1-1.Relation Schema



1-2. DDL

CREATE TABLE Professor(

pno INT NOT NULL,

pname VARCHAR2(30),

pmajor VARCHAR2(8),

pdept VARCHAR2(8),

CONSTRAINT pno PRIMARY KEY(pno)

);

CREATE TABLE Course(

cno VARCHAR2(8) NOT NULL,

cname VARCHAR2(30),

credit VARCHAR2(8),

sessions VARCHAR2(8),

CONSTRAINT cno PRIMARY KEY(cno)

);

CREATE TABLE Student(

sno INT NOT NULL,

pno INT,

sname VARCHAR2(30),

year INT,

dept VARCHAR2(8),

CONSTRAINT sno PRIMARY KEY(sno),

CONSTRAINT pro\_stu\_fk FOREIGN KEY(pno)

REFERENCES Professor(pno)

);

CREATE TABLE Enroll(

sno INT NOT NULL,

cno VARCHAR2(8) NOT NULL,

grade VARCHAR2(2),

exam INT,

CONSTRAINT cou\_en\_fk FOREIGN KEY(cno)

REFERENCES Course(cno),

CONSTRAINT stu\_en\_fk FOREIGN KEY(sno)

REFERENCES Student(sno),

CONSTRAINT en\_pk PRIMARY KEY(cno, sno)

);

CREATE TABLE Lecture(

cno VARCHAR2(8) NOT NULL,

pno INT NOT NULL,

lec\_time VARCHAR2(15),

room VARCHAR2(8),

CONSTRAINT cou\_lec\_fk FOREIGN KEY(cno)

REFERENCES Course(cno),

CONSTRAINT pro\_lec\_fk FOREIGN KEY(pno)

REFERENCES Professor(pno),

CONSTRAINT lec\_pk PRIMARY KEY(cno, pno)

);

1-3. 데이터 입력 명령어

LOAD DATA

INFILE 'c:\Professor.csv'

APPEND INTO TABLE Professor

FIELDS TERMINATED BY ','

(pno, pname, pmajor, pdept)

LOAD DATA

INFILE 'C:\Course.csv'

APPEND INTO TABLE Course

FIELDS TERMINATED BY ','

(cno, cname, credit, sessions)

LOAD DATA

INFILE 'C:\Student.csv'

APPEND INTO TABLE Student

FIELDS TERMINATED BY ','

(sno,sname,year,dept,pno)

LOAD DATA

INFILE 'C:\Enroll.csv'

APPEND INTO TABLE Enroll

FIELDS TERMINATED BY ','

(sno,cno,grade,exam)

LOAD DATA

INFILE 'C:\Lecture.csv'

APPEND INTO TABLE Lecture

FIELDS TERMINATED BY ','

(cno,pno,lec\_time,room)

1-4 질의 명령어

1.

a)

SELECT sname FROM Student WHERE dept='CE';

b)

SELECT cname FROM Course c WHERE cno in (SELECT cno FROM Lecture l INNER JOIN Professor p ON l.pno=p.pno WHERE p.pname='Chun');

c)

SELECT DISTINCT sname, dept

FROM Student

where sno in

(SELECT DISTINCT sno

FROM enroll

where sno = Student.sno and not(grade>'A'));

d)

SELECT DISTINCT sname, dept

FROM student NATURAL JOIN enroll

WHERE grade<>'A';

e)

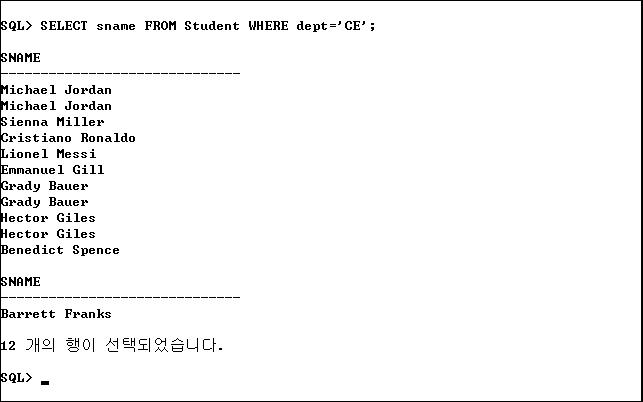
Delete from Lecture Where pno in

(Select pno From Professor Where pdept = 'CS');

