6. Reduction of Generalization

In generalizations, higher level entity sets and lower level entity sets are considered. Make a table for higher level entity set with all its attributes. For lower level entity set, make a table with all its attributes with primary key attributes of its higher level entity set. Consider E-R diagram shown in Figure 2.20, in which Employee is high level entity set and Full Time Employee and Part Time Employee are two lower level entity sets. So, make three tables as given below:

- Employee [Emp-ID, Dept-ID, First-Name, Middle-Name, Last-Name, Salary]
- Full Time Employee [Emp-ID, Qualification]
- Part Time Employee [Emp-ID, Hours-Worked]

7. Reduction of Aggregation

Reduction of aggregation into tables is simple. Consider the E-R diagram shown in Figure 2.19. For all entity sets, make tables as discussed earlier. For making tables for relationship sets, consider the same approach as discussed earlier. Take an example of relationship set Manages. Make a table manages with all descriptive attributes, primary key of entity set Manager and the relationship set works-on.

SOLVED PROBLEMS

Problem 1. Construct an E-R diagram for a hospital with a set of patients and a set of medical doctors. Associate with each patient, a log of various tests and examinations conducted. Construct the appropriate tables for this E-R diagram and list the tables with their attributes, primary key and foreign keys.

Solution. The E-R diagram is shown in Figure 2.28.

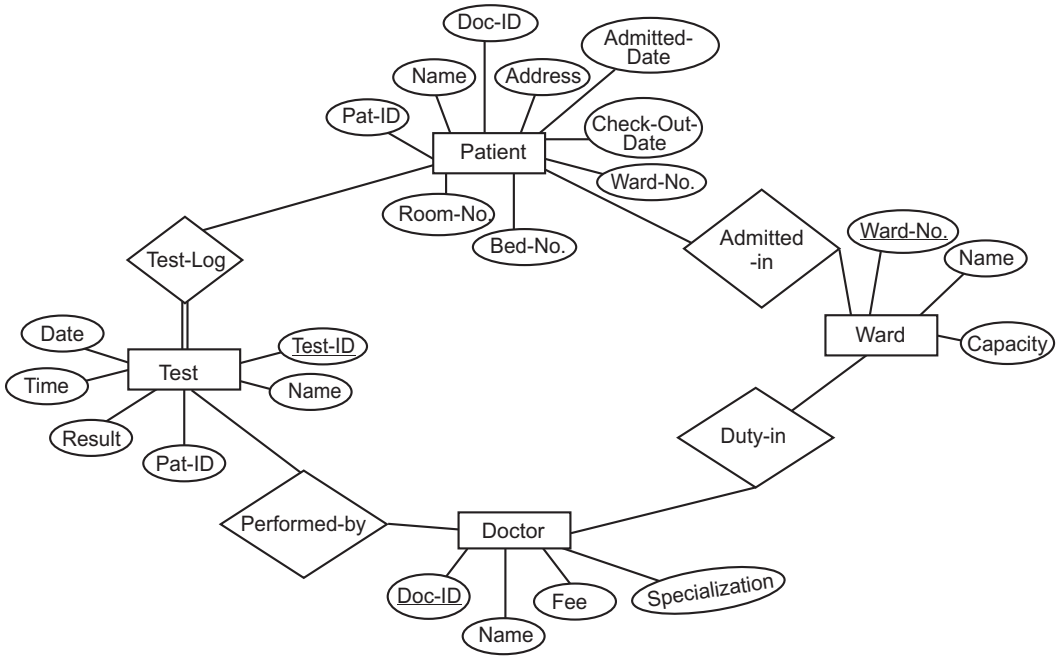

FIGURE 2.28. E-R diagram of hospital.

The Tables are as follows :

Patient (Pat-ID, name, address, admitted-date, check-out-date, room-no., bed-no., ward-no, doc-ID)

Ward (Ward-no., name, capacity)

Doctor (Doc-ID, name, fee, specialization)

Test (Test-ID, name, date, time, result, Pat-ID)

Primary key is shown by _____.

Foreign key is shown by _____.

Problem 2. The people's Bank offers five type of accounts : Loan, checking, premium savings, daily interest saving, and money market. It operates a number of branches and a client of bank can have any number of account. Accounts can be joint *i.e.,* more than one client may be able to operate a given accounts. Identify the entries of interest and show their attribute. What relationship exists among these entities? Draw the corresponding E-R diagram.

Solution. The E-R diagram is shown in Figure 2.29.

