CHAPTER 29

Object-Based Databases

Practice Exercises

29.1 A car-rental company maintains a database for all vehicles in its current fleet. For all vehicles, it includes the vehicle identification number, license number, manufacturer, model, date of purchase, and color. Special data are included for certain types of vehicles:

- Trucks: cargo capacity
- Sports cars: horsepower, renter age requirement
- Vans: number of passengers
- Off-road vehicles: ground clearance, drivetrain (four- or two-wheel drive)

Construct an SQL schema definition for this database. Use inheritance where appropriate.

Answer:

For this problem, we use table inheritance. We assume that **MyDate**, **Color** and **DriveTrainType** are predefined types.

```
create type Vehicle
    (vehicle_id integer,
     license_number char(15),
     manufacturer char(30),
     model char(30),
     purchase_date MyDate,
     color Color)

create table vehicle of type Vehicle

create table truck
    (cargo_capacity integer)
    under vehicle

create table sportsCar
    (horsepower integer
     renter_age_requirement integer)
    under vehicle

create table van
    (num_passengers integer)
    under vehicle

create table offRoadVehicle
    (ground_clearance real
     driveTrain DriveTrainType)
    under vehicle
```

29.2 Consider a database schema with a relation *Emp* whose attributes are as shown below, with types specified for multivalued attributes.

Emp = (ename, ChildrenSet **multiset***(Children), SkillSet* **multiset***(Skills))*
Children = (name, birthday)
Skills = (type, ExamSet **setof***(Exams))*
Exams = (year, city)

Answer the following:

a. Define the above schema in SQL, with appropriate types for each attribute.

b. Using the above schema, write the following queries in SQL.

i. Find the names of all employees who have a child born on or after January 1, 2000.

ii. Find those employees who took an examination for the skill type "typing" in the city "Dayton".

iii. List all skill types in the relation *Emp*.

Answer:

a. No Answer.

b. Queries in SQL.

i. Program:

```
select ename
from emp as e, e.ChildrenSet as c
where 'March' in
        (select birthday.month
         from c
        )
```

ii. Program:

```
select e.ename
from emp as e, e.SkillSet as s, s.ExamSet as x
where s.type = 'typing' and x.city = 'Dayton'
```

iii. Program:

```
select distinct s.type
from emp as e, e.SkillSet as s
```

29.3 Consider the E-R diagram in Figure 29.5, which contains composite, multivalued, and derived attributes.

a. Give an SQL schema definition corresponding to the E-R diagram.

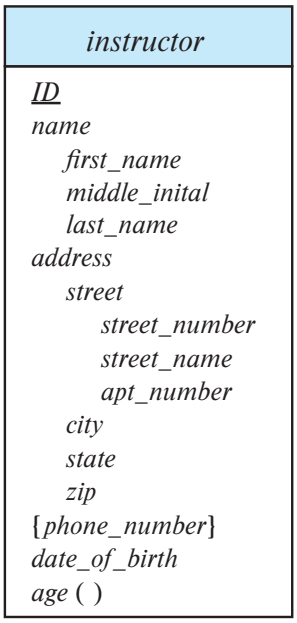

Figure 29.5 E-R diagram with composite, multivalued, and derived attributes.

b. Give constructors for each of the structured types defined above.

Answer:

a. The corresponding SQL:1999 schema definition is given below. Note that the derived attribute *age* has been translated into a method.

```
create type Name
    (first_name varchar(15),
     middle_initial char,
     last_name varchar(15))
create type Street
    (street_name varchar(15),
     street_number varchar(4),
     apartment_number varchar(7))
create type Address
    (street Street,
     city varchar(15),
     state varchar(15),
     zip_code char(6))
create table customer
    (name Name,
     customer_id varchar(10),
```

```
        address Adress,
        phones  char(7)  array[10],
        dob date)
    method integer age( )
```

b.
```
create function Name (f varchar(15), m char, l varchar(15))
returns Name
begin
    set first_name = f;
    set middle_initial = m;
    set last_name = l;
end
create function Street (sname varchar(15), sno varchar(4), ano varchar(7))
returns Street
begin
    set street_name = sname;
    set street_number = sno;
    set apartment_number =ano;
end
create function Address (s Street, c varchar(15), sta varchar(15), zip varchar(6))
returns Address
begin
    set street = s;
    set city = c;
    set state =sta;
    set zip_code =zip;
end
```

29.4 Consider the relational schema shown in Figure 29.6.

a. Give a schema definition in SQL corresponding to the relational schema, but using references to express foreign-key relationships.

b. Write each of the queries below on the schema in Figure 29.6, using SQL.

i. Find the company with the most employees.

ii. Find the company with the smallest payroll.

iii. Find those companies whose employees earn a higher salary, on average, than the average salary at First Bank Corporation.

Answer:

a. The schema definition is given below. Note that backward references can be added, but they are not so important as in OODBS because queries can be written in SQL, and joins can take care of integrity constraints.

```
create type Employee
    (person_name varchar(30),
     street varchar(15),
     city varchar(15))
create type Company
    (company_name varchar(15),
    (city varchar(15))
create table employee of Employee
create table company of Company
create type Works
    (person ref(Employee) scope employee,
    comp ref(Company) scope company,
    salary int)
create table works of Works
create type Manages
    (person ref(Employee) scope employee,
    (manager ref(Employee) scope employee)
create table manages of Manages
```

b. i.
```
select comp->name
from works
group by comp
having count(person) ≥ all(select count(person)
                           from works
                           group by comp)
```

ii.
```
select comp->name
from works
group by comp
```

---

employee (*person_name*, *street*, *city*)
works (*person_name*, *company_name*, *salary*)
company (*company_name*, *city*)
manages (*person_name*, *manager_name*)

Figure 29.6 Relational database for Exercise 29.4.

having sum(*salary*) $\leq$ **all**(**select sum**(*salary*)
from *works*
group by *comp*)

iii. **select** *comp*−>*name*
from *works*
group by *comp*
having avg(*salary*) > (**select avg**(*salary*)
from *works*
where *comp*−>*company_name*="First Bank Corporation")

29.5 Suppose that you have been hired as a consultant to choose a database system for your client's application. For each of the following applications, state what type of database system (relational, persistent programming language-based OODB, object-relational; do not specify a commercial product) you would recommend. Justify your recommendation.

a. A computer-aided design system for a manufacturer of airplanes.

b. A system to track contributions made to candidates for public office.

c. An information system to support the making of movies.

Answer:

a. A computer-aided design system for a manufacturer of airplanes:
An OODB system would be suitable for this. That is because CAD requires complex data types, and being computation oriented, CAD tools are typically used in a programming language environment needing to access the database.

b. A system to track contributions made to candidates for public office:
A relational system would be apt for this, as data types are expected to be simple, and a powerful querying mechanism is essential.

c. An information system to support the making of movies:
Here there will be extensive use of multimedia and other complex data types. But queries are probably simple, and thus an object-relational system is suitable.

29.6 How does the concept of an object in the object-oriented model differ from the concept of an entity in the entity-relationship model?

Answer:
An entity is simply a collection of variables or data items. An object is an encapsulation of data as well as the methods (code) to operate on the data. The data members of an object are directly visible only to its methods. The

outside world can gain access to the object's data only by passing predefined messages to it, and these messages are implemented by the methods.