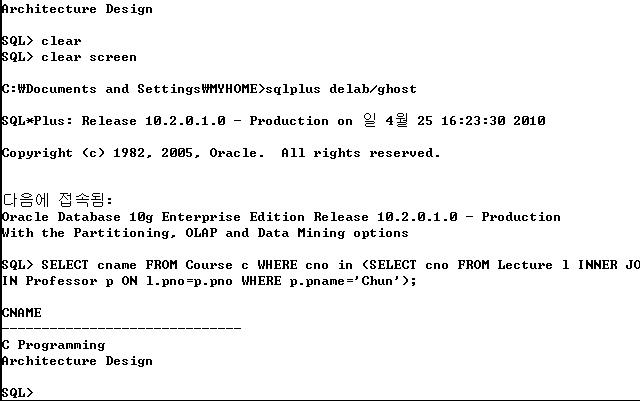
Delete from Enroll Where cno LIKE 'CS%';

Delete from Course Where cno LIKE 'CS%';

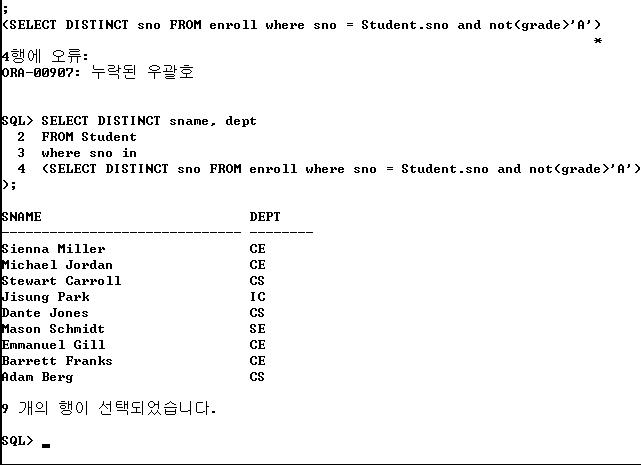
1-1-a



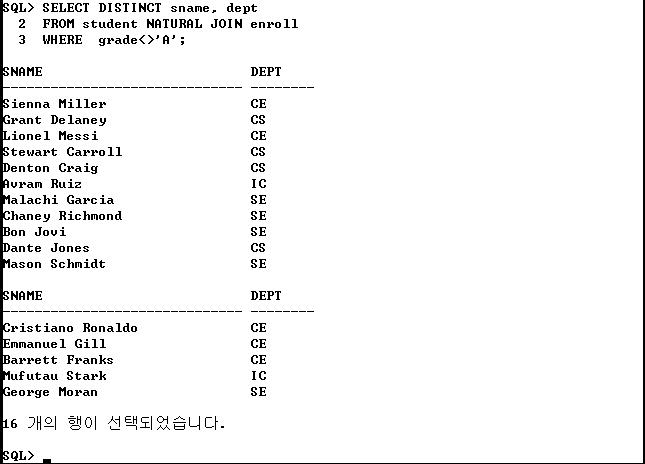
1-1-b.