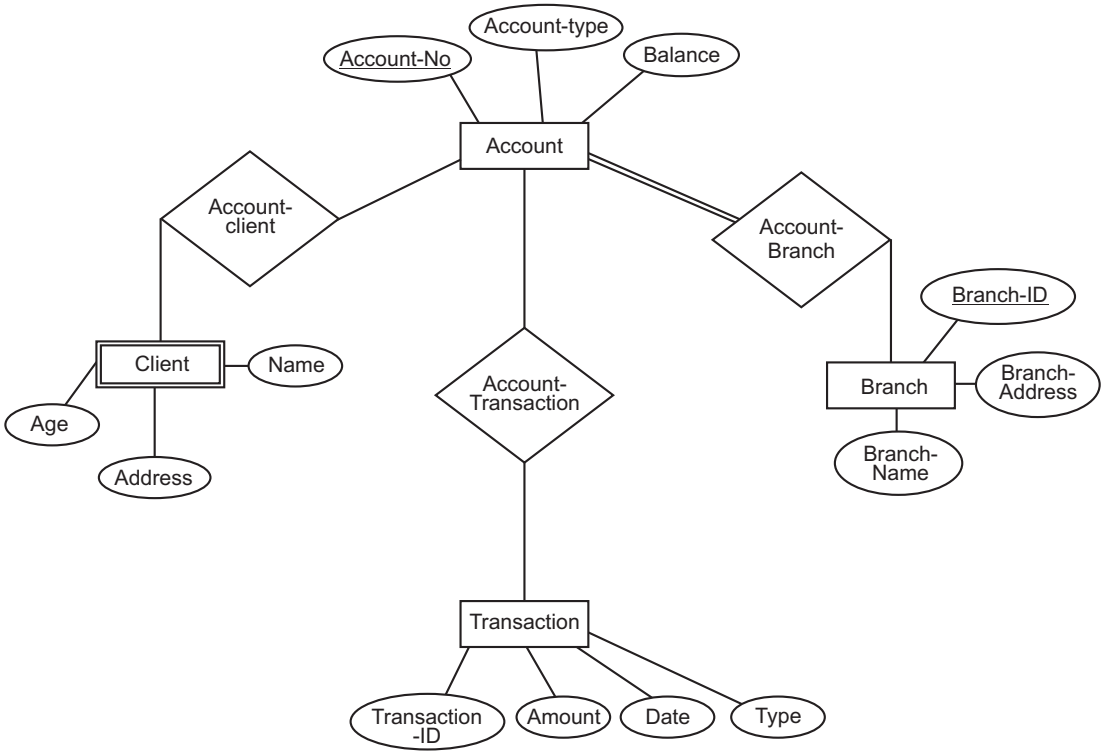

FIGURE 2.29. E-R diagram of bank.

The Entities are as follows :

Account (Account-no, Account-type, Balance)

Branch (Branch-ID, Branch-address, Branch-name)

Transaction (Transaction-ID, Amount, Date, Type)

Client (Account-No., Name, Age, Address)

Relationships are Account-Branch, Account-Transaction, Account-Client.

Problem 3. Draw an entity-Relationship diagram of a manufacturing company which records information about the projects it has on hand, the parts used in projects, the suppliers who supply the parts, the warehouses in which those parts are stored, the employees who work on these projects.

Solution. The E-R diagram is shown in Figure 2.30.

