

1) DDL PET CLINIC

```
CREATE TABLE PetOwner(  
  OwnerId int PRIMARY KEY, Name char(50), Surname char(50),  
  StreetAddress char(100), City char(100),  
  State char(10), ZipCode int)
```

GO

```
CREATE TABLE Pet(  
  PetId char(10) PRIMARY KEY, Name char(20),  
  Kind char(10), Gender char(6), Age int,  
  OwnerId int)
```

GO

```
CREATE TABLE ProcedureHistory(  
  PetId char(10), ProcedureDate date,  
  ProcedureType char(50), Description char(100))
```

GO

```
ALTER TABLE PetOwner  
ADD CONSTRAINT state_michi  
DEFAULT 'Michigan' FOR State
```

GO

```
ALTER TABLE Pet  
ADD CONSTRAINT fh FOREIGN KEY(OwnerId) REFERENCES  
PetOwner(OwnerId)
```

GO

```
ALTER TABLE ProcedureHistory  
ADD CONSTRAINT fh1 FOREIGN KEY(PetId) REFERENCES  
Pet(PetId)
```

GO

Pet Clinic DRL

```
SELECT a.Name as [Pet Name], a.kind, b.Name as [Owner Name]  
FROM Pet a join PetOwner b ON a.OwnerId=b.OwnerId  
ORDER BY a.Name ASC
```

GO

```
SELECT a.PetID as petid, a.Name as NAME, a.Kind as kind, a.Gender as gender  
FROM Pet a join ProcedureHistory b  
ON a.PetID=b.PetId  
WHERE 1=(SELECT MONTH(b.ProcedureDate))  
ORDER BY a.PetID ASC
```

GO

Pet Store Procedure

```
CREATE PROCEDURE usp_PetProcedureDetails @ProcedureType
AS
SELECT p.PetID,p.Name,po.Name,ph.Description
from Pet p join PetOwner po on Pet.OwnerID=PetOwner.OwnerID
join ProcedureHistory ph on p.PetID=ph.PetID
go
```

2) DRL CLOTHING STORE

```
select FirstName+","+SurName as CustomerName
from Customers
where CustomerId in
(select CustomerId from (select Customers.CustomerId, count(Customers.CustomerId) count
from Customers
join Transactions
on Customers.CustomerId= Transactions.CustomerId
group by Customers.CustomerId) a
where a.count>1)
go
```

```
select Description, (RetailPrice - (RetailPrice*Discount)) as Cost_Price
from Clothing
go
```

Clothing Store DDL

```
Create Table Customers(CustomerId int primary key,
firstName varchar(20),
surName varchar(20),
customerState varchar(20))
go
```

```
create table Clothing(ItemCode bigint primary key,
Description varchar(100),
RetailPrice decimal(6,2),
Discount decimal (5,2))
go
```

```
create table Transactions (TransactionId int primary key,
TransactionDate datetime,
CustomerId int,
ItemCode bigint)
```

```
go
```

```
alter table Clothing  
add constraint unq unique(Description)  
go
```

```
alter table Transactions  
add constraint fk foreign key(CustomerId) references Customers(CustomerId)  
go
```

```
alter table Transactions  
add constraint fgk foreign key(ItemCode) references Clothing(ItemCode)  
go
```

Clothing Stored Store Procedure

```
Create proc usp_TransactionDetails  
as  
begin  
select a.FirstName,c.RetailPrice as TotalPrice from customers a join Transactions b  
on a.CustomerId=b.CustomerId  
join clothing c  
on b.ItemCode=c.ItemCode  
group by a.FirstName,c.RetailPrice  
end  
go
```

3) PROCEDURE BIKE STORE

```
CREAT proc usp_MaxNoOfProducts as  
Begin  
SELECT b.brand_name as Brand_Name, c.category_name as Category_Name, p.product_name  
as Product_Name  
FROM Brands b, Categories c, Products p  
WHERE p.brand_id IN( SELECT p.brand_id,count(p.brand_id)  
FROM Products p  
GROUP BY p.brand_id  
ORDER BY count(p.brand_id) desc,Limit 1)  
End  
GO
```