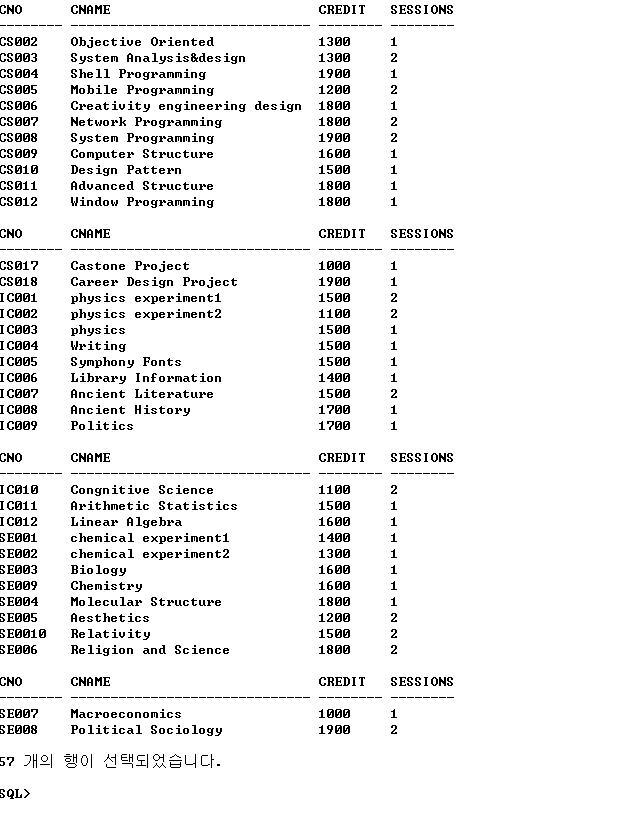
1-1-c



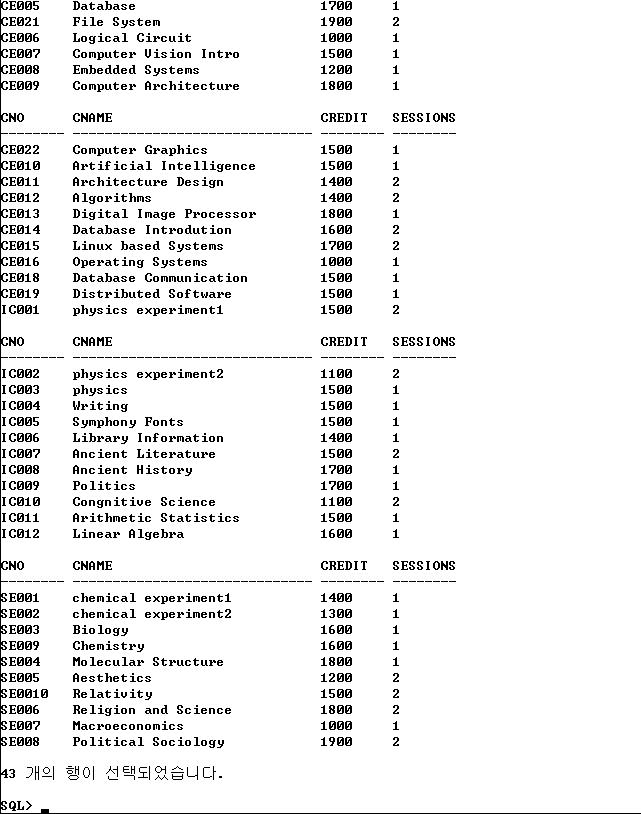
1-1-d



1-1-e



쿼리문 실행후



2-1 . Relation Schema



2-2. DDL

CREATE TABLE Department(

dnumber INT NOT NULL,

dname VARCHAR2(25),

mgrstartdate VARCHAR2(15),

dlocation VARCHAR2(30),

CONSTRAINT d\_num PRIMARY KEY(dnumber)

);

CREATE TABLE Project(

pnumber INT NOT NULL,

pname VARCHAR2(15),

plocation VARCHAR2(30),

CONSTRAINT p\_num PRIMARY KEY(pnumber)

);

CREATE TABLE Employee(

ssn INT NOT NULL,

fname VARCHAR2(15),

lname VARCHAR2(15),

bdate VARCHAR2(12),

address VARCHAR2(30),

sex VARCHAR2(6),

salary INT,

superssn INT,

dno INT,

CONSTRAINT ssn PRIMARY KEY(ssn),

CONSTRAINT dno FOREIGN KEY(dno)