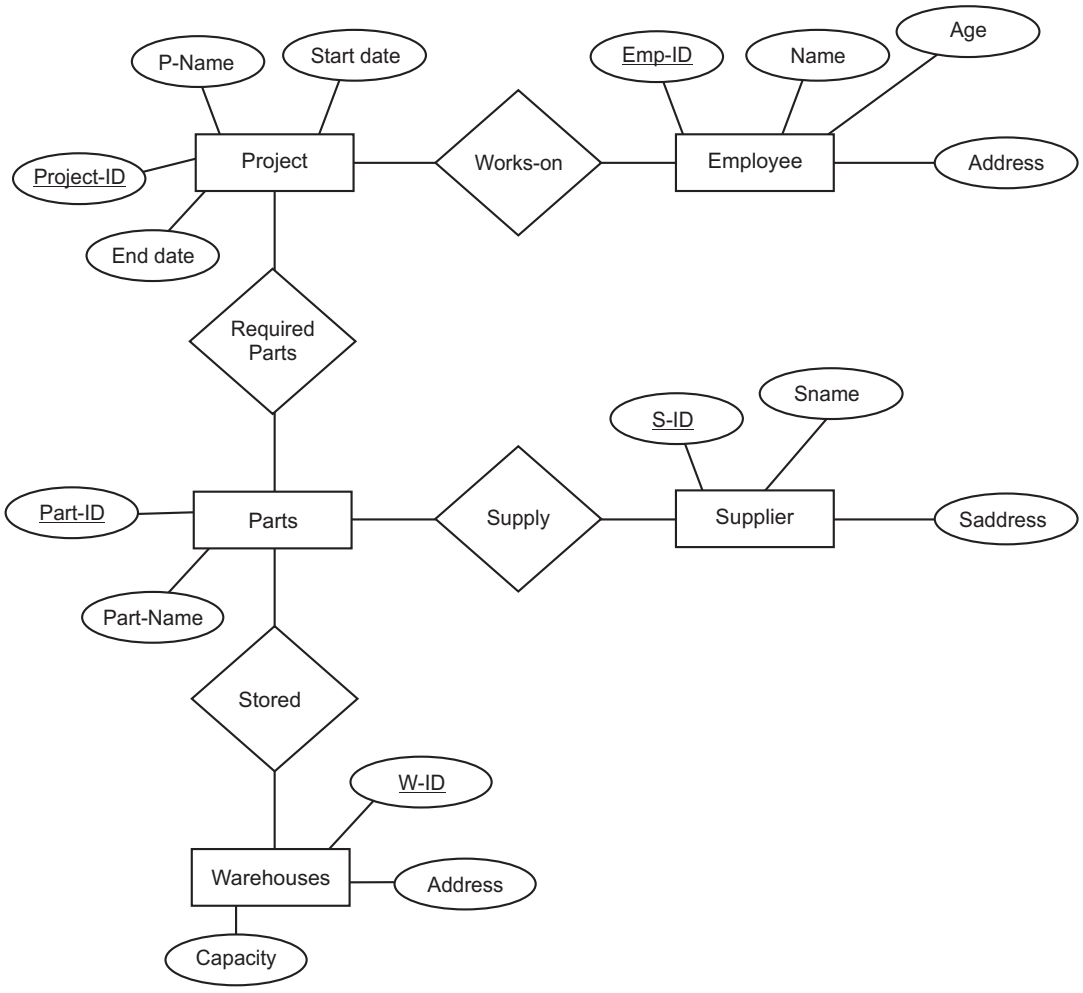

FIGURE 2.30. E-R diagram of manufacturing company.

Problem 4. A chemical or set of chemicals gives rise to another chemical or set of chemicals, when reacts under no condition or a set of conditions.

For example, Methane and Chlorine gives rise to Chloromethane when exposed to light. Here Methane and Chlorine are the reactants, which when under condition "Exposure to sunlight" giving Chloromethane as product.

Similarly, reaction of Water and Sodium gives Sodium Hydroxide and Hydrogen and no condition is required.

There are numerous reactions possible and each reactions has to be given a reaction number. Each chemical and condition has to be given a code.

Answer the following :

(*i*) Identify the entities in the above system.

(*ii*) Identify the attributes of the entities identified in (*i*)

(*iii*) Identify relations and their cardinalities

(*iv*) Draw E-R diagram for the above system.

Solution.

(*i*) Entities are Chemical, Condition and Reaction

(*ii*) Attributes of these entities are

Chemical (Chem-code, name, color, state)

Condition (Cond-Code, details)

Reaction (Reac-number, Reaction-Type)

(*iii*) Relations between these entities are

ON (between Chemical and Condition)

RESULTS (between Condition and Reaction)

PRODUCE (between Reaction and Chemical)

Many-to-Many cardinalities exist between all the above entities.

(*iv*) The E-R diagram is shown in Figure 2.31.

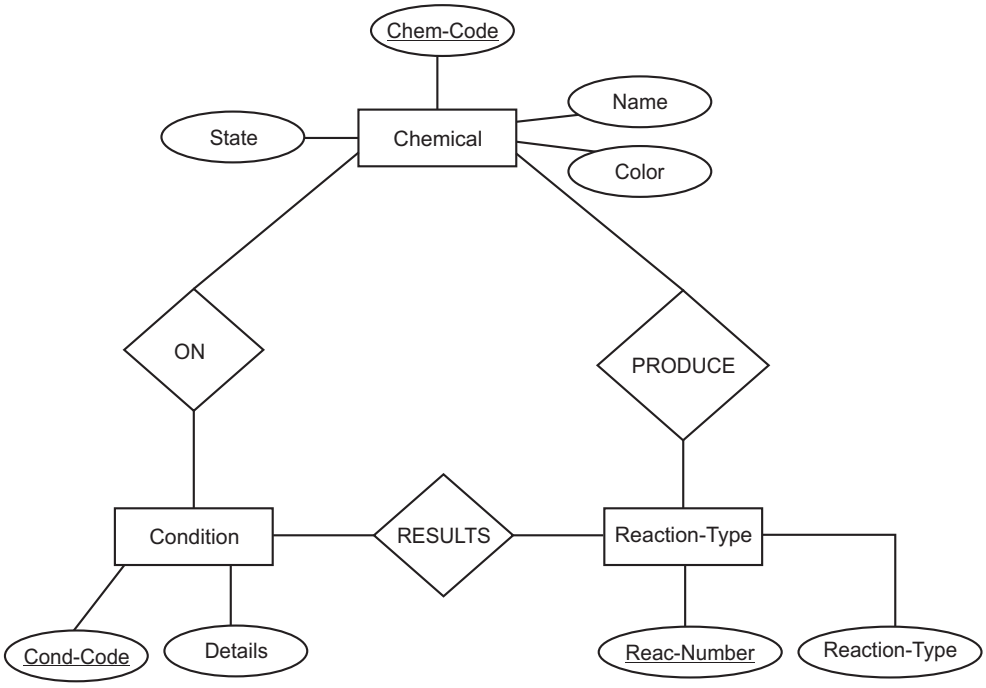

FIGURE 2.31. E-R diagram of chemical reaction.

Problem 5. In a manufacturing industry labourers are given different jobs on different days and each job has its own monthly basic and monthly DA rates as wages to be paid to labours. A labour is not given more than one type of job on a day. A database designer is given the job to design database for above situation and the designer designs one of the tables as :

Field	Type	Remarks
From date	Date	From this date to
To date	Date	This date
Labour Number	Number(6)	Labour Number
Job Done Code	Char(6)	Does the job Done Code
At Basic Rate	Number(10,2)	At the this Basic rate
At DA Rate	Number(10,2)	And this DA rate

Draw E-R diagram for above situation.

Solution. The E-R diagram is shown in Figure 2.32.

