How to teach databases?

M. Dvořáková
Charles University Prague, Faculty of Mathematics and Physics, Prague, Czech Republic.

Abstract. This paper introduces how to teach databases at secondary schools. Databases are not fully standard topic taught at secondary school, but students certainly should learn databases, because data organization and data processing are important nowadays. I describe teaching plan (motivation, definitions explanation, database design and practical work), and I give an example of database in library.

Preface

Teaching informatics at secondary school continues teaching informatics at basic school, therefore we expect that students know something elementary about computers. Secondary school’s target is to improve current skills and to extend theoretical knowledge in the field of informatics and computer technology.

Set of topics, which can be gone over at secondary school, notably depends on total time allocation for informatics at various types of school. For example two lessons a week are given in the first year at four-year grammar school. In the next years informatics is only as optional or facultative subject for interested students. At secondary vocational schools it depends on branch of study, for example schools of economics have even two lessons a week every year of study.

Standard taught topics include texts writing, tables and presentation creation, images adjustment, eventually web pages design.

Databases at secondary school, yes or no?

Databases are not fully standard topic taught at secondary school. In my view reasons are: low time allocation, little qualified teacher, who don’t understand databases, and conversely very qualified teacher, who seems databases too difficult for secondary-school students.

If don’t remains time for databases in prescribed lessons, they should take place in optional seminars. During my practice I have verified that six lessons are enough for meaningful acquaintance with databases. It’s only that, to find necessary minimum of knowledge, which teacher is able to explain and which students are able to realize.

Where the problem lies? While texts writing can be managed at user level, situation with databases is a bit complicated. At database design is necessary to precisely devise database structure. One, who will work with this prepared database, can be only ordinary user. Students in lessons should attempt both approaches.

I think students certainly should learn databases, because data organization and data processing are important nowadays and we meet databases almost at every turn in life.

Teaching materials

Present textbooks of informatics for secondary schools (eg. [2]) contains chapter Database, but they are restricted only for MS Access, in the text are prefered clicks at buttons and menu item to thinking. About mastery in MS Access exists many books, one of them is [1].

I don’t think, that to teach students how to know a lot about a specific software is important. So I developed following teaching plan, which is generally usable without software dependence.

With respect to limited scope of this article I demonstrate this topic only at one detailed example. For teaching at school is of course necessary to have prepared sufficient stock of examples.

Teaching plan

Teaching every new thematic part should start from good motivation. Motivation to databases is easy, one need adduce practical examples and students will agree with us, that they already know
we can imagine databases as various lists of persons, things or events. Database is for example library with list of books, registered readers and seeking book information on web interface, further it is patient’s index at doctor with records of medical examinations and last but not least example is school information system with classes, students, teachers and timetable administration. Thus database knowledge is applicable in many professions, and if only at user level, so it’s bad to have general view about database structure and especially about database features.

Inquiring student could ask a question „Why don’t store these various lists in tables in Excel?“. The answer is that spreadsheet calculator can also process data, but only smaller amount of data. Spreadsheet calculator is apposite for tables, which include calculations. Database is apposite for static text data storage, what means that data aren’t results of computation. Various calculations are performed out of tables – in queries. Databases enable more comfortably sorting data, searching by various criterions, updating and joining.

After motivation we continue with basic definitions explanation, which is essential for practical working.

Exact definition of database isn’t necessary, for secondary-school students is idea of motivation enough. We can mention some of database characteristic: Database is a greater information grouping, data have certain logic and data can be processing. Data are centralizing at one place, what enables easy orientation and searching in database. Stored data doesn’t depend on user’s program.

Essentials of databases are tables, which contains data. Each table is list of columns, which contains information of given type. For example, table Readers can be composed from columns First name, Last name, Birthday, Address. We try to store data in logical blocks as small as possible. Take notice that we store first name and last name in different columns. Kind of information stored in column is determined by a column data type. Most used data types are string, number or date. In our case First name will be string, whereas Birthday matches data type.

