

МЕТОДИЧЕСКИ АСПЕКТИ НА ОБУЧЕНИЕТО ПО БАЗИ ДАННИ

Димитрина Делийска¹, Ирина Христова²

¹ Минно-геоложки университет "Св. Иван Рилски", 1700София, e-mail: dimitrina.deliyska@mgu.bg

² Минно-геоложки университет "Св. Иван Рилски", 1700София, e-mail:irinahr@mgu.bg

РЕЗЮМЕ Успехът във всички сфери на обществения живот изисква предприемането на адекватни решение относно дадено събитие (явление). Това предполага да се разполага с необходимата информация, организирана и структурирана в база данни. Последната е основен компонент на информационните системи.

Изучаването на бази данни в МГУ "Св. Иван Рилски" се извършва в учебни дисциплини, които се изучават в различни специалности. Практическото обучение на студентите се осъществява чрез разработването на самостоятелен курсов проект, в основата на който е изграждане на информационна система в определена предметна област. Създаването и обработването на базата данни в отделните специалности се осъществява с различни системи за управление на база данни (СУБД): MS-Access или MySQL.

В доклада са представени методически аспекти в практическото обучение на студентите, като акцентът е поставен върху формиране на знания за релационна база данни и нейното проектиране независимо от използваната СУБД.

Ключови думи: База данни, Информационни системи, Модели на данни

METHODOLOGICAL ASPECTS OF TRAINING IN DATABASES

Dimitrina Deliyiska¹, Irina Hristova²

¹ University of Mining and Geology "St. Ivan Rilski", 1700 Sofia, e-mail dimitrina.deliyska@mgu.bg

² University of Mining and Geology "St. Ivan Rilski", 1700 Sofia, e-mail irinahr@mgu.bg

ABSTRACT Success in all areas of public life requires the adoption of an adequate decision on an event (phenomenon). This implies having the necessary information which needs to be organized and structured in a database. The latter is an essential component of information systems.

The study of databases at the University of Mining and Geology "St. Ivan Rilski" is carried out in academic disciplines that are taught in different courses of studies. The practical training of students is carried out through the development of an individual course project at the base of which is building an information system in a particular subject area. Within the various courses of studies, creating and processing a database is carried out by means of different database management systems (DBMS): MS-Access or MySQL.

This report presents some methodological aspects in the practical training of students. Emphasis is put on the formation of knowledge concerning relational databases and how to design them irrespective of the underlying DBMS. The sequence of actions taken in the development of an information system using MS-Access is presented.

Keywords: databases, information systems, data models

ВЪВЕДЕНИЕ

По дефиниция Информационна система (ИС), съгласно Международната организация по стандартизация (International Organization for Standardization) е "взаимосвързана съвкупност от средства, методи и персонал, използвана за обработка, съхраняване на данни с цел предоставяне на информация за достигане на поставена цел". Съществена част на всяка ИС е нейната база данни (БД).

В доклада е представена методика за практическото обучение на студентите по БД. Целта е да се формират теоретични знания по БД и компетенции за нейното създаване. Това се постига чрез разработването на курсов проект, свързан с определена предметна област. Подходът е избран изхождайки от фазите за изработване на един проект, определени в софтуерното инженерство: анализ, проектиране, реализация, тестване и експлоатация.

БАЗА ДАННИ. МОДЕЛИ НА ДАННИТЕ

При формирането на понятията от теория за БД се използват следните две основания:

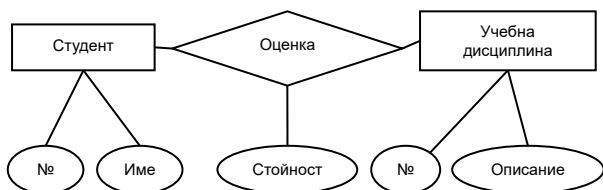
1. Дефиниция на БД – организирана съвкупност от взаимосвързани и заедно съхранявани данни при минимално количество дублиране с цел оптималното им използване за едно или повече приложения.
2. Анализ на целите на ИС и изискванията към нейното функциониране

Проектирането на БД започва със събирането на данни относно основните понятия (обекти) в предметната област и процесите, които протичат между тях. Всеки обект се отличава с определени свойства, като само част от тях се явяват съществени за ИС. Представянето на свойствата, в зависимост от техния характер, се извършва с различен тип информация: числова, текстова, тип – дата и т.н. Тук се формират основните понятия от теорията на БД:

- ❑ **Поле, атрибут (field, attribute)** - описва свойство на даден обект. От гледна точка на теорията за БД това е най-малката неделима информация. Всяко поле има: уникално име; тип – в зависимост от представяната информация и дължина;
- ❑ **Запис (record)** – описва един обект – логически свързана последователност от полета. Всеки запис е уникален;
- ❑ **Файл (file)** – логически свързана последователност от екземпляри на записа;
- ❑ **БД (database)** - ...логически свързаност последователност от файлове.