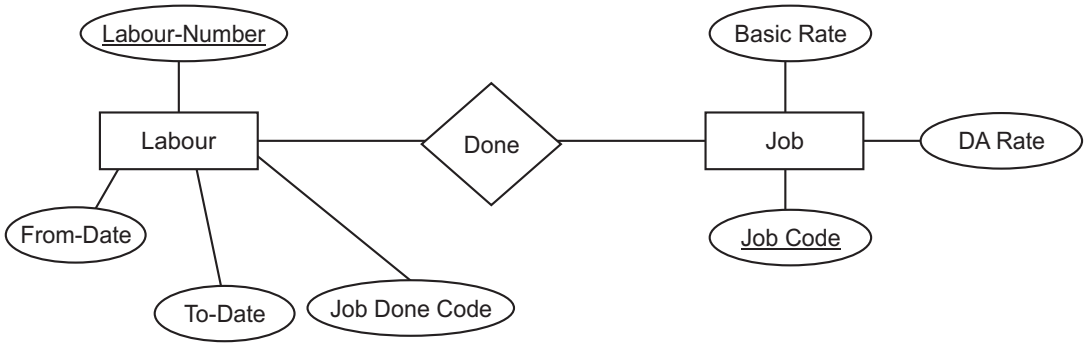

FIGURE 2.32. E-R diagram of manufacturing industry.

Problem 6. Translate the given E-R diagram to relational schema.

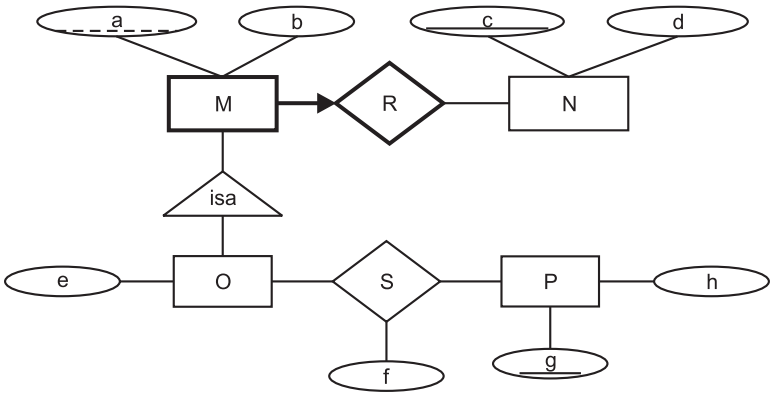

FIGURE 2.33

Solution.
- N is an entity, so we will create a table for it: $N(\underline{c}, d)$.
- P is an entity, so we will create a table for it: $P(h, \underline{g})$.

- Since M is a weak entity, we will create *one* table for it *and* R, which contains the key of N as a key: $M\_R(\underline{a}, b, \underline{c})$, where c is a foreign key of N. Because R is a weak entity, we must delete a M_R tuple if the corresponding N tuples disappears.
- Now we create a relation for O, which must include the key of M. The key of M includes the key of N since it is a weak entity, resulting in: $O(e, \underline{a, c})$, where a and c are a foreign key of M_R. Note that technically speaking c is really a foreign key of N, but since the requirements are that you must refer to the *entire* key of a table, we must have it refer to M_R's key, rather than N's.
- S is a many to many relationship, so we will create a table for it which includes the attributes of S and the keys of O and P, which together form the primary key of S: $S(f, \underline{a, c, g})$, where a and c are foreign key references to O, and g is a foreign key reference to P.

Problem 7. Consider the following E-R diagram:

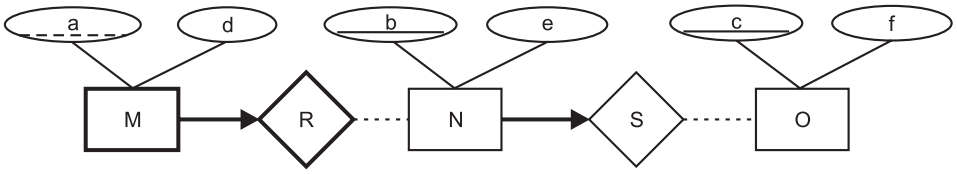

FIGURE 2.34

(*a*) This diagram presents two conflicting rules that apply to whether N should be represented by its own table in the relational schema. What are they?

(*b*) Which one would you use and why?

Solution. (*a*) The two rules are:

(*i*) M is a weak entity dependant on N

(*ii*) N is the many side of a many to one relationship (S – denoted by the arrow) so N and S should be represented by the same relation in the relational schema.

(*b*) Because M is a weak entity, we have no choice on how to model it; it must include the information about N's key. The choice is what do we do about NS. If we follow both rules, we have the relations:

(*i*) $NS(\underline{b}, e, c)$ – note that c is *not* needed as part of a key because we know which S relationship we are referring to based only on the many side of the relationship (N)

(*ii*) $MR(\underline{a, b}, d)$ – with a foreign key to NS

(*iii*) $O(\underline{c}, f)$

which would mean that the concept of MR depends now on NS, not just on N. On the one hand, one could argue that this isn't a problem. For one thing, it'd be worse if c were part of the key of NS, but it isn't. Besides, this makes for smaller numbers of tables, and less duplication. Since we have the fact that there is total participation for N in S (denoted by the thick line from N to S), there aren't going to be any null values. So combining them. On the other hand, we now have the fact that M depends on the relationship with S.