REFERENCES Department(dnumber)

);

CREATE TABLE Works\_on

(

ssn INT NOT NULL,

pnumber INT NOT NULL,

hours INT,

CONSTRAINT pro\_work FOREIGN KEY(pnumber)

REFERENCES Project(pnumber),

CONSTRAINT emp\_work FOREIGN KEY(ssn)

REFERENCES Employee(ssn),

CONSTRAINT work\_pk PRIMARY KEY(ssn, pnumber)

);

2-3 데이터 입력 명령어

LOAD DATA

INFILE 'C:\Department.csv'

APPEND INTO TABLE Department

FIELDS TERMINATED BY ','

(dnumber,dname, mgrstartdate,dlocation)

LOAD DATA

INFILE 'C:\Project.csv'

APPEND INTO TABLE Project

FIELDS TERMINATED BY ','

(pnumber,pname,plocation)

LOAD DATA

INFILE 'C:\Employee.csv'

APPEND INTO TABLE Employee

FIELDS TERMINATED BY '.'

(ssn,fname,lname,bdate,address,sex,salary,superssn,dno)

LOAD DATA

INFILE 'C:\Work\_on.csv'

APPEND INTO TABLE Work\_on

FIELDS TERMINATED BY ','

(ssn,pnumber,hours)

2-4. 질의 명령어

2.

a)

Select fname,lname FROM Employee WHERE ssn in

(Select ssn FROM Works\_on w INNER JOIN Project p ON w.pnumber=p.pnumber WHERE hours>=10 AND pname='X');

b)

Select fname,lname FROM Employee WHERE superssn =

(SELECT ssn FROM Employee WHERE fname='Jonghoon' AND lname='Chun');

c)

Select pname,sum(hours)

FROM Project p INNER JOIN Works\_on w

ON p.pnumber=w.pnumber

GROUP BY pname;

d)

Select fname,lname

From Employee

Where ssn in

(Select ssn

FROM Project p INNER JOIN Works\_on w

ON p.pnumber=w.pnumber);

e)

SELECT e.fname,e.lname

FROM Employee e

WHERE e.ssn not in

(Select ssn FROM Works\_on WHERE e.ssn=ssn);

f)

SELECT dname,avg(e.salary)

FROM Employee e INNER JOIN Department d

ON e.dno = d.dnumber

GROUP BY dname;

g)

SELECT avg(e.salary)

FROM Employee e

Where e.sex='F';

h)

Select fname,lname,address

From Employee

WHERE address='SEOUL'

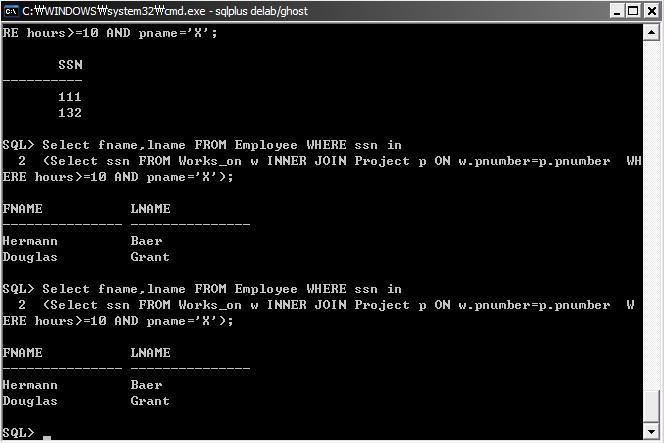
AND dno not in

(Select dnumber

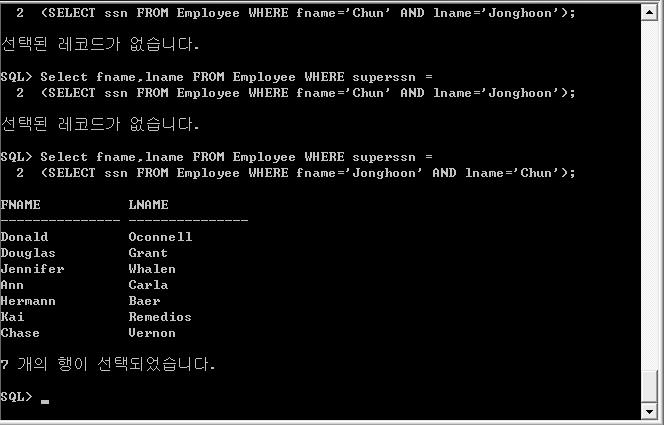
FRom Department

Where dlocation='SEOUL');