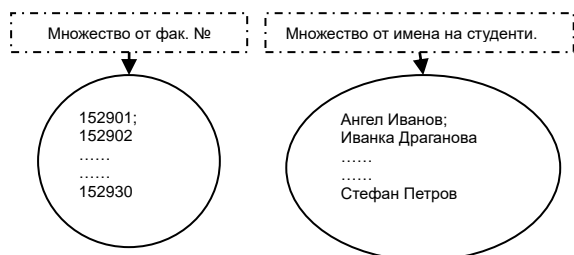
Възможните модели данни са:

- ❑ Мрежов модел – основава се на таблици и представяне чрез граф;
- ❑ Йерархичен модел – частен случай на мрежовия. Данните са разпределени в сегменти, йерархично свързани в дърво;
- ❑ Реляционен модел – основава се на реляционната алгебра (Cood [1]);
- ❑ ERM (Entity RelationShipModel) – при този модел се изграждат диаграмите на Р.Р.Chen. Обектите (entity) се разглеждат като множество от типови елементи /студенти, преподаватели, клиенти, доставчици и т.н./ и се представят с правоъгълник. Техните атрибути /номер, наименование, описание и т.н./ - с овали. Връзката между обектите /оценяване, продажби и т.н./ - чрез ромб.



Фиг. 1 ERM схема

На този етап се определят и ресурсите за създаването на БД. Съгласно учебните програми за изучаване са предвидени две системи за управление на БД: MS Access и MYSQL. И двете поддържат реляционния модел на БД. Проектът, разработван от студентите, също използва реляционния модел на БД. Това налага да се доуточнят понятията, свързани с реляционните БД. Нека са дадени две множества: множеството на факултетните номера и множеството от имена на студенти



Фиг. 2

Понятието релация е математическа категория, и се дефинира като декартовото произведение на две или повече множества.

Нека с A означим множеството на факултетните номера, а с B – множеството от имената на студентите, тогава всяко подмножество $R = A \times B$ се нарича двоична релация от A в B . На всеки факултетен номер от A се съпоставя точно един студент от B .

Наредената двойка < факултетен №, име на студент > $\in R$ се нарича двоична релация.

Например за студентите в МГУ могат да се опишат със следната релация:

Студенти < фак. №, име, бащино име, фамилия, код на специалност, име на специалност, факултет, код на дисциплина, име на дисциплина, оценка, ПК, населено място, адрес >
--

Всяка релацията в реляционните БД се представя от таблица със следните основни свойства [2]:

- ❑ всеки елемент на таблицата (клетка) съдържа само един елемент данни (фак.№, име, бащино име, фамилия и т.н.);
- ❑ стълбовете (колони) са еднородни, тоест съдържат данни от един и същи тип (само факултетни номера, само имена и т.н.);
- ❑ на стълбовете са присвоени уникални имена фак.№, име, бащино име, фамилия ...);
- ❑ липсват два еднакви реда;
- ❑ при работа с таблицата стълбовете и редовете са независими от тяхното съдържание.

Тук се дефинира и понятието първичен ключ. Той определя уникалността на реда. Може да бъде едно поле (обикновен ключ) или множество от полета (съставен ключ).

Определят се съответствия между понятията, използвани в теорията на реляционните бази и тези, които се използват в SQL базите данни:

- ❑ поле \leftrightarrow стълб, колона (column);
- ❑ запис, кортеж (tuple) \leftrightarrow ред (row);
- ❑ релация (relation) \leftrightarrow таблица (table).

Основни предимства на реляционните БД:

- ❑ Достъпът до произволна информация в таблиците се осъществява чрез селекция (избор на редове) и проекция (избор на колона);
- ❑ Свързването на различни таблици от БД се извършва чрез еднакви ключови полета;
- ❑ Добавянето на нови редове и колони е лесно.

Към недостатъците се причисляват:

- ❑ Изисква се повече памет: дължината на полетата е фиксирана и съобразена с най-голямата стойност, съхранявана в тях; ключовите полета се повтарят нееднократно;
- ❑ Достъпът до записите (редовете) е твърде бавен поради последователното обхождане на таблиците.