Problem 8. Convert the following ER – diagram into a relational database (the primary keys are underlined):

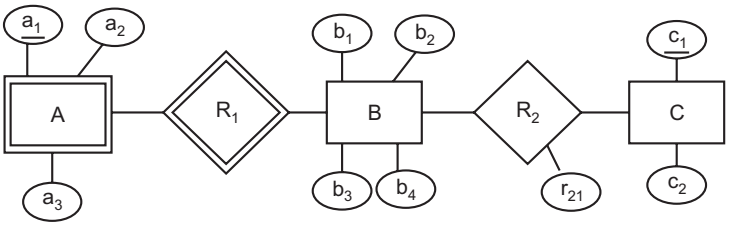

FIGURE 2.35

Solution. The relational database schema for the given *ER* diagram is as follows:

$A(\underline{a_1},\ \underline{b_1},\ a_2,\ a_3)$

$B(\underline{b_1},\ b_2,\ b_3,\ b_4)$

$C(\underline{b_1},\ c_2)$

$D(\underline{b_1},\ \underline{c_1},\ r_{21})$

Problem 9. Map the following ER diagram to a relational database. Give the relation names and attributes in them. Also mention the primary key and foreign keys if any for each table.

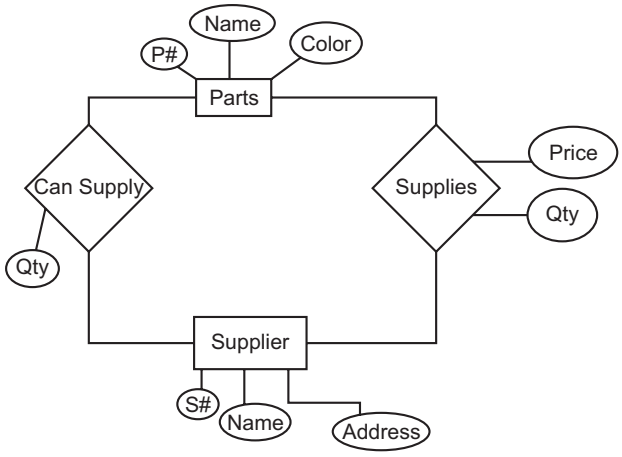

FIGURE 2.36

Solution. The following relations with attribute names are obtained from the given *ER* Diagram. The primary keys are underlined and Foreign keys are defined with each relation.

Parts(P#, Name, Color). There is no Foreign Key.

Supplier(S#, Name, Address). There is no Foreign Key.

Can_Supply(P#, S#, QTY). P# references Parts.P# and S# references Supplier.S#.

Supplies(P#, S#, Qty, Price). P# references Parts.P# and S# references Supplier.S#.

Problem 10. Consider the following ER diagram:

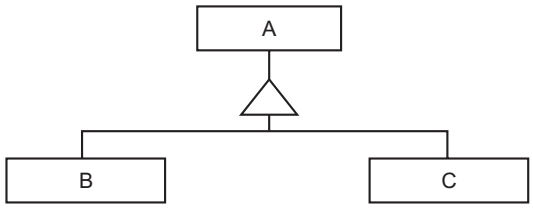

FIGURE 2.37

where *A*, *B* and *C* are entity sets.

1. Specify the condition(s) that is(are) necessary in order to represent all three sets with a single table.
2. Specify the condition(s) that is(are) necessary in order to represent all three sets with two tables, one for *B* and one for *C*.

Solution. 1. The ISA relationship must be disjoint.

B and *C* must have the same attributes.

2. The ISA relationship must be total.

Problem 11. Suppose we define a database about the customers of a bank and the loans they have received from the bank. For each customer we need to record information about their name, address, phone number and the company they work for. For each loan we need to record the amount, the interest rate, date the loan was issued, and the date the loan should be paid off.

(*i*) Is it a good idea to represent the company for which a customer works as an attribute of the customer or as a relationship? Briefly justify your answer.

(*ii*) Which is the best way to represent the relationship between the customer and their loans:

(*a*) by defining the loan as an attribute of the customer, or

(*b*) by making the loan a separate entity set and defining a relationship set between it and the customer?

Briefly justify your answer.

Solution. (*i*) The company should be an attribute of the customer, assuming each customer works for a single company. We don't need to keep any information for each company.

(*ii*) The loan should be a separate entity set associated with a customer through a relationship.

Reasons:

- A customer may have more than one loans.
- A loan has additional information on its own.

Problem 12. (*a*) Construct an E-R diagram for the following description.

Design a database for the reservation office of a bus company.

– Each bus has a unique number. We also store its class and capacity.

– Each place has a unique name and location information of the latitude and longitude.
– Routes have a starting place and an ending place; also, some of them have several intermediate places.
– A number of buses are scheduled to a route. A bus is assigned to one schedule; some buses can have multiple schedule. We store the date and starting time of each schedule.
– A member of our company can book a bus by specifying a schedule. We store a unique id, first name, and last name of each member.
– For each reservation, credit card number, the number of passengers, and the reservation datetime are stored.

