1.1 This chapter has described several major advantages of a database system. What are two disadvantages? 2

1.2 List five ways in which the type declaration system of a language such as Java or C++ differs from the data definition language used in a database. 2

1.3 List six major steps that you would take in setting up a database for a particular enterprise. 2

1.6 Keyword queries used in Web search are quite different from database queries. List key differences between the two, in terms of the way the queries are specified, and in terms of what is the result of a query. 2

1.7 List four applications you have used that most likely employed a database system to store persistent data. 5

1.8 List four significant differences between a file-processing system and a DBMS. 5

1.9 Explain the concept of physical data independence, and its importance in database systems. 5

1.10 List five responsibilities of a database-management system. For each responsibility, explain the problems that would arise if the responsibility 5

1.11 List at least two reasons why database systems support data manipulation using a declarative query language such as SQL, instead of just providing a a library of C or C++ functions to carry out data manipulation. 5

1.13 What are five main functions of a database administrator? 5

1.14 Explain the difference between two-tier and three-tier architectures. Which is better suited for Web applications? Why? 10

2.1 Consider the relational database of Figure 2.14. What are the appropriate primary keys?

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

Figure 2.14 Relational database for Exercises 2.1, 2.7, and 2.12.

5

2.2 Consider the foreign key constraint from the *dept_name* attribute of *instructor* to the *department* relation. Give examples of inserts and deletes to these relations, which can cause a violation of the foreign key constraint. 5

2.6 Consider the following expressions, which use the result of a relational algebra operation as the input to another operation. For each expression, explain in words what the expression does.

a. $\sigma_{year \geq 2009}(takes) \bowtie student$

b. $\sigma_{year \geq 2009}(takes \bowtie student)$

c. $\Pi_{ID, name, course\_id}(student \bowtie takes)$

5

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

Figure 2.14 Relational database for Exercises 2.1, 2.7, and 2.12.

2.7 Consider the relational database of Figure 2.14. Give an expression in the relational algebra to express each of the following queries:

a. Find the names of all employees who live in city "Miami".

b. Find the names of all employees whose salary is greater than \$100,000.

c. Find the names of all employees who live in "Miami" and whose salary is greater than \$100,000.

5

branch(branch_name, branch_city, assets)
customer (customer_name, customer_street, customer_city)
loan (loan_number, branch_name, amount)
borrower (customer_name, loan_number)
account (account_number, branch_name, balance)
depositor (customer_name, account_number)

Figure 2.15 Banking database for Exercises 2.8, 2.9, and 2.13.

2.7 Consider the relational database of Figure 2.14. Give an expression in the relational algebra to express each of the following queries:

a. Find the names of all employees who live in city "Miami".

b. Find the names of all employees whose salary is greater than $100,000.

c. Find the names of all employees who live in "Miami" and whose salary is greater than $100,000. 5

branch(branch_name, branch_city, assets)
customer (customer_name, customer_street, customer_city)
loan (loan_number, branch_name, amount)
borrower (customer_name, loan_number)
account (account_number, branch_name, balance)
depositor (customer_name, account_number)

Figure 2.15 Banking database for Exercises 2.8, 2.9, and 2.13.

2.9 Consider the bank database of Figure 2.15.

a. What are the appropriate primary keys?

b. Given your choice of primary keys, identify appropriate foreign keys. 5

2.11 Describe the differences in meaning between the terms *relation* and *relation schema*. 2

employee (person_name, street, city)
works (person_name, company_name, salary)
company (company_name, city)

Figure 2.14 Relational database for Exercises 2.1, 2.7, and 2.12.

2.12 Consider the relational database of Figure 2.14. Give an expression in the relational algebra to express each of the following queries:

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

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

c. Find the names, street address, and cities of residence of all employees who work for "First Bank Corporation" and earn more than $10,000. 5

branch(branch_name, branch_city, assets)
customer (customer_name, customer_street, customer_city)
loan (loan_number, branch_name, amount)
borrower (customer_name, loan_number)
account (account_number, branch_name, balance)
depositor (customer_name, account_number)

Figure 2.15 Banking database for Exercises 2.8, 2.9, and 2.13.