Реляционният модел на БД поставят определени изисквания:

- ❑ Отсъствие на излишества – данните се въвеждат и съхраняват само веднъж;
- ❑ Непротиворечивост на данните в БД – в противен случай възникват аномалии при включване, изключване и модифициране на данните.

Много съществен методически аспект е формиране на понятието нормализация на БД.

Нормализацията е процес на декомпозиране на множествата от елементи данни в стабилни подмножества, чрез изследване и разширяване на вътрешните им връзки. Нормализация е процес на опростяване на връзките между данните и преминава през следните етапи (форми):

- Ненормализирана БД – БД данни, която съдържа повтарящи се групи – атрибути, които имат различни стойности за един и същи ключ, например за дадената по-долу релация студентът изтриването на една дисциплина води до изтрива целия запис:

Студенти<фак.№, име, бащино име, фамилия, код на специалност, име на специалност, факултет, код на дисциплина, име на дисциплина, оценка код на дисциплина, име на дисциплина, оценка >

- Първа нормална форма (ПНФ)
Една БД е в ПНФ, ако не съдържа повтарящи се групи. Повтарящата се група се отделя в нова релация (таблица)

Студенти<фак.№, име, бащино име, фамилия, код на специалност, име на специалност, факултет код на дисциплина, ПК, населено място, адрес >
Дисциплина< код на дисциплина, име на дисциплина, оценка >

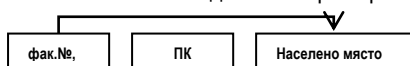
- Втора нормална форма (ВНФ)
Една БД се намира във ВНФ, ако е в първа нормална форма и всеки не атрибут е напълно функционално зависи от всички атрибути, а не само на част от тях)

Например данните за специалността: зависят само от код на специалност.

Студенти<фак.№, име, бащино име, фамилия, код на специалност, код на дисциплина, ПК, населено място, адрес >
Специалност<код на специалност, име на специалност, факултет >
Дисциплина< код на дисциплина, име на дисциплина, оценка >

- Трета нормална форма (ТНФ)

Една БД се намира в ТНФ, ако е в Втора нормална форма и липсват транзитивни зависимости между неключовите елементи данни. Например:



Студенти<фак.№, име, бащино име, фамилия, код на специалност, код на дисциплина, ПК, адрес >
Специалност<код на специалност, име на специалност, факултет >
Дисциплина< код на дисциплина, име на дисциплина, оценка >
Населени_места<ПК, населено място >

В релационната БД от съществено значение е определяне на връзките между различните таблици. Последните са дадени в таблица 1.

Таблица 1

1 : 1	факултетен номер : ЕГН ¹
1 : M	1 факултет : много специалности
M : 1	много специалности : 1 факултет
M : N	много студенти : много учебни дисциплини

Системите за Управление на Бази Данни (СУБД)

Основен инструмент за изграждане и манипулиране на БД са СУБД. Разпространените СУБД (MS Access, MySQL, Oracle, Paradox dBase, FoxPro, Cliper, Sybase) предоставят на потребителите различни възможности, които в преобладаващата си част са подобни. Основните функции могат да бъдат разпределени в групи и са дадени в Таблица 2. В доклада методиката на обучението на студентите е изградена на базата на MS Access.

Таблица 2

Създаване на БД	- Определяне структурата на таблицата; - Операции над таблица – изтриване, копиране; - Въвеждане на данни в таблиците; - Установяване на връзките между таблиците.
Актуализация на БД	- Изменение на структурата на таблиците; - Модификация (обновяване, промяна) на данните; - Изтриване на данните.
Обработка на	- Извличане на справки от една или повече таблици; - Отпечатване на данните; - Автоматизиране работата с БД (създаване на макроси и/или модули.

Описаната в доклада методика е базирана върху MS Access. Първоначално се извършва запознаване на студентите с общата структура на MS Access и нейните функционални възможности. Прави се кратка характеристика на отделните обекти, поддържани в MS Access: Table, Form, Query, Report, Macro, необходими на студенти в разработването на курсовия им проект. Продължителността на обучението позволява процесът условно да бъде разделен на две части (етапа). В първата всички студентите разработват една БД „Студент”, елементи от която са представени в доклада. Във втората част – самостоятелна работа на студента върху конкретен курсов проект.