(*b*) Convert your E-R diagram into a relational schema.

Solution. (*a*) The ER diagram is given below.

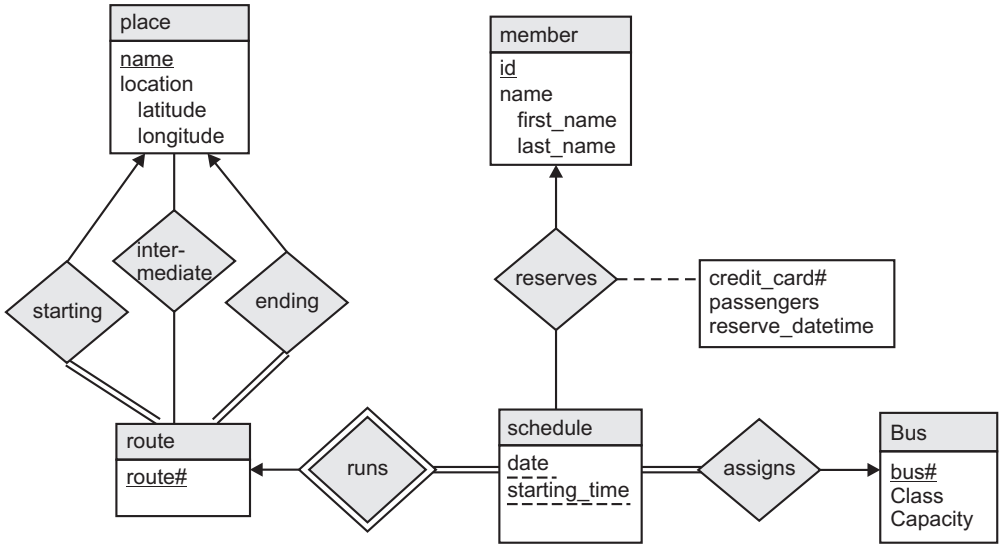

FIGURE 2.38

(*b*) The corresponding relational schema is as follows:

place(name, latitude, longitude)

route(route#)

starting_place(route#, name)

ending_place(route#, name)

intermediate_place(route#, name)

schedule(route#, day, starting_time)

bus(bus#, class,capacity)

assignment(route#, day, starting_time, bus#)

member(id, first_name, last_name)

reservation(id, route#, day, starting_time, credit_card#, passengers, reserve_datetime)

Problem 13. Consider the following Entity/Relationship diagram:

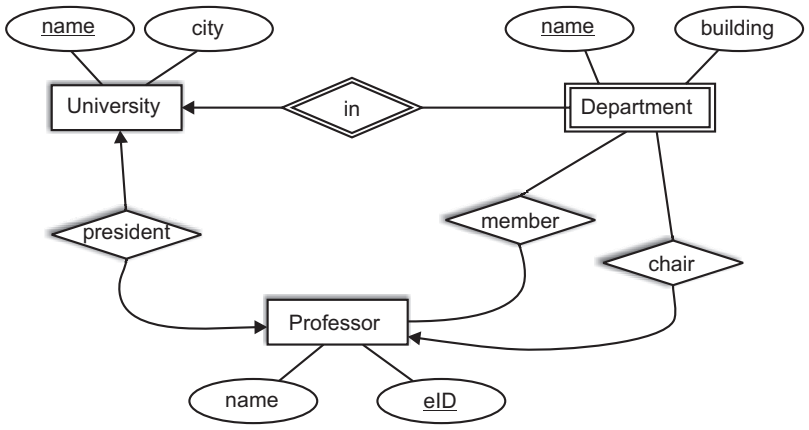

FIGURE 2.39

Which of the following statements are true according to this Entity/Relationship diagram?

1. Each department must be in exactly one university.
2. A university may have no departments.
3. No two departments can have the same name.
4. No two universities can have the same name.
5. A professor can be president of more than 1 university.
6. A university can have no president.
7. A department can have no chair.
8. A professor can be chair of more than one department.
9. There cannot be two universities in the same city.
10. Two departments with the same name must not be in two different universities.

Solution. 1. True 2. True 3. False 4. True 5. False 6. False 7. True 8. True 9. False 10. False

Problem 14. Consider the following ER diagram:

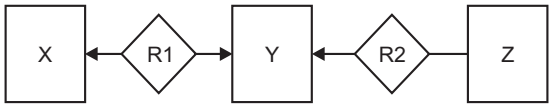

FIGURE 2.40

Which of the following cardinalities is valid for the entity sets? Do not guess. There is one point for each correct answer, –1 for each incorrect answer, and 0 points if you leave the answer blank.

1. $|X| = 0, |Y| = 0, |Z| = 0.$
2. $|X| = 0, |Y| = 0, |Z| = 8.$
3. $|X| = 5, |Y| = 5, |Z| = 0.$
4. $|X| = 3, |Y| = 3, |Z| = 6.$
5. $|X| = 2, |Y| = 0, |Z| = 0.$
6. $|X| = 0, |Y| = 5, |Z| = 5.$

Solution. 1. Valid 2. Invalid 3. Valid 4. Valid 5. Valid 6. Invalid

Problem 15. You have been tasked with designing a database for the Indian Census department to store all of their data, past and future. The database must conform to these constraints:

1. There has been a census every 10 years. The year of each census is unique to that census. There are also some notes as well as the total population of the India.
2. Each state has a unique name, and a value for its square area. Every state participates individually in every census, providing its population.
3. Every person has a unique SSN, as well as a name and birthday. Each person participates in every census by providing their age.
4. An address has a unique identifier, as well as a street name, city, state, and zipcode.
5. A person lives at only one address.

