

CHAPTER 1: DATABASES AND DATABASE USERS**Answers to Selected Exercises**

1.8 - Identify some informal queries and update operations that you would expect to apply to the database shown in Figure 1.2.

Answer:

(a) (Query) List the names of all students majoring in Computer Science.

(b) (Query) What are the prerequisites of the Database course?.

(c) (Query) Retrieve the transcript of Smith. This is a list of <CourseName, SectionIdentifier, Semester, Year, Grade> for each course section that Smith has completed.

(d) (Update) Insert a new student in the database whose Name=Jackson, StudentNumber=23, Class=1 (freshman), and Major=MATH.

(e) (Update) Change the grade that Smith received in Intro to Computer Science section 119 to B.

1.9 - What is the difference between controlled and uncontrolled redundancy?

Answer:

Redundancy is when the same fact is stored multiple times in several places in a database. For example, in Figure 1.5(a) the fact that the name of the student with StudentNumber=8 is Brown is stored multiple times. Redundancy is controlled when the DBMS ensures that multiple copies of the same data are consistent; for example, if a new record with StudentNumber=8 is stored in the database of Figure 1.5(a), the DBMS will ensure that StudentName=Smith in that record. If the DBMS has no control over this, we have uncontrolled redundancy.

1.10 - Specify all the relationships among the records of the database shown in Figure 1.2.

Answer:

(a) Each SECTION record is related to a COURSE record.

(b) Each GRADE_REPORT record is related to one STUDENT record and one SECTION record.

(c) Each PREREQUISITE record relates two COURSE records: one in the role of a course and the other in the role of a prerequisite to that course.

1.11 - Give some additional views that may be needed by other user groups for the database shown in Figure 1.2.

Answer:

(a) A view that groups all the students who took each section and gives each student's grade. This may be useful for printing the grade report for each section for the

university administration's use.

(b) A view that gives the number of courses taken and the GPA (grade point average) for each student. This may be used to determine honors students.

1.12 – Cite some examples of integrity constraints that you think can apply to the database shown in Figure 1.2.

Answer:

We give a few constraints expressed in English. Following each constraint, we give its type in the relational database terminology that will be covered in Chapter 6, for reference purposes.

- (a) The StudentNumber should be unique for each STUDENT record (key constraint).
- (b) The CourseNumber should be unique for each COURSE record (key constraint).
- (c) A value of CourseNumber in a SECTION record must also exist in some COURSE record (referential integrity constraint).
- (d) A value of StudentNumber in a GRADE_REPORT record must also exist in some STUDENT record (referential integrity constraint).
- (e) The value of Grade in a GRADE_REPORT record must be one of the values in the set {A, B, C, D, F, I, U, S} (domain constraint).
- (f) Every record in COURSE must have a value for CourseNumber (entity integrity constraint).
- (g) A STUDENT record cannot have a value of Class=2 (sophomore) unless the student has completed a number of sections whose total course CreditHours is greater than 24 credits (general semantic integrity constraint).

1.13 - Give examples of systems in which it may make sense to use traditional file processing instead of a database approach.

Answer:

1. Small internal utility to locate files
2. Small single user application that does not require security (such as a customized calculator or a personal address and phone book)
3. Real-time navigation system (with heavy computation and very little data)
4. The students may think of others.

1.14 - Consider Figure 1.2.

- a. If the name of the 'CS' (Computer Science) Department changes to 'CSSE' (Computer Science and Software Engineering) Department and the corresponding prefix for the course number also changes, identify the columns in the database that would need to be updated.
- b. Can you restructure the columns in COURSE, SECTION, and PREREQUISITE tables so that only one column will need to be updated?

Answer:

a. The following columns will need to be updated.

Table	Column(s)
STUDENT	Major
COURSE	CourseNumber and Department
SECTION	CourseNumber
PREREQUISITE	CourseNumber and PrerequisiteNumber

b. You should split the following columns into two columns:

Table	Column	Split Columns
COURSE	CourseNumber	CourseDept and CourseNum
SECTION	CourseNumber	CourseDept and CourseNum
PREREQUISITE	CourseNumber	CourseDept and CourseNum
PREREQUISITE	PrerequisiteNumber	PreReqDept and PreReqNum

Note that in the COURSE table, the column CourseDept will not be needed after the above change, since it is redundant with the Department column.