Bike Stored DDL

```
create table Categories(category_id int PRIMARY KEY,  
category_name varchar(255) NOT NULL)  
Go
```

```
create table Brands(brand_id int PRIMARY KEY,  
brand_name varchar(255) NOT NULL)  
Go
```

```
insert into Categories values(1,'Children Bicycles')  
insert into Categories values(2,'Comfort Bicycles')  
insert into Categories values(3,'Cruisers Bicycles')  
insert into Categories values(4,'Cyclocross Bicycles')  
Go
```

```
insert into Brands values(1,'Electra')  
insert into Brands values(2,'Haro')  
insert into Brands values(3,'Heller')  
insert into Brands values(4,'Pure Cycles')  
insert into Brands values(5,'Ritchey')  
Go
```

Bike Stored DRL

```
SELECT TOP 1 B.brand_name AS Brand_name From Brands B  
JOIN Products P ON B.brand_id=P.brand_id  
Order BY P.list_Price desc  
go
```

```
SELECT P.product_name AS Product_name,P.model_year AS Model_year FROM Products P  
JOIN Categories c ON p.Category_id=C.Category_id  
WHERE C.Category_name Like '%children%'  
Go
```

MOVIE DDL

```
create table Movies(  
id int Primary Key,  
Title varchar(100),  
Director varchar(100),  
Year int,  
Length_Minutes int)
```

```

GO
Create Table BoxOffice(
Movie_id int,
Rating float,
Domestic_sales bigint,
International_sales bigint)
GO
ALTER TABLE Movies ALTER COLUMN Title varchar(100) NOT NULL
GO
ALTER TABLE Movies ALTER COLUMN Director varchar(100) NOT NULL
GO
ALTER TABLE Movies ALTER COLUMN Year int NOT NULL
GO
ALTER TABLE Movies ALTER COLUMN Length_Minutes int NOT NULL
GO
ALTER TABLE Movies ADD CONSTRAINT UQ_Title UNIQUE(Title)
GO
ALTER TABLE BoxOffice
    ADD CONSTRAINT FK__Boxoffice_Movies
        FOREIGN KEY (Movie_id)
        REFERENCES Movies (id)
GO

```

MOVIE DRL

```

select m.Title, m.Director, m.Year, b.Rating from
Movies m
join BoxOffice b
on m.Id = b.Movie_id
where m.Year between 2000 and 2005 and b.Rating>5
go

```

```

select cast(concat(Title, 'in the Year ', Year) as varchar(117)) as [Movie and Release Year]
from Movies
go

```

Lodging DRL

```

SELECT room_id "Room Number", room_type_id "Room Type"
from room
where room_id NOT IN
( SELECT r.room_id from room r join booking b
on r.room_id = b.room_no) AND room_type_id != "single"

```

GO

```
select r2.room_type_id as "Room Type",
count(r1.room_no) as "No Of Bookings"
from room_type r2 left outer join room r on
(r.room_type_id=r2.room_type_id)
left outer join booking r1
on (r.room_id = r1.room_no)
group by r2.room_type_id
order by count(r1.room_no) desc
GO
```

Lodging DDL

```
CREATE TABLE room_type (room_type_id varchar(6) primary key,description varchar(100))
go
CREATE TABLE rate (room_type_id varchar(6) not null,occupancy int not null,amount
decimal(10,2) default 0)
go
CREATE TABLE room (room_id int primary key,room_type_id varchar(6),max_occupancy int)
go
alter table room add foreign key(room_type_id) references room_type(room_type_id)
go
alter table rate add constraint pk_name primary key (room_type_id,occupancy)
go
alter table rate add foreign key(room_type_id) references room_type(room_type_id)
go
```