(*a*) Draw an ER diagram for this database. Be sure to mark the multiplicity of each relationship (1-1, 1-many, many-many, etc) of the diagram. Decide the key attributes and identify them on the diagram by underlining them. State all assumptions you make.

(*b*) Translate your ER diagram into a relational schema. Select approaches that yield the fewest number of relations; merge relations where appropriate. Specify the key of each relation in your schema. If the names of your foreign keys do not match the primary key name, please state the link in your assumptions.

Solution. (*a*)

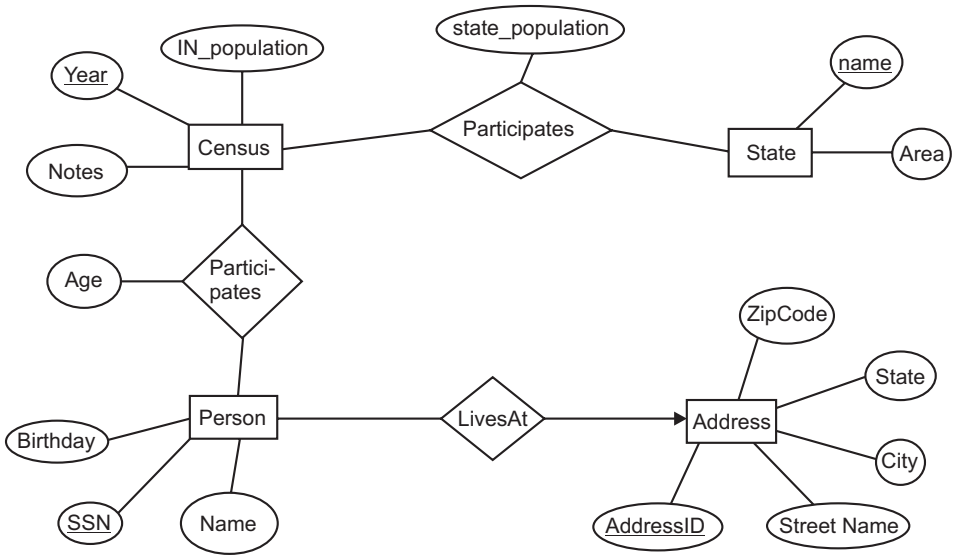

FIGURE 2.41

(*b*) Census(CensusYear, Notes, INPopulation);

State(StateName, Area);

StateParticipate(StateName, CensusYear, statepopulation);

Person(SSN, Name, Birthday, AddressID);

PersonParticipate(SSN, CensusYear, PersonAge);

Address(AddressID, Street, City, State, ZipCode).

Problem 16. Translate your Entity-Relationship Model (ER Diagram) into a logical model (DB Schema). For each relation in your schema, provide its name, attributes and keys (underlined attributes).Translate your Entity-Relationship Model (ER Diagram) from the question above into a logical model (DB Schema). For each relation in your schema, provide its name, attributes and keys (underlined attributes).

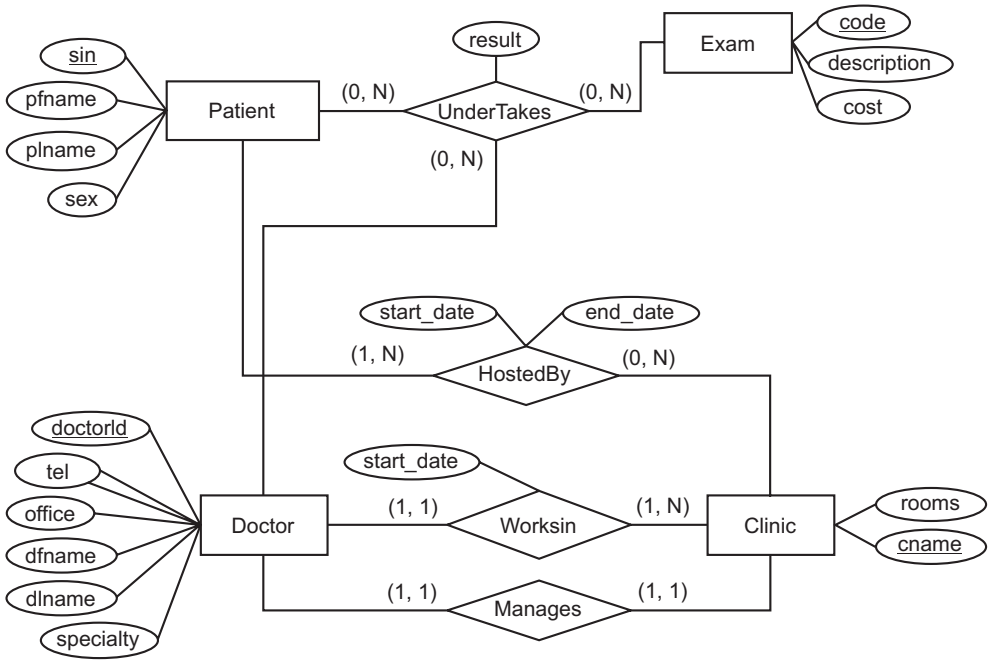

FIGURE 2.42

Solution.