2.13 Consider the bank database of Figure 2.15. Give an expression in the relational algebra for each of the following queries:

a. Find all loan numbers with a loan value greater than $10,000.

b. Find the names of all depositors who have an account with a value greater than $6,000.

c. Find the names of all depositors who have an account with a value greater than $6,000 at the "Uptown" branch. 5

2.14 List two reasons why null values might be introduced into the database. 2

2.15 Discuss the relative merits of procedural and nonprocedural languages. 5

Practice Exercises

7.1 Construct an E-R diagram for a car insurance company whose customers
own one or more cars each. Each car has associated with it zero to any
number of recorded accidents. Each insurance policy covers one or more
cars, and has one or more premium payments associated with it. Each 10
payment is for a particular period of time, and has an associated due date,
and the date when the payment was received.

7.2 Consider a database used to record the marks that students get in different
exams of different course offerings (sections). 10

a. Construct an E-R diagram that models exams as entities, and uses a ternary relationship, for the database.

b. Construct an alternative E-R diagram that uses only a binary relationship between *student* and *section*. Make sure that only one relationship exists between a particular *student* and *section* pair, yet you can represent the marks that a student gets in different exams.

7.3 Design an E-R diagram for keeping track of the exploits of your favorite sports team. You should store the matches played, the scores in each match, 10
the players in each match, and individual player statistics for each match. Summary statistics should be modeled as derived attributes.

7.4 Consider an E-R diagram in which the same entity set appears several times, with its attributes repeated in more than one occurrence. Why is allowing 10
this redundancy a bad practice that one should avoid?

7.5 An E-R diagram can be viewed as a graph. What do the following mean in terms of the structure of an enterprise schema?

a. The graph is disconnected. 10

b. The graph has a cycle.

7.6 Consider the representation of a ternary relationship using binary relationships as described in Section 7.7.3 and illustrated in Figure 7.27b (attributes not shown).

10

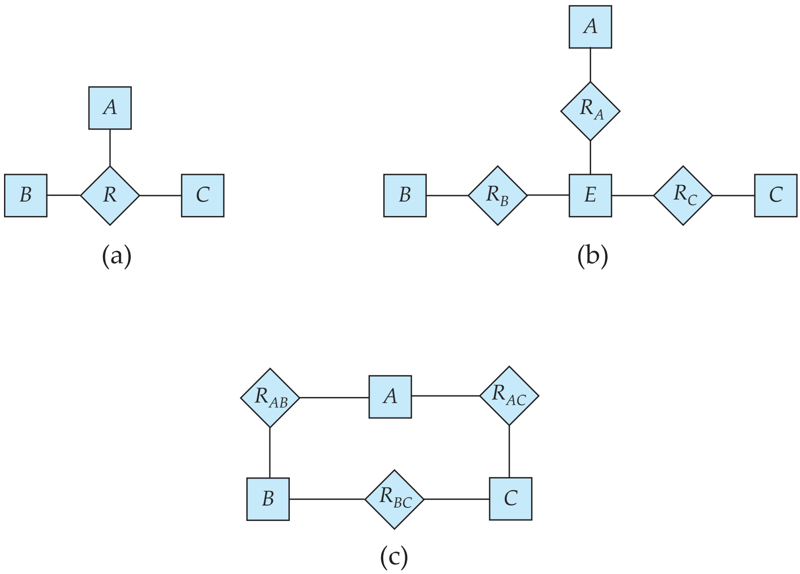


Figure 7.27 E-R diagram for Practice Exercise 7.6 and Exercise 7.24.

a. Show a simple instance of E, A, B, C, R_A, R_B, and R_C that cannot correspond to any instance of A, B, C, and R.

b. Modify the E-R diagram of Figure 7.27b to introduce constraints that will guarantee that any instance of E, A, B, C, R_A, R_B, and R_C that satisfies the constraints will correspond to an instance of A, B, C, and R.

c. Modify the translation above to handle total participation constraints on the ternary relationship.

d. The above representation requires that we create a primary-key attribute for E. Show how to treat E as a weak entity set so that a primary-key attribute is not required.

7.7 A weak entity set can always be made into a strong entity set by adding to
its attributes the primary-key attributes of its identifying entity set. Outline 5
what sort of redundancy will result if we do so.

7.8 Consider a relation such as *sec_course*, generated from a many-to-one rela-
tionship *sec_course*. Do the primary and foreign key constraints created on 5
the relation enforce the many-to-one cardinality constraint? Explain why.

