###### **­Cairo University**

**Faculty of Computers and Information Information Systems Department Database Systems 1**

**Model Answer of Questions Booklet**

**Part 1: DDL Exercises**

**Question-1:**

**Consider the following schema named University:**

Student (SSN, Name, City, Age, Major)

Course (CrsCode, Name)

Registered (SSN, CrsCode, Semester, Year)

Department (DeptCode, Name)

* Write SQL DDL statements for declaring the University schema and relations. Specify appropriate keys and referential integrity constraints.

CREATE database University

CREATE TABLE University.Student

(

SSN INT NOT NULL,

Name VARCHAR(45),

Age INT,

City VARCHAR(15),

PRIMARY KEY (SSN),

Major VARCHAR(5) FOREIGN KEY REFERENCES Department (DeptCode)

);

CREATE TABLE University.Course

(

CrsCode SMALLINT NOT NULL,

Name VARCHAR(45),

PRIMARY KEY (CrsCode)

);

CREATE TABLE University.Department

(

DeptCode VARCHAR(5) NOT NULL,

Name VARCHAR(15) NOT NULL,

PRIMARY KEY (DeptCode)

);

CREATE TABLE University.Registered

(

SSN INT NOT NULL FOREIGN KEY REFERENCES Student (SSN),

CrsCode SMALLINT NOT NULL FOREIGN KEY REFERENCES Course (CrsCode), Semester VARCHAR(45) NOT NULL,

Year VARCHAR (45),

PRIMARY KEY (SSN, CrsCode)

);

* Create a table containing students’ names majoring on CS.

CREATE TABLE CS\_Student AS (select \* from Student where Major=’CS’)

---------------------------------------------------------------------------------------------------------------------

**Question-2:**

* Create table Customer which contains CustomerID as a primary key, Customer name, Age (should be not less than 18), Salary, City and the default value of the City should be ‘Cairo’.

CREATE TABLE CUSTOMER

(

ID INT NOT NULL,

NAME VARCHAR (20) NOT NULL,

AGE INT NOT NULL CHECK (AGE >= 18),

SALARY INT,

City VARCHAR (25) DEFAULT 'Cairo',

PRIMARY KEY (ID)

);

* Add new Column called Customer Address.

ALTER TABLE CUSTOMER

ADD ADDRESS char (25)

* Change the data type of the Column Address to Varchar (30).

ALTER TABLE CUSTOMER

ALTER COLUMN ADDRESS Varchar (30)

* Drop the created table Customer.

DROP TABLE CUSTOMER

---------------------------------------------------------------------------------------------------------------------

**Question-3:**

* Create table Employee (id, name, dept, age, salary, location).

CREATE TABLE employee  
( id int,  
name char(20),  
dept char(10),  
age int,  
salary int,  
location char(10), PRIMARY KEY (id)  
);

* Add a column "experience" to the employee table.

ALTER TABLE employee

ADD experience int;

* Drop the column "location" from the employee table.

ALTER TABLE employee

DROP column location;

* Modify the column salary in the employee table.

ALTER TABLE employee

Alter column salary float;

* Change the name of the table employee to my\_employee.

RENAME employee TO my\_emloyee;

---------------------------------------------------------------------------------------------------------------------

**Question-4:**

* Create an employee table with Primary Key constraint.

CREATE TABLE employee  
( id int PRIMARY KEY,  
name char(20),  
dept char(10),  
age int,  
salary int,  
location char(10)  
);

---------------------------------------------------------------------------------------------------------------------

**Question-5:**

* Create table product (product\_id, product\_name, supplier\_name, unit\_price).

CREATE TABLE product  
( product\_id int PRIMARY KEY,  
product\_name char(20),  
supplier\_name char(20),  
unit\_price int  
);

* Create table order\_items (order\_id, product\_id, supplier\_name, unit\_price).

CREATE TABLE order\_items  
( order\_id int PRIMARY KEY,

product\_id int,  
supplier\_name char(20),  
unit\_price int  
);

* Add foreign key constraint for order\_items.product\_id.

CREATE TABLE order\_items  
( order\_id int PRIMARY KEY,  
product\_id int FOREIGN KEY REFERENCES product(product\_id),  
supplier\_name char(20),  
unit\_price int  
); ---------------------------------------------------------------------------------------------------------------------

**Question-6:**

* Create employee table (id, name, dept, age, salary, location) and name of the department is not null.

CREATE TABLE employee  
( id int primary key,  
name char(20) NOT NULL,  
dept char(10),  
age int,  
salary int,  
location char(10)  
);

---------------------------------------------------------------------------------------------------------------------

**Question-7:**

* Create an employee table with the location should be Unique key.

CREATE TABLE employee  
( id int PRIMARY KEY,  
name char(20),  
dept char(10),  
age int,  
salary int,  
location char(10) UNIQUE  
);

---------------------------------------------------------------------------------------------------------------------

**Question-8:**

* Create table employee and the gender attribute should accept ‘M’ or 'F’.

CREATE TABLE employee  
( id int PRIMARY KEY,  
name char(20),  
dept char(10),  
age int,  
gender char(1) CHECK (gender in ('M','F')),  
salary int,  
location char(10)  
);

---------------------------------------------------------------------------------------------------------------------

**Part 2: DML Exercises**

**Question-1:**

* Given the following schema:

Movie (title, year, length, inColor, studioName, producerC#)

StarsIn (movieTitle, movieYear, starName)

MovieStar (name, address, gender, birthdate)

MovieExec (name, address, cert#, netWorth)

Studio (name, address, presC#)

1. Find the address of MGM studios.

SELECT address

FROM studio

WHERE name = ‘MGM’;

1. Find Sandra Bullock’s birthdate.

SELECT birthdate

FROM moviestar

WHERE name = ‘Sandra Bullock’;