Doctor

doctorId	dfname	dlname	office	specialty	cname	start_date

Tel

doctorId	tel

Patient

sin	pfname	plname	sex

Exam

code	description	cost

Clinic

cname	rooms	doctorId

Undertakes

sin	code	doctorId	result

Hosted By

sin	cname	start_date	end_date

Problem 17. Consider an application that needs to manage data for a travel agency. It needs to store the following entities and relationships:

- Hotels: have attributes name, address, price
- Resorts: are Hotels, that also have an attribute minimum-stay
- Activities: have attributes name, season
- Has: is a relationship between Resorts and Activities

Solution. Assumption: activities are uniquely identified their names (you could make other assumptions; it's O.K. as long as you stated them clearly).

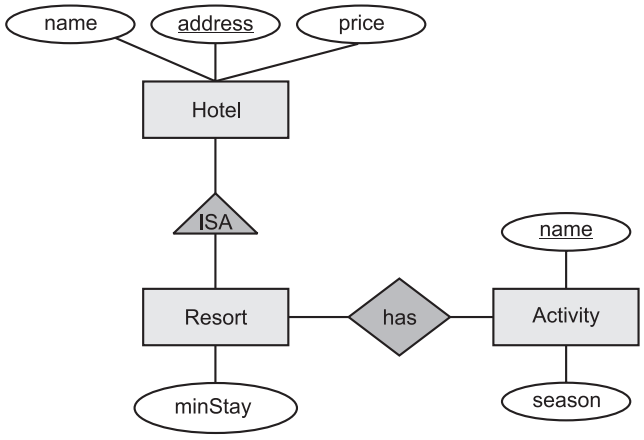

FIGURE 2.43

Problem 18. Design an E/R diagram for an application domain consisting of the following entity sets:

- Projects. Attributes: name, budget
- Teams. Attributes: team_name
- Empolyees. Attributes: name, phone_number
- Consultants. Attributes: name, phone_number, hourly_rate

And the following relationships:

- Each team works on one or more projects.
- Each project has an auditor, who is an employee
- Consultants are employees

Your answer should consist of an E/R diagram with entity sets, attributes (make sure you create appropriate keys: you may incorporate new attributes if needed), relationships, and inheritance.

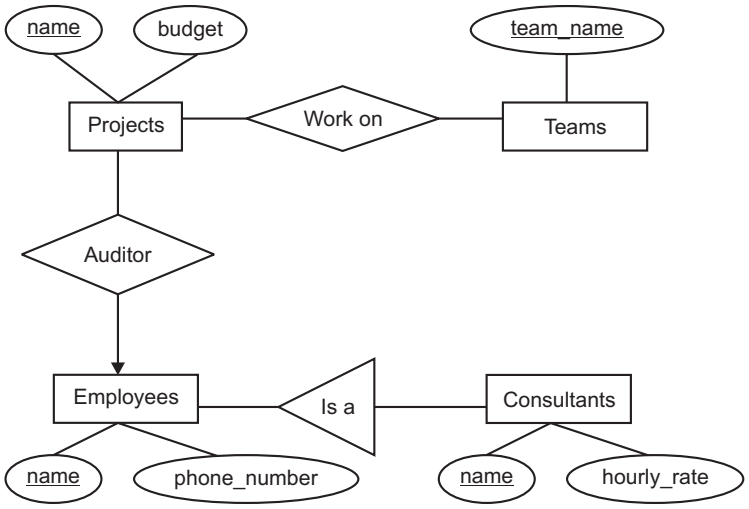

FIGURE 2.44

The E/R diagram shown below is for the following scenario: A publishing company produces academic books on various subjects. Authors who specialise in one or more particular subject write books. The company employs a number of editors who do not have particular specializations but who take sole responsibility for editing one or more publications. A publication covers a single subject area but may be written by one or more author – the contribution of each author is recorded as a percentage for the purposes of calculating royalties. Give a reason about which relation has the incorrect cardinality in the E/R diagram.

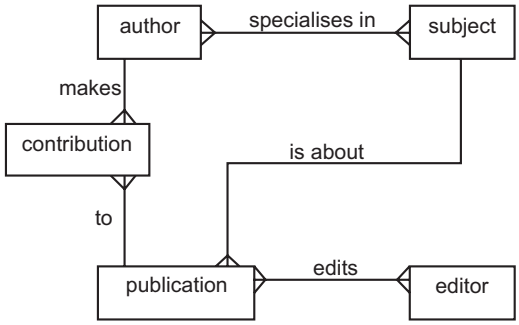

FIGURE 2.45

Solution. From the specification, "...[Editors] take sole responsibility for editing one or more publications...". Thus an editor can edit more than one publication (one to many), but each publication has only a single editor. Thus the relationship for "edits" should be one to many, not many to many.