7.9 Suppose the *advisor* relationship were one-to-one. What extra constraints
are required on the relation *advisor*to ensure that the one-to-one cardinality 5
constraint is enforced?

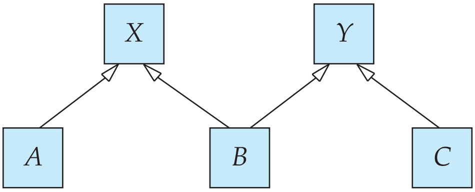
7.10 Consider a many-to-one relationship R between entity sets A and B. Sup-
pose the relation created from R is combined with the relation created from 5
A. In SQL, attributes participating in a foreign key constraint can be null.
Explain how a constraint on total participation of A in R can be enforced
using **not null** constraints in SQL.

7.11 In SQL, foreign key constraints can only reference the primary key attributes
of the referenced relation, or other attributes declared to be a super key us-
ing the **unique** constraint. As a result, total participation constraints on a 10
many-to-many relationship (or on the "one" side of a one-to-many rela-
tionship) cannot be enforced on the relations created from the relationship,
using primary key, foreign key and not null constraints on the relations.

a. Explain why.

b. Explain how to enforce total participation constraints using complex check constraints or assertions (see Section 4.4.7). (Unfortunately, these features are not supported on any widely used database currently.)

7.12 Figure 7.28 shows a lattice structure of generalization and specialization (attributes not shown). For entity sets A, B, and C, explain how attributes



10

Figure 7.28 E-R diagram for Practice Exercise 7.12.

are inherited from the higher-level entity sets X and Y. Discuss how to handle a case where an attribute of X has the same name as some attribute of Y.

7.13 **Temporal changes**: An E-R diagram usually models the state of an enterprise at a point in time. Suppose we wish to track *temporal changes*, that is, changes to data over time. For example, Zhang may have been a student between 1 September 2005 31 May 2009, while Shankar may have had instructor Einstein as advisor from 31 May 2008 to 5 December 2008, and again from 1 June 2009 to 5 January 2010. Similarly, attribute values of an entity or relationship, such as *title* and *credits* of *course*, *salary*, or even *name* of *instructor*, and *tot_cred* of *student*, can change over time.

One way to model temporal changes is as follows. We define a new data
type called **valid_time**, which is a time-interval, or a set of time-intervals. 10
We then associate a *valid_time* attribute with each entity and relationship, recording the time periods during which the entity or relationship is valid. The end-time of an interval can be infinity; for example, if Shankar became a student on 2 September 2008, and is still a student, we can represent the end-time of the *valid_time* interval as infinity for the Shankar entity. Similarly, we model attributes that can change over time as a set of values, each with its own *valid_time*.

a. Draw an E-R diagram with the *student* and *instructor* entities, and the *advisor* relationship, with the above extensions to track temporal changes.

b. Convert the above E-R diagram into a set of relations.

It should be clear that the set of relations generated above is rather complex, leading to difficulties in tasks such as writing queries in SQL. An alternative approach, which is used more widely is to ignore temporal changes when designing the E-R model (in particular, temporal changes to attribute values), and to modify the relations generated from the E-R model to track temporal changes, as discussed later in Section 8.9.

Exercises

7.14 Explain the distinctions among the terms primary key, candidate key, and superkey. 5

7.15 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 the various tests and examinations conducted. 10

7.16 Construct appropriate relation schemas for each of the E-R diagrams in Practice Exercises 7.1 to 7.3. 10

7.17 Extend the E-R diagram of Practice Exercise 7.3 to track the same information for all teams in a league. 10

7.18 Explain the difference between a weak and a strong entity set. 5

7.19 We can convert any weak entity set to a strong entity set by simply adding appropriate attributes. Why, then, do we have weak entity sets? 5

7.20 Consider the E-R diagram in Figure 7.29, which models an online bookstore.

a. List the entity sets and their primary keys.

b. Suppose the bookstore adds Blu-ray discs and downloadable video to its collection. The same item may be present in one or both formats, with differing prices. Extend the E-R diagram to model this addition, ignoring the effect on shopping baskets. 10

c. Now extend the E-R diagram, using generalization, to model the case where a shopping basket may contain any combination of books, Blu-ray discs, or downloadable video.