1. Find all the stars that appear either in a movie made in 1980 or a movie with “Love” in the title.

SELECT starName

FROM StarsIn

WHERE movieYear = 1980

OR movieTitle LIKE ‘%Love%’;

1. Find all executives worth at least $10,000,000.

SELECT name

FROM MovieExec

WHERE netWorth >= 10,000,000;

1. Find all the stars who either are male or live in Miami (have Miami as a part of their address).

SELECT name

FROM MovieStar

WHERE gender = ‘M’

OR address LIKE ‘%Miami %’

1. Who were the male stars in Terms of Endearment.

SELECT name

FROM MovieStar, StarsIn

WHERE gender = ‘M’

AND name = starName

AND movieTitle = ‘Terms of Endearment’;

1. Which stars appeared in movies produced by MGM in 1995?

SELECT starName

FROM MovieStar, Movie

WHERE title = movieTitle

AND year = movieYear

AND year = 1995

AND studioName = ‘MGM’;

1. Which movies are longer than *Gone With the Wind*?

SELECT M1.title

FROM Movie AS M1, Movie AS M2

WHERE M2.title = ‘Gone With the Wind’

AND M1.length > M2.length;

1. Which executives are worth more than *Merv Griffin*?

SELECT M1.name

FROM MovieExec AS M1, MovieExec AS M2

WHERE M2.name = ‘Mery Griffin’

AND M1.networth > M2.networth;

---------------------------------------------------------------------------------------------------------------------

**Question-2:**

* Given the following schema:

Classes (class, type, country, numGuns, bore, displacement)

Ships (name, class, launched)

Battles (name, date)

Outcomes (ship, battle, result)

1. Find the countries whose ships had the largest number of guns.

SELECT country

FROM classes

WHERE numGuns = (SELECT MAX(numGuns)

FROM classes);

1. Find the classes of ships at least one of which was sunk in a battle.

SELECT DISTINCT class

FROM Ships

WHERE name IN (SELECT ship

FROM Outcomes

WHERE result = ‘sunk’);

1. Find the names of the ships with a 16-inch bore.

SELECT name

FROM ships

WHERE class IN (SELECT class

FROM classes

WHERE bore = 16);

1. Find the battles in which ships of the Kongo class participated.

SELECT DISTINCT battle

FROM ships, outcomes

WHERE name = ship

AND class = ‘Kongo’;

1. Find the names of the ships whose number of guns was the largest for those ships of the same bore.

SELECT name

FROM ships, classes AS C1

WHERE ships.class = C1.class

AND numGuns = (SELECT MAX(numGuns)

FROM classes AS C2

WHERE C1.bore = C2.bore);

1. Find the number of battleship classes.

SELECT count(\*)

FROM classes

WHERE type = ‘bc’;

1. Find the average number of guns of battleship classes.

SELECT avg(numGuns)

FROM classes

WHERE type = ‘bc’;

1. Find the average of guns of battleships.

SELECT avg(numGuns)

FROM ships, classes

WHERE ships.class =classes.class

AND type = ‘bc’;

1. Find for each class the year in which the first ship of that class was launched.

SELECT class, launched

FROM ships AS S1

WHRE launched <= ALL (SELECT year

FROM ships AS S2

WHERE S2.class = S1.class);

1. Find for each class the number of ships of that class sunk in battle.

SELECT classes.class, count(\*)

FROM classes, ships, outcomes

WHERE classes.class = ships.class

AND ship = name

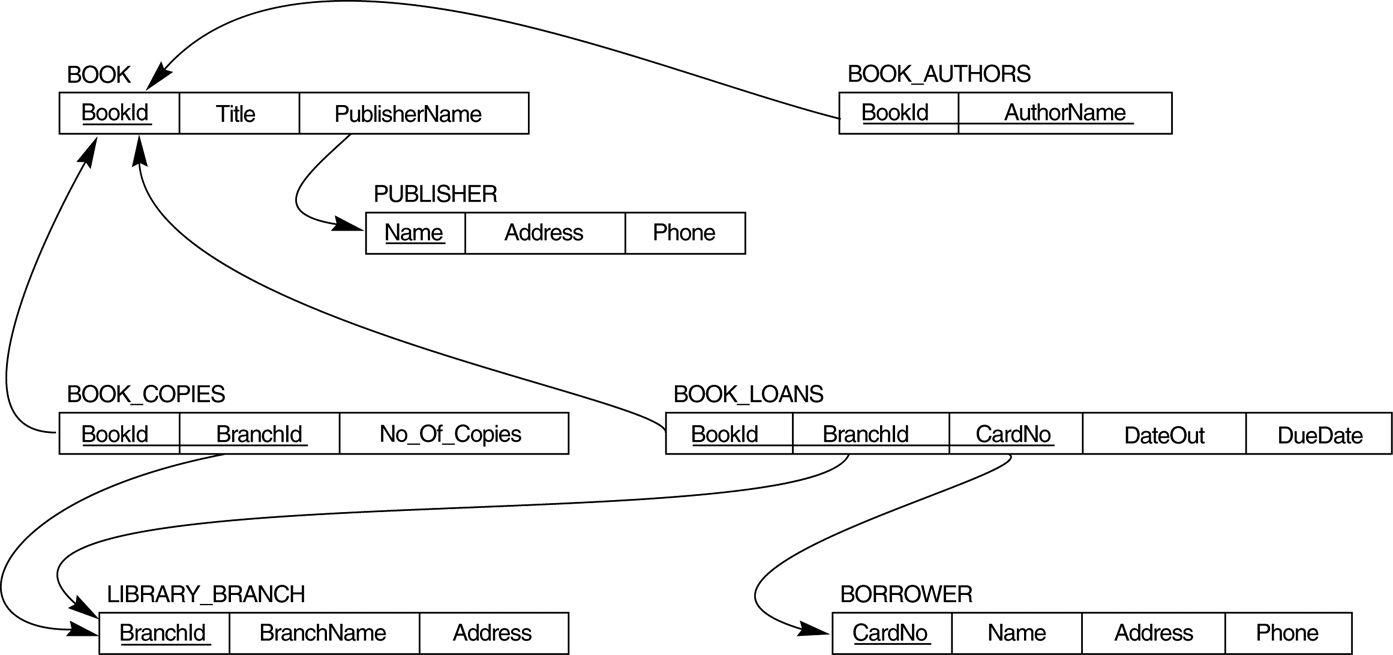
AND result = ‘sunk’

