoh - @siQty
where cToyId = @cToyid

四

```
create table audit
(
 changetype char(15),
 changetime datetime,
 cOrderNo char(6),
 dOrderDate datetime,
 cCartId char(6),
 cShopperId char(6),
 cShippingModeId char(2),
 mShippingCharges money,
 mGiftWrapCharges money,
 cOrderProcessed char(1),
 mTotalCost money,
```

```
 dExpDelDate datetime
)


create trigger Orders_change
 on Orders
 for delete,insert,update
 as
 declare @insertcount int
 declare @deletecount int
 declare @changetype char(10)
 declare @changetime datetime
 declare @updatetype char(4)

 select @insertcount = count(*) from inserted
 select @deletecount = count(*) from deleted
 select @changetype =
 case
     when @insertcount0 and @deletecount>0
          then 'update'
     when @insertcount=0 and @deletecount>0
          then 'delete'
    else 'insert'
 end
 select @changetime = getdate()
 if @changeType='update'
   select @updateType='old'
   insert into
audit(changeype,changetime,cOrderNo,dOrderDate,cCartId,cShopperId,cSh
ippingModeId,
mShippingCharges,mGiftWrapCharges,cOrderProcessed,mTotalCost ,dExpDel
Date )
   select @changeType+@updateType,
@changeTime,cOrderNo,dOrderDate,cCartId,cShopperId,cShippingModeId,
mShippingCharges,mGiftWrapCharges,cOrderProcessed,mTotalCost ,dExpDel
Date  from deleted
 if @changeType='update'  select @updateType=' new'
 insert into
audit(changeype,changetime,cOrderNo,dOrderDate,cCartId,cShopperId,cSh
ippingModeId,
mShippingCharges,mGiftWrapCharges,cOrderProcessed,mTotalCost ,dExpDel
Date )
 select @changeType+@updateType,
@changeTime,cOrderNo,dOrderDate,cCartId,cShopperId,cShippingModeId,
```

```
mShippingCharges,mGiftWrapCharges,cOrderProcessed,mTotalCost ,dExpDel
Date  from inserted
```

五

```
create view view_(订单编号,时间,金额,n1,n2,c1,c2)
 as
 select
cOrderNo,dOrderDate,mTotalCost,vLastName,vFirstName,Shopper.cCountryI
d,cCountry
 from Orders,Shopper,Country
 where Orders.cShopperId= Shopper.cShopperId and Shopper.cCountryId =
Country.cCountryId
```

六

```
select n1,n2,count(*) as 订单笔数,sum(金额) as 总金额
 from view_
 group by n1,n2
```

六

修改影响多个基表，视图不可更新

七

```
create view user_8208180813
     as
     select *
     from Orders
     where Year(dOrderDate) < 2017
```

八

```
create procedure prcGenOrder
@OrderNo char(6) output
as
select @OrderNo=Max(cOrderNo) from  Orders
select @OrderNo=
      case
          when  @OrderNo>=0  and  @OrderNo<</span>9  Then
                        '00000'+Convert(char,@OrderNo+1)
          when  @OrderNo>=9  and  @OrderNo<</span>99  Then
                        '0000'+Convert(char,@OrderNo+1)
          when  @OrderNo>=99  and  @OrderNo<</span>999  Then
                        '000'+Convert(char,@OrderNo+1)
          when  @OrderNo>=999  and  @OrderNo<</span>9999  Then
                        '00'+Convert(char,@OrderNo+1)
          when  @OrderNo>=9999  and  @OrderNo<</span>99999  Then
                        '0'+Convert(char,@OrderNo+1)
          when  @OrderNo>=99999  Then  Convert(char,@OrderNo+1)
```

```
      end
   return


begin transaction
declare @count int
set  @count=0
declare @OrderNo char(6)
exec prcGenOrder @OrderNo output
set @count=@count+@@ERROR
declare @cCartId char(6)
declare @cShopperId char(6)
declare @ToyId char(6)
declare @SiQty char(6)
declare @mToyRate money
set @cCartId='000005'
set @cShopperId='000001'
set @ToyId ='000005'
set @SiQty =5
select @mToyRate=mToyRate from Toys
where cToyId=@ToyId
insert into Orders
values(@OrderNo,getdate(),@cCartId,@cShopperId,NULL,NULL,NULL,NULL,NU
LL,NULL)
set @count=@count+@@ERROR
insert into OrderDetail
values(@OrderNo,@ToyId,@SiQty,NULL,NULL,NULL,@mToyRate*@SiQty)
set @count=@count+@@ERROR
if(@count>0)
begin
print'发生错误'
rollback
end
else commit
select* from Orders
select* from OrderDetail
九
select distinct dOrderDate,SUM(mTotalCost) AS TotalCost
into OrderTemp
from Orders
group by dOrderDate;
select * from OrderTemp
declare @date varchar(20)
declare @cost money
```

```
declare youbiao
cursor for
select dOrderDate,TotalCost from OrderTemp
open youbiao
print'      日期          订单状态    订单总价值'
while (@@fetch_status=0)
begin
fetch youbiao into @date,@cost

 if(@cost>150)
   print @date+'    '+'High Sales'+'    '+cast(@cost as varchar(10))
 else
   print @date+'    '+'Low Sales'+'    '+cast(@cost as varchar(10))

end
close youbiao
deallocate youbiao
```