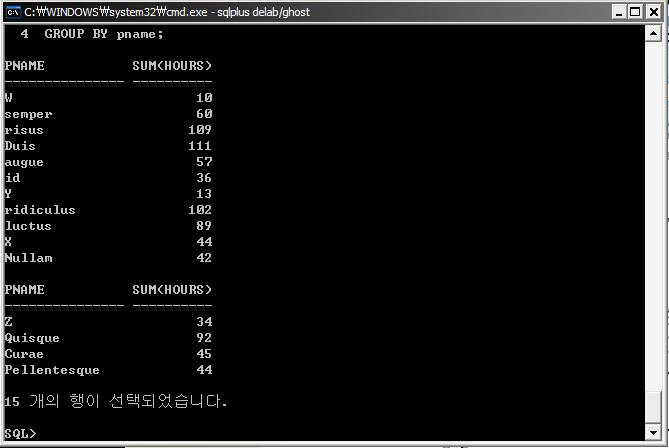
2-a.



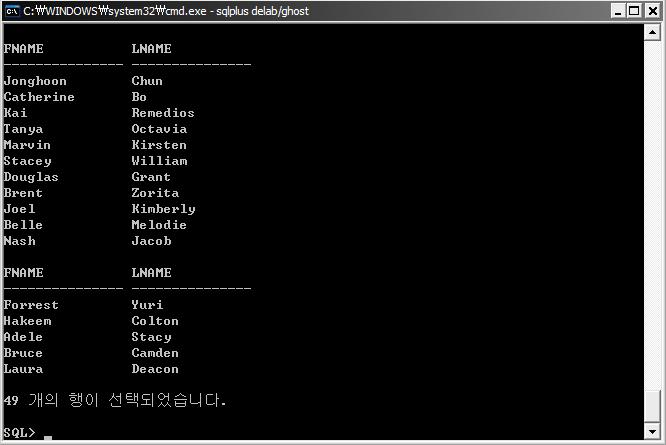
2-b.



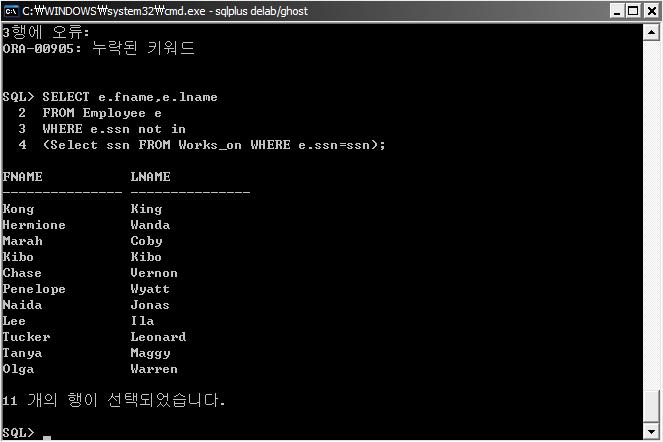
2-c.