GROUP BY classes.class;

---------------------------------------------------------------------------------------------------------------------

**Question-3:**

Given a database schema for a library management system as follows:



1. How many copies of the book titled The Lost Tribe are owned by the library branch whose name is "Sharpstown"?

SELECT bc.No\_Of\_Copies

FROM BOOK b, BOOK\_COPIES bc, LIBRARY\_BRANCH bl

WHERE b.BookId = bc.BookId

AND bc.BranchId = bl.BranchId

AND Title='The Lost Tribe' AND BranchName='Sharpstown';

1. How many copies of the book titled The Lost Tribe are owned by each library branch?

SELECT BranchName, No\_Of\_Copies

FROM ((BOOK NATURAL JOIN BOOK\_COPIES) NATURAL JOIN LIBRARY\_BRANCH )

WHERE Title='The Lost Tribe';

1. Retrieve the names of all borrowers who do not have any books checked out.

SELECT Name

FROM BORROWER B

WHERE CardNo NOT IN (SELECT CardNo

FROM BOOK\_LOANS );

1. For each book that is loaned out from the "Sharpstown" branch and whose DueDate is today, retrieve the book title, the borrower's name, and the borrower's address.

SELECT B.Title, R.Name, R.Address

FROM BOOK B, BORROWER R, BOOK\_LOANS BL, LIBRARY\_BRANCH LB

WHERE LB.BranchName='Sharpstown'

AND LB.BranchId=BL.BranchId

AND BL.DueDate='today'

AND BL.CardNo=R.CardNo

AND BL.BookId=B.BookId

1. For each library branch, retrieve the branch name and the total number of books loaned out from that branch.

SELECT L.BranchName, COUNT(\*)

FROM LIBRARY\_BRANCH L, BOOK\_LOANS BL

WHERE BL.BranchId = L.BranchId

GROUP BY L.BranchName;

1. Retrieve the names, addresses, and number of books checked out for all borrowers who have more than five books checked out.

SELECT B.Name, B.Address, COUNT(\*)

FROM BORROWER B, BOOK\_LOANS L

WHERE B.CardNo = L.CardNo

GROUP BY B.CardNo, B.Name, B.Address

HAVING COUNT(\*) > 5;

1. For each book authored (or co-authored) by "Stephen King", retrieve the title and the number of copies owned by the library branch whose name is "Central".

SELECT Title, No\_Of\_Copies

FROM (((BOOK\_AUTHORS NATURAL JOIN BOOK) NATURAL JOIN BOOK\_COPIES) NATURAL JOIN LIBRARY\_BRANCH)

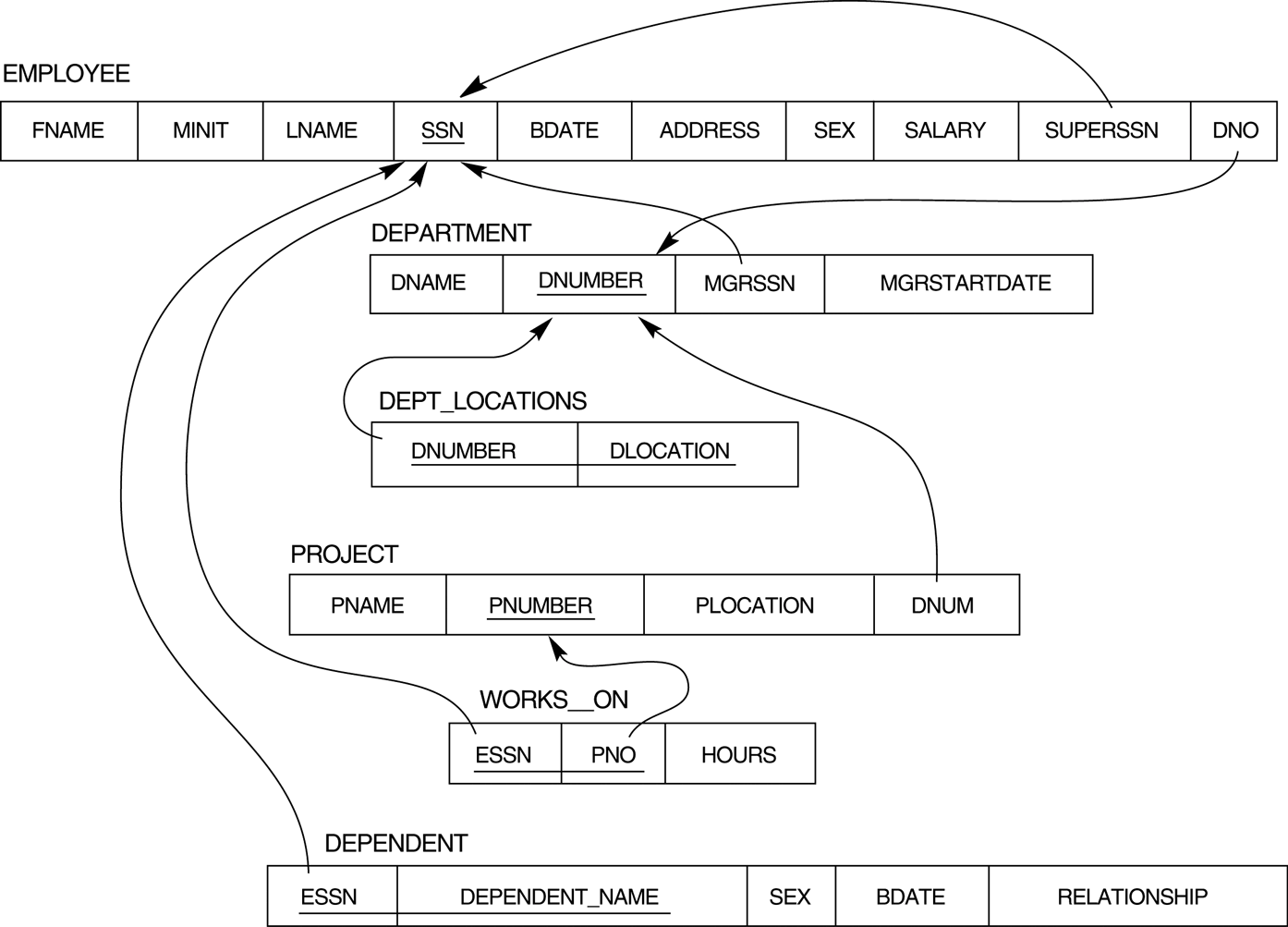
WHERE Author\_Name='Stephen King'