7.21 Design a database for an automobile company to provide to its dealers to assist them in maintaining customer records and dealer inventory and to assist sales staff in ordering cars.

Each vehicle is identified by a vehicle identification number (VIN). Each individual vehicle is a particular model of a particular brand offered by the company (e.g., the XF is a model of the car brand Jaguar of Tata Motors). Each model can be offered with a variety of options, but an individual car may have only some (or none) of the available options. The database needs to store information about models, brands, and options, as well as information about individual dealers, customers, and cars. 10

Your design should include an E-R diagram, a set of relational schemas, and a list of constraints, including primary-key and foreign-key constraints.

7.22 Design a database for a world-wide package delivery company (e.g., DHL or FedEX). The database must be able to keep track of customers (who ship items) and customers (who receive items); some customers may do both. 10

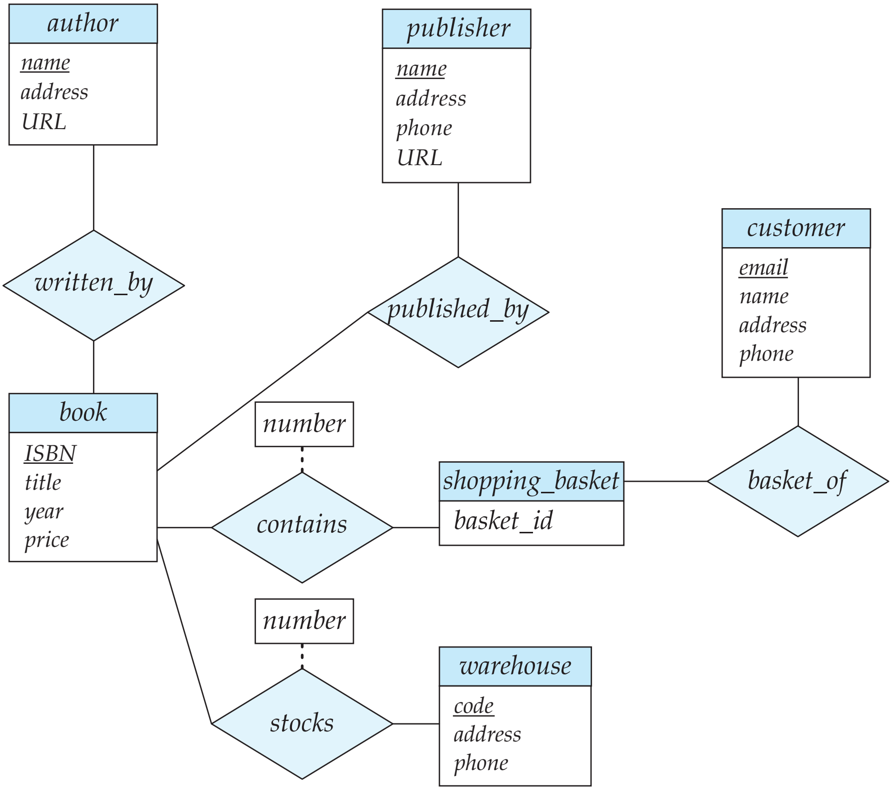


Figure 7.29 E-R diagram for Exercise 7.20.

Each package must be identifiable and trackable, so the database must be able to store the location of the package and its history of locations. Locations include trucks, planes, airports, and warehouses.

Your design should include an E-R diagram, a set of relational schemas, and a list of constraints, including primary-key and foreign-key constraints.

7.23 Design a database for an airline. The database must keep track of customers and their reservations, flights and their status, seat assignments on individual flights, and the schedule and routing of future flights.

Your design should include an E-R diagram, a set of relational schemas, and a list of constraints, including primary-key and foreign-key constraints.

7.26 Design a generalization–specialization hierarchy for a motor vehicle sales company. The company sells motorcycles, passenger cars, vans, and buses. Justify your placement of attributes at each level of the hierarchy. Explain why they should not be placed at a higher or lower level. 10

7.27 Explain the distinction between condition-defined and user-defined constraints. Which of these constraints can the system check automatically? 10 Explain your answer.

7.28 Explain the distinction between disjoint and overlapping constraints. 10

7.29 Explain the distinction between total and partial constraints. 5