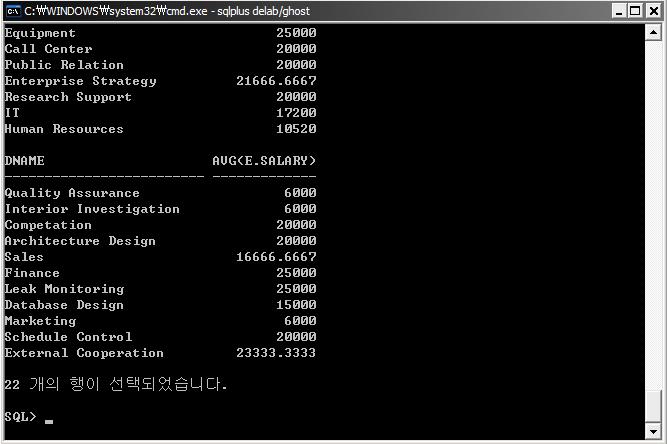
2-d.



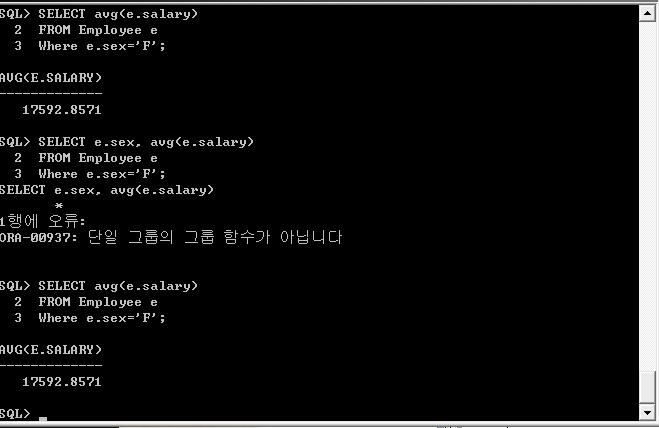
2-e.



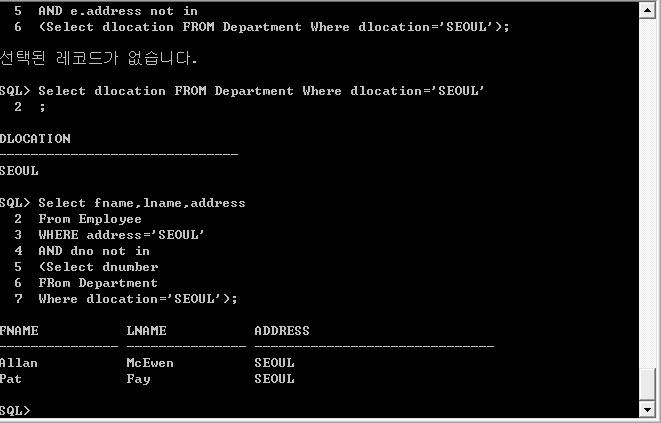
2-f.



2-g.



2-h.



3-1.relation schema



3-2.DDL

CREATE TABLE Warehouse(

code VARCHAR2(15) NOT NULL,

address VARCHAR2(30),

phone VARCHAR2(20),

CONSTRAINT ware\_pk PRIMARY KEY(code)

);

CREATE TABLE Author(

name VARCHAR2(55) NOT NULL,

address VARCHAR2(35) NOT NULL,

url VARCHAR2(40),

CONSTRAINT author\_pk PRIMARY KEY(name, address)

);

CREATE TABLE Publisher(

name VARCHAR2(25) NOT NULL,

address VARCHAR2(35),

phone VARCHAR2(15),

url VARCHAR2(40),

CONSTRAINT pub\_pk PRIMARY KEY(name)

);

CREATE TABLE Customer(

email VARCHAR2(30) NOT NULL,

name VARCHAR2(30),

address VARCHAR2(30),

phone VARCHAR2(20),

CONSTRAINT cus\_pk PRIMARY KEY(email)

);

CREATE TABLE Book(

ISBN VARCHAR2(15) NOT NULL,

title VARCHAR2(80),

year INT,

price INT,

CONSTRAINT book\_pk PRIMARY KEY(ISBN)

);

CREATE TABLE Stocks(

ISBN VARCHAR2(15) NOT NULL,

code VARCHAR2(15) NOT NULL,

num INT,

CONSTRAINT book\_stock\_fk FOREIGN KEY(ISBN)