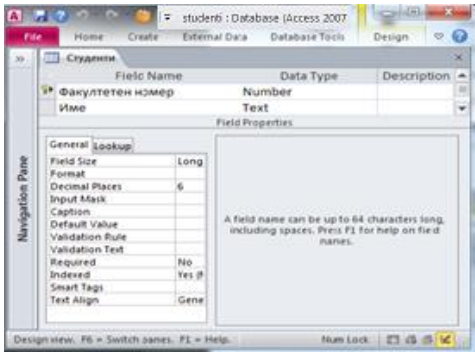
Обучението се извършва в следната последователност:

Проектиране на таблиците в БД.

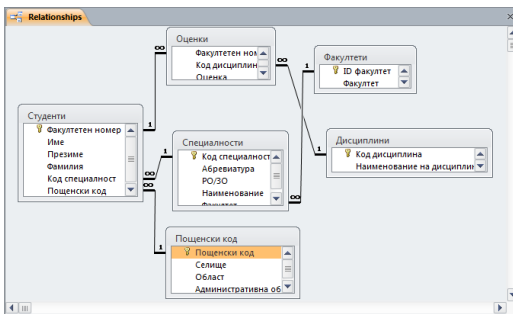
Всяка таблица се състои от два основни елемента: структура (Design View), фиг.3, и данни (Datasheet View). В режим Design View – се описват всички полета на таблицата, като се акцентира на основните им свойства: име, тип, дължина (формат при числова информация). Определят се първичния ключ и вторичните ключове (чрез тях се осъществява връзката на една таблица с друга).

Втората стъпка в проектирането на БД е изграждането на връзките между таблиците (Database Tools —> Relationships), представена на фиг. 4. Препоръчително е това да се извърши преди запълването на таблиците с данни.

¹ Конфиденциална информация, поради което не присъства в релацията за студентите



Фиг. 3 Design View за таблица „Студенти“



Фиг. 4 Връзките между таблиците

Фиг. 5 Форма за въвеждане на оценки на студентите

Field:	Факултетен номер	Име	Фамилия	Наименование на дисциплината	Оценка
Table:	Студенти	Студенти	Студенти	Дисциплини	Оценки
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	[<=веди Факултетен н				
or:					

```

SELECT Студенти.[Факултетен номер], Студенти.Име, Студенти.Фамилия, Avg(Оценки.Оценка) AS [среден успех]
FROM Студенти INNER JOIN Оценки ON Студенти.[Факултетен номер] = Оценки.[Факултетен номер]
GROUP BY Студенти.[Факултетен номер], Студенти.Име, Студенти.Фамилия
HAVING ((Студенти.[Факултетен номер])=[<=веди факултетет номер]);

```

Фиг. 6 Извличане на оценка по факултетен №

Въвеждане и редактиране на данните

Съществуват две възможности за въвеждане и редактиране на данните:

- В режим Datasheet View;
- Посредством форми (Form), фиг. 5.

Обектите Form се използват за визуализиране на данните от БД върху екрана на монитора.

Редактиращите операции се обособяват условно в две групи: операции над един запис (търсене, добавяне, изтриване) и операции над едно поле (търсене, модификация).

Извличане на информация от БД

Основно предназначение на всяка БД е извличане и обработка на данни от нея. В MS Access са вградени два езика за обработка на данните: SQL (Structured Query Language - структурен език на заявките) и QBE (Query-By-View MS Access е предвиден специален обект Query (запитване, заявка, справка).

Основният акцент в обучението е формирането на критерии за извличане на данни:

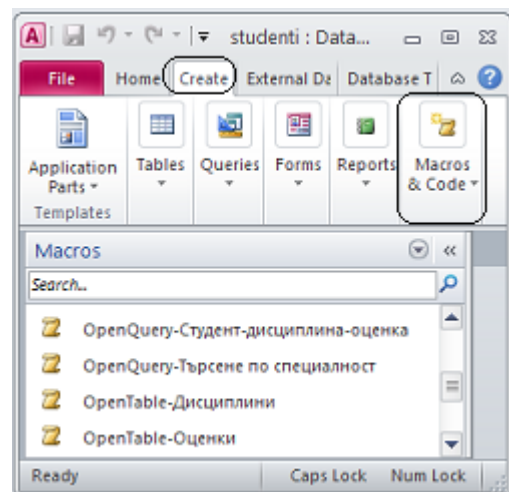
- Типове запитвания;
- Работа с Expression Builder;
- Вградени функции в MS Access.

На фиг.6 е представена информация за извличане на оценката по факултетен номер.