AND BranchName='Central';

---------------------------------------------------------------------------------------------------------------------

**Question-4:**

Given a database schema of a company as follows:



1. Retrieve the names of employees in department 5 who work more than 10 hours per week on the 'ProductX' project.

SELECT LNAME, FNAME

FROM EMPLOYEE, WORKS\_ON, PROJECT

WHERE DNO=5

AND SSN=ESSN

AND PNO=PNUMBER

AND PNAME='ProductX'

AND HOURS>10;

1. For each project, list the project name and the total hours per week (by all employees) spent on that project.

SELECT PNAME, SUM (HOURS)

FROM PROJECT, WORKS\_ON

WHERE PNUMBER=PNO

GROUP BY PNAME;

1. Retrieve the names of employees who work on every project.

SELECT E.LNAME, E.FNAME

FROM EMPLOYEE E

WHERE NOT EXISTS (SELECT PNUMBER

FROM PROJECT

WHERE PNUMBER NOT IN (SELECT PNO

FROM WORKS\_ON

WHERE ESSN=E.SSN));

1. Retrieve the names of employees who do not work on any project.

SELECT LNAME, FNAME

FROM EMPLOYEE

WHERE SSN NOT IN (SELECT ESSN

FROM WORKS\_ON);

1. Find the names and addresses of employees who work on at least one project located in Houston but whose department has no location in Houston.

SELECT LNAME, FNAME, ADDRESS

FROM EMPLOYEE

WHERE EXISTS (SELECT \*

FROM WORKS\_ON W, PROJECT P, DEPT\_LOCATIONS DL

WHERE W.PNO = P.PNUMBER

AND P.DNUM = DL.DNUM

AND DL.DLOCATION <> ‘Houston’);

1. List the last names of department managers who have no dependents.

SELECT E.LNAME, E.FNAME

FROM EMPLOYEE E, DEPARTMENT D

WHERE E.SSN = D.MRGSSN

AND NOT EXISTS (SELECT DEPENDENT\_NAME

FROM DEPENDENT

WHERE ESSN=E.SSN)

1. Find details of those employees whose salary is > the average salary for all employees. Output salary in descending order.

SELECT \*

FROM Employee

WHERE Salary > (SELECT AVG (Salary)

FROM Employee)

ORDER BY Salary DESC;

1. Find details of those employees whose salary is > the average salary for all employees in his/her department. Output salary in ascending order.

SELECT E.\*

FROM EMPLOYEE E, (SELECT DNO, AVG (SALARY) AS LTB

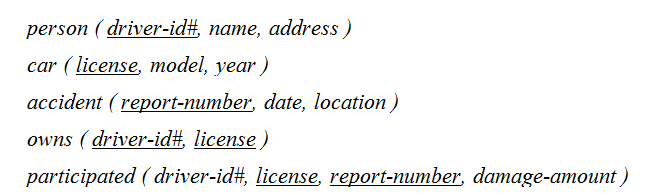
FROM EMPLOYEE

GROUP BY DNO) AS A

WHERE E.DNO = A.DNO

AND E.SALARY > LTB;

---------------------------------------------------------------------------------------------------------------------**Question-5:**

Given the following schema:

|  |
| --- |
|  |
|  |
|  |
|  |
|  |

1. Find the total number of people who owned cars that were involved in accidents in 1989.

Select count (distinct name)

From accident, participated, person

Where accident.report-number = participated.report-number

and participated.driver-id = person.driver-id

and date between date ’1989-00-00’

and date ‘1989-12-31’

1. Find the number of accidents in which the cars belonging to John Smith were involved.

Select count (distinct \*)

From accident

Where exists

(Select \*

From participated, person

Where participated.driver-id = person.driver-id

and person.name = ‘John Smith’

and accident.report-number = participated.report-number)

1. Delete the Mazda belonging to John Smith.

Delete car

Where model = ‘Mazda’

and license in (Select license

From person p, owns o

Where p.name = ‘John Smith’ and p.driver-id = o.driver-id)

1. Update the damage amount for the car with license number ‘AABB2000’ in the accident with report number ‘AR2197’ to $3000.

Update participated

Set damage-amount = 3000

Where report-number = ‘AR2197’

and driver-id in (Select driver-id

From owns

Where license = ‘AABB2000’)

---------------------------------------------------------------------------------------------------------------------

**Part 3: Relational Algebra Exercises**

**Question-1:**

Given the following schema:

Branch (branch\_name, branch\_city, assets)

Customer (customer\_name, customer\_street, customer\_city)

Account (account\_number, branch\_name, balance)

Loan (loan\_number, branch\_name, amount)

Depositor (customer\_name, account\_number)

Borrower (customer\_name, loan\_number)

**Write the corresponding Relational Algebra Expression for each of the following questions:**

1. Find all loans of over $1200.

σamount>1200 (loan).