REFERENCES Book(ISBN),

CONSTRAINT ware\_stock\_fk FOREIGN KEY(code)

REFERENCES Warehouse(code),

CONSTRAINT stock\_pk PRIMARY KEY(ISBN, code)

);

CREATE TABLE Written\_by(

name VARCHAR2(55) NOT NULL,

address VARCHAR2(35) NOT NULL,

ISBN VARCHAR2(15) NOT NULL,

CONSTRAINT aut\_wri\_fk FOREIGN KEY(name,address)

REFERENCES Author(name,address),

CONSTRAINT book\_wri\_fk FOREIGN KEY(ISBN)

REFERENCES Book(ISBN),

CONSTRAINT wri\_pk PRIMARY KEY(name, address, ISBN)

);

3-3. 데이터 입력 명령어

LOAD DATA

INFILE 'C:\Warehouse.csv'

APPEND INTO TABLE Warehouse

FIELDS TERMINATED BY ','

(code,address,phone)

LOAD DATA

INFILE 'C:\Author.csv'

APPEND INTO TABLE Author

FIELDS TERMINATED BY ','

(name,address,url)

LOAD DATA

INFILE 'C:\Publisher.csv'

APPEND INTO TABLE Publisher

FIELDS TERMINATED BY ','

(name,address,phone,url)

LOAD DATA

INFILE 'C:\Customer.csv'

APPEND INTO TABLE Customer

FIELDS TERMINATED BY ','

(email,name,address,phone)

LOAD DATA

INFILE 'C:\Book.csv'

APPEND INTO TABLE Book

FIELDS TERMINATED BY ','

(ISBN,title,year,price)

LOAD DATA

INFILE 'C:\Stocks.csv'

APPEND INTO TABLE Stocks

FIELDS TERMINATED BY ','

(ISBN,code,num)

LOAD DATA

INFILE 'C:\Written\_by.csv'

APPEND INTO TABLE Written\_by

FIELDS TERMINATED BY ','

(name,address,ISBN)

LOAD DATA

INFILE 'Published\_by.csv'

APPEND INTO TABLE Published\_by

FIELDS TERMINATED BY ','

(name,ISBN)

LOAD DATA

INFILE 'C:\Shopping\_basket.csv'

APPEND INTO TABLE Shopping\_basket

FIELDS TERMINATED BY ','

(basketid)

LOAD DATA

INFILE 'C:\Basket\_of.csv'

APPEND INTO TABLE Basket\_of

FIELDS TERMINATED BY ','

(email,basketid)

LOAD DATA

INFILE 'C:\Contains.csv'

APPEND INTO TABLE Contains

FIELDS TERMINATED BY ','

(basketid,ISBN,num)

3-4 Sql 질의 명령어

a)

Select title

FROM Written\_by w INNER JOIN Book b

ON w.ISBN = b.ISBN

WHERE w.name='Rolling J.K' AND b.year=2010;

b)

Select title

FROM Book

WHERE price>=10000 AND ISBN in

(Select w.ISBN FROM Written\_by w INNER JOIN Published\_by p

ON w.ISBN = p.ISBN

WHERE w.name='C%' AND p.name='Harry');

c)

Select title

FROM Book b INNER JOIN Written\_by w

ON b.ISBN=w.ISBN

WHERE w.name='brown T.I' AND b.ISBN in

(

select ISBN

From Stocks

group by ISBN

having sum(num)>1000

);

d)

Select title, price

FROM Contains c INNER JOIN Book b

ON c.ISBN = b.ISBN

WHERE c.basketid='CCC003';

e)