A reader is stored in table as single row with filled first name, last name, birth date and address. A row is also called a record. Each record should be unique, because we needn’t state when one reader is stored in table more then once or two readers have same identifier. The solution is marking columns with unique values as primary key. After that, insertion of two rows with same primary key is denied.

We should not design tables with too few or too many columns. As I’ve said a table consists from attributes of one logical subject. It our case it means, that one table saves data about readers and another table saves information about books.

One table database is not real database. The purpose of database usage is in table bindings. We can use data from all tables together. Binding exists between two tables, which both contains column with same information. For example table Readers have identifier column, so table with book borrowings contains column with same data type and the value is identifier of reader, which made that borrow. Each borrowing easily points to involved reader.

Tables A and B can be associated in three ways:

- 1:1 – At most one B record belongs to each A record and vice versa. This binding is not so often. It makes sense for dividing huge table in two pieces (security reason for example) or for detaching optional attribute domain.
- 1:N – A number of B records belongs to each A record and each B record belongs only to one A record. This way is most used, for example a reader can have more then one borrow.
- N:M – A number of B records belongs to each A record and vice versa. For example a reader can bespeak more than one book, each book can be reserved by more than one reader.

Table binding can be also called relation. Be aware that word relation in this way means binding not mathematical relation.

Remark: Database can be defined in mathematical manner. Database is a finite set of relation schemas $R_1$, ..., $R_k$. Relation schema $R_i(A_1;D_1, ... A_n;D_n)$ is in a fact table with columns $A_i$ with assigned data types $D_i$. A record is an instance of relation, i. e. $n$-tuple $(a_1, ..., a_n)$, where $a_i\in D_i$. These databases are referred also as relation databases. As you can see, this style of definition, which is
presented in academical textbooks (eg. in [3]), is very formal and is not adequate for teaching at secondary school.

Next we will follow with database design. We want to capture part of reality in form of several joined tables. It’s very important think over database structure first. Think over which data will be stored in which table and how will be tables joined. Structure of filled database cannot be easily changed. Design should be made with respect to way of database usage and to best satisfy of user requirements.

Result of design should be description of real objects, attributes of the objects and associations among these objects. Example mentioned above, the simple library database, can be represented by three tables:

- Readers (ID_reader, First name, Last name, Birthday, Street, Town)
- Books (ID_book, Author, Title, Year of issue, Publisher)
- Borrowings (ID_reader, ID_book, Date of borrowing, Back)

Primary keys are marked by highlighting. We assume only one exemplar of each book. Table Borrowings stores history of borrows and simplify relation between readers and books. During the library existence one reader can borrow more than one book and a book is gradually borrowed to different readers, so direct binding between tables Readers and Books is N:M. This bind type is realized by auxiliary third table, in our case table Borrowings. This auxiliary table is associated to both tables with 1:N binding.

Depending on students branch of study and on time allocation can be database design leaving out. Database structure prepare the teacher, students suggest primary keys and column data type.

Now we start practical work on computer. In chosen program (Microsoft Office Access or OpenOffice.org Base) we show to students, how to design tables and data types, how to mark column with primary key, how to join tables and how to insert records.

We won’t waste the time with tables filling, but we will prepare sufficient amount of meaningful data, which students will copy and then try database processes.

Right in table we can sort, search and pick off by a filter. But if we need have stored results, then we compose a query. Queries are database components, we can them repeatedly activate and of course modify. Queries advantage is working with data from several tables. For example a query is “Which books are borrowed by reader older than fifteen years?”. 

Next database component are forms. Forms are useful for data scanning and data formatting in a graphic window. Just via forms meets users databases.

Conclusion

Teaching databases can be interesting for students, because it is topic which they didn’t meet at basic school and they are motivated with practice usage. Teacher can adjust this topic to his requirements and students skills as I have tried to demonstrate in this text. I wish this contribution would be inspiration for you not only for teaching databases.

References