1. Find the loan number for each loan of an amount greater than $1200.

∏loan\_number (σamount>1200 (loan))

1. Find the names of all customers who have a loan, an account, or both, from the bank.

∏customer\_name(borrower) ∪ ∏customer\_name(depositor)

1. Find the names of all customers who have a loan and an account at the bank.

∏customer\_name(borrower) ∩ ∏customer\_name(depositor)

1. Find the names of all customers who have a loan at the Perryridge branch.

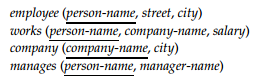
∏customer\_name(σbranch\_name=“Perryridge”(σborrower.loan\_number =loan.loan\_number(borrower x loan)))

1. Find the names of all customers who have a loan at the Perryridge branch but do not have an account at any branch of the bank.

∏customer\_name(σbranch\_name=“Perryridge”(σborrower.loan\_number=loan.loan\_number(borrower x loan))) – ∏customer\_name(depositor)

---------------------------------------------------------------------------------------------------------------------

**Question-2:**



Consider the relational database in the above figure, where the primary keys are underlined. Give an expression in the relational algebra to express each of the following queries:

1. Find the names of all employees who work for First Bank Corporation.

Πperson-name(σcompany-name = “First Bank Corporation”(works))

1. Find the names and cities of residence of all employees who work for First Bank Corporation.

Πperson-name, city(employee (σcompany-name=“First Bank Corporation”(works)))

1. Find the names, street address, and cities of residence of all employees who work for First Bank Corporation and earn more than $10,000 per annum.

Πperson-name, street, city(σ(company-name = “First Bank Corporation”∧salary > 10000) works employee)

1. Find the names of all employees in this database who live in the same city as the company for which they work.

Πperson-name (employee works company)

1. Find the names of all employees who live in the same city and on the same street as do their managers.

Πperson-name((employeemanages) (manager-name = employee2.person-name ∧ employee.street = employee2.street ∧ employee.city = employee2.city)(ρemployee2 (employee)))

1. Find the names of all employees in this database who do not work for First Bank Corporation.

Πperson-name(employee) − Πperson-name (σ(company-name = “First Bank Corporation”) (works))

1. Find the names of all employees who earn more than every employee of Small Bank Corporation.

Πperson-name(works)−(Πworks.person-name(works (works.salary ≤works2.salary ∧ works2.company-name = “Small Bank Corporation”) ρworks2(works)))

1. Assume the companies may be located in several cities. Find all companies located in every city in which Small Bank Corporation is located.

Πcompany-name (company ÷ (Πcity (σcompany-name = “Small Bank Corporation” (company))))

---------------------------------------------------------------------------------------------------------------------

**Part 4: Entity Relationship Modeling Exercises**

**Question-1:**

Assume we have the following application that models soccer teams, the games they play, and the players in each team. In the design, we want to capture the following:

* We have a set of teams, each team has an ID (unique identifier), name, main stadium, and to which city this team belongs.
* Each team has many players, and each player belongs to one team. Each player has a number (unique identifier), name, DoB, start year, and shirt number that he uses.
* Teams play matches, in each match there is a host team and a guest team. The match takes place in the stadium of the host team.
* For each match we need to keep track of the following:

- The date on which the game is played

- The final result of the match

- The players participated in the match. For each player, how many goals he scored, whether or not he took yellow card, and whether or not he took red card.

- During the match, one player may substitute another player. We want to capture this substitution and the time at which it took place.

* Each match has exactly three referees. For each referee we have an ID (unique identifier), name, DoB, years of experience. One referee is the main referee and the other two are assistant referee.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Team, Player, Match, Referee

--------------------------------------------------------------------------------------------------------------------**Question-2:**

UPS prides itself on having up-to-date information on the processing and current location of each shipped item. To do this, UPS relies on a company-wide information system. Shipped items are the heart of the UPS product tracking information system. Shipped items can be characterized by item number (unique), weight, dimensions, insurance amount, destination, and final delivery date. Shipped items are received into the UPS system at a single retail center. Retail centers are characterized by their type, uniqueID, and address. Shipped items make their way to their destination via one or more standard UPS transportation events (i.e., flights, truck deliveries). These transportation events are characterized by a unique scheduleNumber, a type (e.g, flight, truck), and a deliveryRoute.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Item, Retail Center, Transportation Event

---------------------------------------------------------------------------------------------------------------------**Question-3:**

As a part of its project management database, the company wants to store information about employees, projects and bookings. For each employee, the following information is stored: Employee ID, First and Last name, Rank, and billing rate. Employees are organized into solution sets, each solution set has a head of the solution set (SS), who is the resource owner for all employees in that SS. For each solution set we record the SS ID and the SS name. For scheduling purposes, we want to store information about the head of each solution set, and about assignment of employees to solution sets. An employee can belong to only one solution set. The scheduling system also stores information about project. For each project, the following information is stored: Project ID, Status, Location and Client name. As a part of the scheduling system, we store information about each calendar day in a year. When a booking is requested for an employee, the employee is scheduled to work on a particular project, on a particular day for the specified amount of time (10%-100%). For each booking we also record current status.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Employee, Booking, Solution Set, Project, Calendar Day

---------------------------------------------------------------------------------------------------------------------**Question-4:**

The organizers of the international multi-conference need to keep track of a large collection of workshops associated with the event. Initial requirements analysis brings out the following information about what needs to be recorded:

* Each workshop has a name, and happens on a particular date or dates, as some workshops last more than one day.
* There are several participants, each of which may sign up to one or more workshops.
* For each participant, it is important to record their name, email address, and the workshops which they wish to attend.
* There are a number of meeting rooms at the conference venue, each of a fixed capacity. Meetings rooms are identified by a floor and room number.
* Every workshop needs an allocated meeting room; where a workshop lasts for two days, it will use the same room on both days.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Participant, Workshop, Room, Day