Select title

FROM Book

WHERE ISBN in

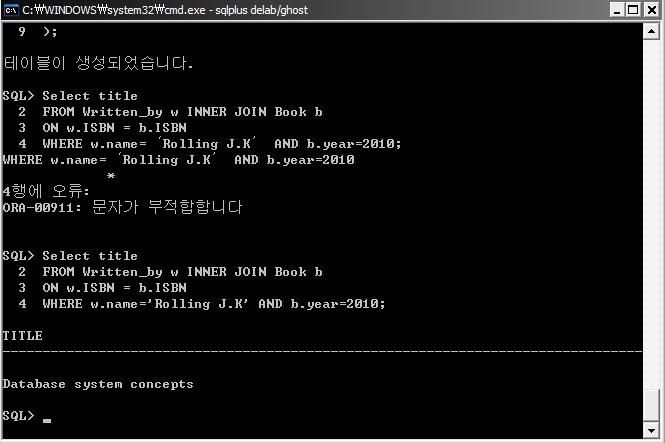
(Select ISBN

FROM Contains c INNER JOIN Basket\_of b

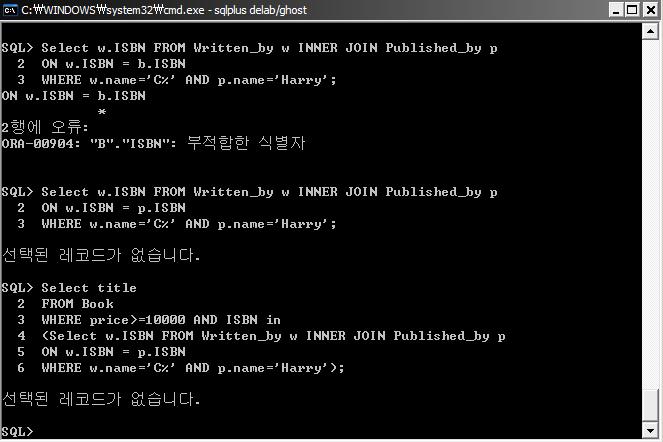
ON b.basketid=c.basketid

WHERE c.ISBN in (Select ISBN FROM Stocks WHERE num=0) AND b.email in (Select email FROM Customer WHERE Customer.name='Cristal'));

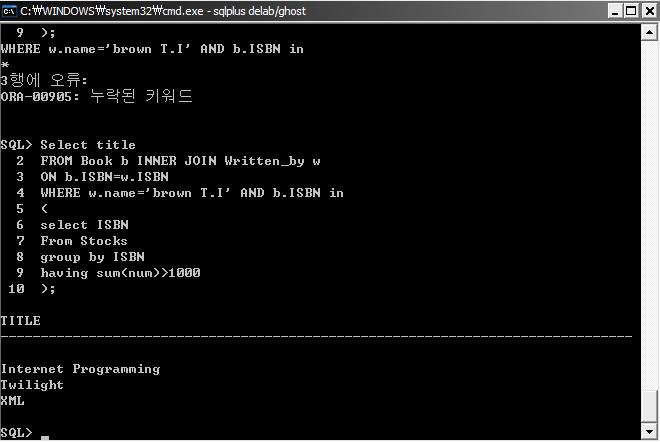
3-a.



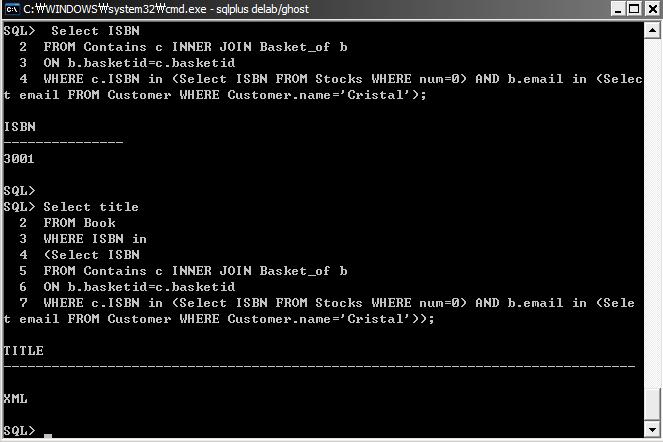
3-b.



3-c.



3-d.



3-e.

