

Tutorial - Week 11

Part One: Database design, SQL, and Views

A European company sells items worldwide and maintains a catalogue in several currencies.

A first table contains the product catalogue. Each product has an ID (primary key), a name, a unit price (in euro) and a stock level. A second table contains client details. Each client has an ID (primary key), a name, and an address.

A third table stores information on sales. Each entry has a client ID, a product ID, and a number of units sold. The primary key for this table is (client ID, product ID).

1. Create these tables, using constraints for the attribute values and for considerations on primary and foreign keys.
2. Add entries in these tables.
3. Create a view which will contain, for each sell, the client name, the product name, and the value of the sale in three currencies (euro, dollar, pound). Assume that $\text{€}1 = \$1.31 = \text{£}0.88$.
4. Try SELECT queries on this view.
5. Try adding an entry in this view. Explain what happens and why.
6. Delete this view, and create a new one, replacing the client and product names by the respective IDs.
7. Try adding an entry in this view. Is there any difference in behaviour, compared to the previous view. Why?

Part Two: Normalisation

Question 1:

A college keeps details about each registered student and the modules this student studied:

- regno - registration number
- n - student name
- a - student address
- tno - tutor number

- tna - tutor name
- dc - diploma code
- dn - diploma name
- mc - module code
- mn - module name
- res - module exam result

These details are part of the following DETAILS relation:

details(regno,n,a,tno,tna,dc,dn,(mc,mn,res))

Reduce this relation to the third normal form, explaining each step and assuming the following functional dependencies:

- dc -> dn
- tno -> tna
- mc,mn -> res
- n -> a
- mc -> mn

(You may want to use functional dependency diagrams)

Question 2:

Reduce the following relation to BCNF, detailing all the steps involved.

Supplier(sno,sname,saddress,(partno, partdesc,(custid,custname,custaddr,quantity)))

You can assume the following functional dependencies:

- sno -> sname,saddr
- sno,partno -> partdesc
- sno,partno,custid -> quantity
- sname -> sno
- custid -> custname,custaddr

You may also assume that:

- Suppliers supply many parts to many customers.
- Each customer deals with only one supplier.
- Supplier names are unique.
- Customer names are not unique.

Question 3:

A firm wishes to keep the following data for each employee:

- employee number
- employee name
- employee address
- salary
- current job code
- job history (job promotion code + year)
- office location
- telephone number
- project number
- project name
- task number
- task name
- project budget
- task expenditure to date
- department number
- department name

For each employee, there can be none, one or more “job promotion code + year” entries. The office location uniquely depends on the telephone number, and there may be more than one employee using the same telephone and more than one telephone in the one office. Tasks are numbered uniquely only within each project. An employee may be concurrently assigned to more than one project and task, but belongs to one department.

Reduce this data to third normal form, detailing each step, and any further assumptions you are making.