---------------------------------------------------------------------------------------------------------------------

**Question-5:**

On-line payment system stores information about all customers, including name, id, address, e-mail and password. Each customer has set up a specific method of payment, which may be a credit card payment or automated direct withdrawal. For all types of payment we store the following information: an ID and the date the method of payment was set up. For credit card payments we store CC number and type and the expiration date. For automated withdrawal we store the name of financial institution, the routing number, account number and the date of monthly withdrawal.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Customer, Method of Payment (Credit Card, Direct Withdrawal)

---------------------------------------------------------------------------------------------------------------------

**Question-6:**

A salesperson may manage many other salespeople. A salesperson is managed by only one salesperson. A salesperson can be an agent for many customers. A customer is managed by one salesperson. A customer can place many orders. An order can be placed by one customer. An order lists many inventory items. An inventory item may be listed on many orders. An inventory item is assembled from many parts. A part may be assembled into many inventory items. Many employees assemble an inventory item from many parts. A supplier supplies many parts. A part may be supplied by many suppliers.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Salesperson, Customer, Order, Product, Part, Employee, Supplier

---------------------------------------------------------------------------------------------------------------------

**Question-7:**

The setting is that you are one organizer of a poster exhibition on Global Problems of the 21st Century, and you must design a database to keep track of the administration of the exhibition. Three main phases are recognized for the exhibition: the submission, the selection and the presentation phase. These are further explained below.

**Submission Phase:**

Graphic designers create posters for the exhibition to illustrate one of the chosen global problems. Relevant information on designers includes their name and their affiliation, i.e. the organization they work for. A poster has a title and is assigned an identification number, and it may be created by several graphic designers; although each individual designer may only be involved with one poster. Where a group of graphic designers create a poster, we distinguish between the main designer and the co-designers. In case of a single graphic designer, that person is considered to be the main designer of the poster. The main designer is always the point of contact, so should provide an email address.

**Selection Phase:**

All posters created for this exhibition are judged by members of a jury. A judge is a graphic design expert with experience in communication for raising public awareness and for public benefit. Judge information that is of relevance to the organizing committee includes the judge's name, their affiliation and email. Each poster is judged by three different judges. When judging a poster, a judge gives a decision: accept or reject. A poster is selected for the exhibition only if all three judges give an accept decision. Note that judges are not allowed to compete in Global Problems of the 21st Century themselves.

**Presentation Phase:**

All selected posters are then presented in the exhibition by their main graphic designers. The poster presentation is allocated a stand and an exhibition session. Each exhibition session takes place at a specific date, and 4 session topics have been announced: human rights, environmental pollution, poverty, and war.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Graphic Designer, Poster, Judge, Exhibition Session

---------------------------------------------------------------------------------------------------------------------

**Question-8:**

A database is to be set up to record details of experts on different subjects. The database will be used to contact experts to be called as witnesses in court cases. The following information needs to be recorded. For each expert, their witness identity number, name, affiliation, and email address. For each field of expertise, the name of the field. For each expert, all the fields of expertise that they are expert in. For each field of expertise, at most one identified leading expert.

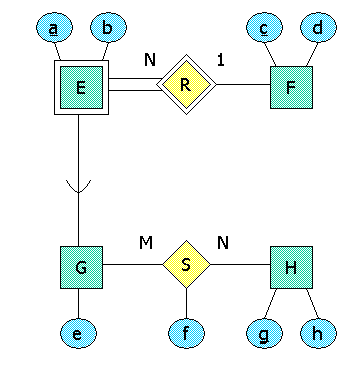
Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design.

**Main entities that should be included:** Expert, Field

---------------------------------------------------------------------------------------------------------------------

**Part 5: Converting ER Model to Relational Model Exercises**

**Question-1:**



1. Translate the above ER diagram to relations, via assigning a relation for the superclass and another relation for each subclass.

E(a,b,c): a weak entity set's relation includes the key of its owner, c in this case.

F(c,d): an ordinary entity-set-to-relation transformation.

G(a,c,e): in the required approach, a subclass's relation gets all its attributes and the key of its superclass.

H(g,h): an ordinary entity-set-to-relation transformation.

S(a,c,f,g): a many-many relationship's relation has the keys from all connected entity sets ({a,c} and g in this case), plus whatever attributes are attached to the relationship itself (f here).

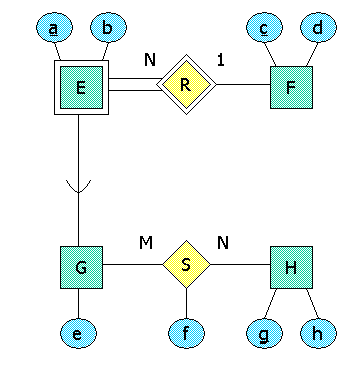
1. Using the above diagram, identify the relation in the list below that would be one of those that you construct.
2. E(a,b,c).
3. G(a,c,e).
4. S(a,c,f,g).
5. R(a,c).
6. E(a,b).
7. E(a,b,e).
8. G(a,b,c,e).
9. G(a,e).
10. S(a,f,g)
11. S(a,c,g)
12. S(a,c,e,f,g,h)
13. F(a,c,d)

Correct choices are: E(a,b,c), G(a,c,e), S(a,c,f,g).