Автоматизиране функционалността на БД

За да постигнем автоматизиране на нашите задачи и да добавяме функционалност на формуляри, отчети е необходимо да използваме макроси.

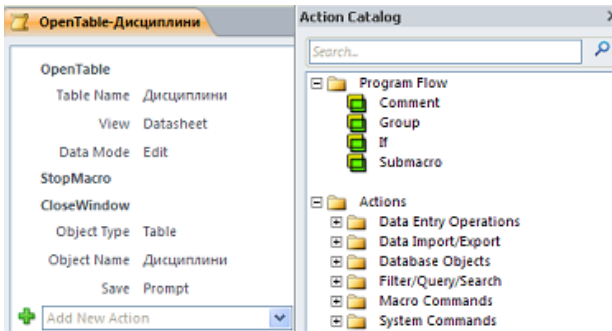
В Access е позволено макросите да се възприемат като „опростен език за програмиране“, т.е. създавайки макрос да указваме какво ще бъде действието (OnClick) на мишката посочвайки даден бутон. На фиг. 7 и на фиг. 8 са показани съответно от къде се създава макрос и стъпки при създаването на му.



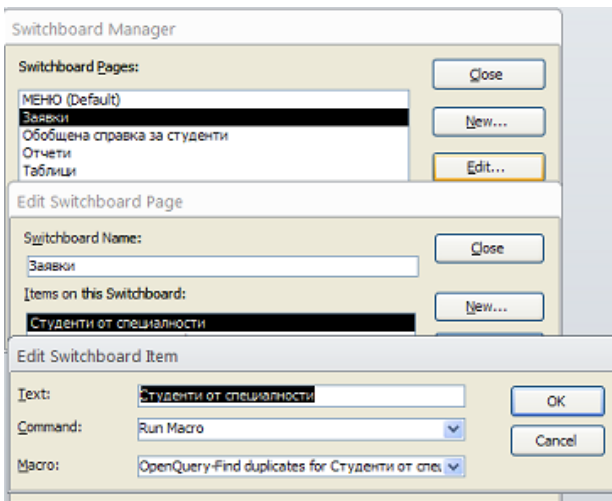
Фиг. 7 Създаване на макрос

За доближаването на проекта до информационна система се използва Switchboard. Това е форма, чрез която можем лесно да „сърфираме“ из нашата база данни. Необходимо е да програмираме, т.е. настоим бутоните му така че да водят до необходимата информацията само с едно кликане с мишката. На фиг. 9 е представено

проектирането на навигационното табло.

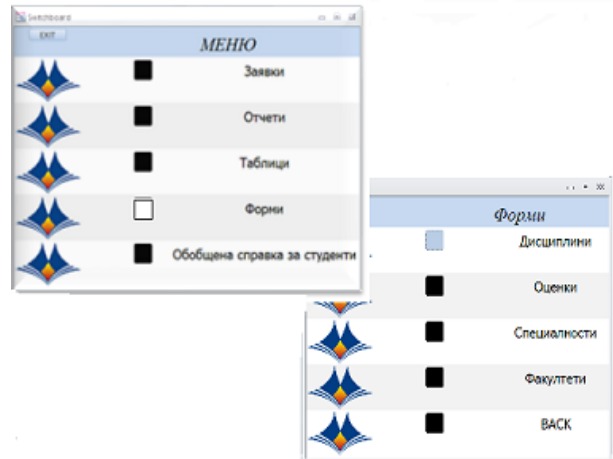


Фиг. 8 Създаване на макрос за отваряне на таблица



Фиг. 9 Проектиране на Switchboard.

В представената база данни макроси са използвани за отваряне на заявки и отчети за прикачване към входното бутонно табло (Switchboard).



Фиг. 10 Визуализация на входно табло. Меню.

Заклучение

В заключение може да се каже, че след завършване на обучителния курс и успешна защита на самостоятелните курсови проекти, студентите са усвоили:

- основни знания по анализиране, структуриране, и представяне на информацията в единна база данни;
- извличане на данни по конкретно зададени критерии;
- работа със специализиран софтуер за СУБД;

Литература

Cood E.F., Relational Completeness of Data Base Sublanguages in data base system, B. 6. Prentice-Hall, 1972" .

Мартин Дж., Организация баз данных в вычислительных системах, Москва, Мир, 1982.

Вискас Дж. Всичко за Microsoft Access 2000, София, Софтпрес, 2000г.

Статията е препоръчана за публикуване от кат. „Информатика“.