1. Identify the incorrect choices in point b and state the reason.

Incorrect Choice 1: R(a,c)

|  |
| --- |
| Choice Explanation: a many-one relationship is not turned into a relation. Incorrect Choice 2: E(a,b) Choice Explanation: since E is weak, it needs help in forming a key. Follow the identifying relationship to see what other entity set must contribute to the key of E. Incorrect Choice 3: E(a,b,e) Choice Explanation: since e is only an attribute of subclass G, we do not use e as an attribute of all E entities. Incorrect Choice 4: G(a,b,c,e) Choice Explanation: in the required approach, the relation for a subclass does not include non key attributes that belong to the superclass. Incorrect Choice 5: G(a,e) Choice Explanation: since E is weak, it needs help in forming a key. Follow the identifying relationship to see what other entity set must contribute to the key of E. Incorrect Choice 6: S(a,f,g) Choice Explanation: since E is weak, it needs help in forming a key. Follow the identifying relationship to *see* what other entity set must contribute to the key of E. Incorrect Choice 7: S(a,c,g) Choice Explanation: do not forget that a many-many relationship gets a relation that includes its own attributes. Incorrect Choice 8: S(a,c,e,f,g,h) Choice Explanation: the relation for a many-many relationship has only the key attributes of its participating entity sets. Incorrect Choice 9: F(a,c,d) Choice Explanation: possible error: you are confusing the direction of a weak entity set's identifying relation. ---------------------------------------------------------------------------------------------------------------------  **Question-2:** |



1. Translate the above ER diagram to relations, using the "discriminating" attribute *t*.

Here are the relations constructed from the ER elements:

* E(a,b,c,e,t): a weak entity set's relation includes the key of its owner, c in this case. In addition, the required method puts the attributes of each subclass in this relation, and also uses the discriminating attribute t, that here can only have values "in G" or "not in G."
* F(c,d): an ordinary enitity-set-to-relation transformation.
* H(g,h): an ordinary enitity-set-to-relation transformation.
* S(a,c,f,g): a many-many relationship's relation has the keys from all connected entity sets ({a,c} and g in this case), plus whatever attributes are attached to the relationship itself (f here).

1. Using the above diagram, identify the relation in the list below that would be one of those that you construct.
2. E(a,b,c,e,t)
3. R(a,c)
4. F(c,d)
5. F(a,c,d)
6. S(a,c,f,g)
7. E(a,b,e,t)
8. E(a,b,c,e)
9. G(a,b,c,e)
10. G(a,b,c,e,t)
11. S(a,f,g)
12. S(a,c,g)
13. S(a,c,e,f,g,h)

The correct choices are E(a,b,c,e,t), F(c,d) , and S(a,c,f,g).

1. Identify the incorrect choices in point b and state the reason.

Incorrect Choice 1: F(a,c,d)

Choice Explanation: possible error: you are confusing the direction of a weak entity set's identifying relation.

Incorrect Choice 2: R(a,c)

Choice Explanation: a many-one relationship is not turned into a relation

Incorrect Choice 3: E(a,b,e,t)

Choice Explanation: since E is weak, it needs help in forming a key. Follow the identifying relationship to see what other entity set must contribute to the key of E.

Incorrect Choice 4: E(a,b,c,e)

Choice Explanation: if these are the attributes of E, then there is no way to tell whether an entity is in the subclass G.

Incorrect Choice 5: G(a,b,c,e)

Choice Explanation: in the required approach, the G-entities are taken care of by the relation for the superclass of G.

Incorrect Choice 6: G(a,b,c,e,t)

Choice Explanation: in the required approach, the G-entities are taken care of by the relation for the superclass of G.

Incorrect Choice 7: S(a,f,g)

Choice Explanation: since E is weak, it needs help in forming a key. Follow the identifying relationship to see what other entity set must contribute to the key of E.

Incorrect Choice 8: S(a,c,g)

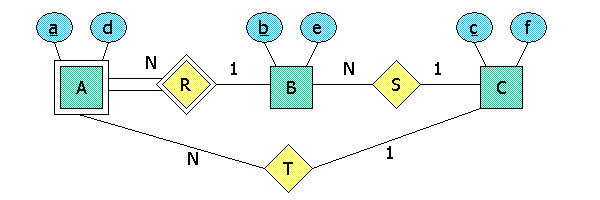
Choice Explanation: do not forget that a many-many relationship gets a relation that includes its own attributes.

Incorrect Choice 9: S(a,c,e,f,g,h)

Choice Explanation: the relation for a many-many relationship has only the key attributes of its participating entity sets.

-------------------------------------------------------------------------------------------------------------------**Question-3:**

Convert the ER diagram below to relations.

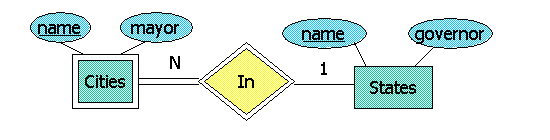


A(a, b ,c, d); this relation is the translation of weak entity set A, plus the many-one relationship T. B(b, c, e); this relation comes from entity set B plus the many-one relationship S. C(c, f); this relation comes from entity set C.

---------------------------------------------------------------------------------------------------------------------

**Question-4:**

1. Convert this E/R diagram to relations, resolving the dual use of the attribute "name" in some reasonable way.



One correct translation to relations is: Cities(cityName, stateName, mayor) and States(name, governor).

There are others, of course, since we cannot dictate what the attributes of the relation are called. However, note that we cannot use the term "name" for both the city name and the state name in Cities.

Also, observe that because Cities is a weak entity set, its relation must include both its key and that of the "owning" entity set States. That is why we are forced to have two "names" in the Cities relation. Finally, observe that in being an identifying relationship for a weak entity set, does not get a relationship of its own. Its information is kept within the Cities relation.

1. Referring to the above diagram, indicate which of the database schemas below is the most reasonable translation from the ER diagram above into relations?
2. Cities(cityName, stateName, mayor), States(name, governor)
3. Cities(name, stateName, mayor), In(name, stateName), States(name, governor)
4. Cities(cityName, mayor), States(name, governor)
5. Cities(cname, mayor), States(sname, gov)
6. Cities(cname, sname, mayor), In(cname, sname), States(sname, gov)
7. Cities(name, name, mayor), States(name, governor)
8. Cities(name, mayor), States(name, governor)
9. Cities(cityName, stateName, mayor), In(cityName, stateName), States(name, governor)
10. Cities(name, stateName, mayor), States(name, governor)
11. Cities(cname, sname, mayor), States(sname, gov) Cities(cname, sname, mayor), States(sname, gov) Cities(name, stateName, mayor), States(name, governor) Cities(cityName, stateName, mayor), States(name, governor)
12. Referring to the above diagram, indicate the incorrect translations from the ER diagram above into relations and state why?

Incorrect Choice 1: Cities(cityName, stateName, mayor), In(cityName, stateName), States(name, governor)

Choice Explanation: remember, an identifying relationship such as In does not need a relation. Its information is always contained in the relation of the weak entity set it helps identify, Cities in this case.

Incorrect Choice 2: Cities(name, mayor), States(name, governor)

Choice Explanation: because Cities is a weak entity set, its relation must include not only its own attributes, but the key of the "owning" entity set(s), States in this case. Since both Cities and States have an attribute "name," we need to rename at least one of them so we can tell, in the relation, whether the city name or the state name is being referred to.

Incorrect Choice 3: Cities(name, name, mayor), States(name, governor)

Choice Explanation: while the attributes "name" from both Cities and States are required in the relation (because Cities is a weak entity set), we cannot use "name" as an attribute twice in the relation for Cities. We must rename at least one of them, e.g., call them cityName and stateName.

Incorrect Choice 4: Cities(cname, sname, mayor), In(cname, sname), States(sname, gov)

Choice Explanation: remember, an identifying relationship such as In does not need a relation. Its information is always contained in the relation of the weak entity set it helps identify, Cities in this case.

Incorrect Choice 5: Cities(cname, mayor), States(sname, gov)

Choice Explanation: because Cities is a weak entity set, its relation must include not only its own attributes, but the key of the "owning" entity set(s), States in this case. Since both Cities and States have an attribute "name," we need to rename at least one of them so we can tell, in the relation, whether the city name or the state name is being referred to.

Incorrect Choice 6: Cities(cityName, mayor), States(name, governor)

Choice Explanation: because Cities is a weak entity set, its relation must include not only its own attributes, but the key of the "owning" entity set(s), States in this case. Since both Cities and States have an attribute "name," we need to rename at least one of them so we can tell, in the relation, whether the city name or the state name is being referred to.

Incorrect Choice 7: Cities(name, stateName, mayor), In(name, stateName), States(name, governor)

Choice Explanation: remember, an identifying relationship such as It does not need a relation. Its information is always contained in the relation of the weak entity set it helps identify, Cities in this case.

---------------------------------------------------------------------------------------------------------------------

**Question-5:**

Here is a table representing a relation named R:

|  |  |  |
| --- | --- | --- |
| A | B | C |
| 0 | 1 | 2 |
| 3 | 4 | 5 |
| 6 | 7 | 8 |

* 1. Identify the attributes of R, the schema of R, the tuples of R, the components of the tuples for each attribute of R.

The schema of R is R(A,B,C). The three column headers, A, B, and C, are the attributes of R. Each of the rows except the top (column headers) is a tuple; that is, the tuples are (0,1,2), (3,4,5), and (6,7,8). Each of the integers 0 through 8 is the value of some component of some tuple --- a component for the attribute heading its column. Thus, for instance, 0 is an A-component and 4 is a B-component.

* 1. State which of the following is NOT a true statement about relation R?

|  |  |
| --- | --- |
| 1. R has four tuples. 2. 0 is a tuple of R. 3. (A,B,C) is a tuple of R 4. The schema of R is R(A,B,C). 5. A is an attribute of R. 6. B is an attribute of R. 7. C is an attribute of R. | 1. (0,3,6) is a tuple of R 2. (0,1,2) is a tuple of R. 3. (3,4,5) is a tuple of R. 4. (6,7,8) is a tuple of R. 5. 0 is the value of the A-component of one of the tuples of R. 6. 4 is the value of the B-component of one of the tuples of R. |

Correct Choice 1: R has four tuples. Correct Choice 2: 0 is a tuple of R. Correct Choice 3: (A,B,C) is a tuple of R. Correct Choice 4: (0,3,6) is a tuple of R. Incorrect Choice 1: (0,1,2) is a tuple of R. Choice Explanation: the first tuple (second row) is indeed the tuple (0,1,2). Incorrect Choice 2: the schema of R is R(A,B,C). Choice Explanation: the top row (column headers) are the attributes A, B, and C. The name of the relation is R. A convenient representation for the schema of a relation is the name of the relation followed by a parenthesized list of the attributes, in order. Incorrect Choice 3: (3,4,5) is a tuple of R. Choice Explanation: the second tuple (third row) is indeed the tuple (3,4,5). Incorrect Choice 4: (6,7,8) is a tuple of R. Choice Explanation: the last (third) tuple is indeed (6,7,8). Incorrect Choice 5: A is an attribute of R. Choice Explanation: A is the column header of the first column, and therefore is one of the attributes of R. Incorrect Choice 6: B is an attribute of R. Choice Explanation: B is the column header of the second column, and therefore is one of the attributes of R. Incorrect Choice 7: C is an attribute of R. Choice Explanation: C is the column header of the third column, and therefore is one of the attributes of R. Incorrect Choice 8: R has three tuples. Choice Explanation: the top row is the column headers, or attributes of the relation. Thus, only the last three rows are tuples. Incorrect Choice 9: 0 is the value of the A-component of one of the tuples of R. Choice Explanation: indeed, 0 is the value in the first (or A-) component of the first tuple (0,1,2). Incorrect Choice 10: 4 is the value of the B-component of one of the tuples of R. Choice Explanation: indeed, 4 is the value in the second (or B-) component of the second tuple (3,4,5). ---------------------------------------------------------------------------------------------------------------------