Національний університет “ Києво-Могилянська Академія “

# Курсова робота

на тему:

**“ Розробка складних інформаційно-пошукових систем “**

### Науковий керівник: Студентки 3-го курсу

Кулябко П.П. департаменту комп’ютерних технологій

## Фіялка Світлани

## Київ-1998

Зміст:

## Постановка задачі ………………………………………2 ст.

Опис предметної області ……………………………….3 ст.

Інформаційна модель концептуального рівня ………..4 ст.

Реалізація ………………………………………………..8 ст.

Засоби інтефейсу ……………………………………….21 ст.

*Постановка задачі* .

В наш час системи баз даних користуються все більшим попитом. Так як обсяг самих баз даних може бути дуже великим, доцільно використовувати систему керування базами даних. Майже усі сучасні системи побудовані на основі реляційної моделі керування базами даних. Назва “*реляційна*” пов”язана з тим, що кожний запис в такій базі даних містить інформацію, яка відноситься (related) тільки до одного конкретного об”єкту. Крім того, з даними двох типів можна працювати як з єдиним цілим, заснованим на значеннях пов”язаних (related) між собою даних. Система керування базами даних надає можливість контролювати задання структури і опис даних, роботу з ними і організацію колективного користування інформацією, вона збільшує можливості і полегшує каталогізацію та ведення великих обсягів інформації в числених таблицях.

Для розробки складних прикладних програм потрібна потужна СКБД та система розробки прикладних програм, яка допомагає автоматизувати основні задачі прикладної програми.

В даній роботі була використана система СКБД – Access. *Мета роботи*: розробити складну інформаційно-пошукову систему з обраної предметної області, яка повинна забезпечувати підтримку і організацію даних в ній, проведення необхідних змін та вдосконалень, вибір необхідної інформації. Система має бути орієнтована на звичайного користувача, який без зайвих труднощів зможе отримати, ввести, редагувати дані. Для зручного представлення даних та їх використання мають бути розроблені відповідні засоби інтерфейсу.

*Опис предметної області.*

В якості предметної області для розробленого проекту реляційної бази даних було вибрано *бібліотечну справу* в певному одному вищому навчальному закладі. В даній бібліотеці існує три *відділи*: читальна зала для студентів, які вивчають гуманітарні науки, читальна зала для студентів-природничників та читальна зала для студентів факультету суспільних наук. В кожному відділі працює один *бібліотекар*, який і займається обслуговуванням студентів, які прийшли в цей відділ. Так як всі *читачі* одночасно є і студентами, то окрім відомостей про їх ім”я та прізвище, доцільно знати назву факультету, де вони навчаються та курс. Всі читачі є студентами лише даного вищого навчального закладу. Основний фонд бібліотеки – *книги.* Для кожної книги працівниками бібліотеки робиться запис про її ідентифікаційний код, який є унікальним, не може бути застосований до іншої книги або змінений. Кожна книга може знаходитися лише в одному відділі. Розподілення за відділами відбувається на основі тематики книги. Для кожної книги надаються дані про рік її видання, місце та назву видавництва, де вона була видана. В бібліотеці є як одна, так і декілька книг видані одним і тим сами видавництвом. Основна операція в бібліотеці – це обслуговування читачів. Бібліотекар для кожного читача веде абонементний запис, де фіксуються назви книг, які він взяв та дата їх взяття.

Один читач може читати декілька книг одразу. Так як в бібліотеці може бути декілька однакових книг, то і декілька читачів можуть читати одну книгу. Запис про дату взяття книги необхідний для контролю читачів. Якщо студент порівняно довго тримає книгу, то дані про нього заносяться в реєстр читачів-боржників. Обслуговування цього студента в бібліотеці припиняється. Студент має право знову взяти книжки лише в тому випадку, коли він поверне в бібліотеку взяту ним книгу та відбуде “срок покарання”, тобто ще після здачі книги студент на буде обслуговуватися певний час. Всі операції по переведенню читача в стан боржників і навпаки виконує бібліотекар.

*Інформаційна модель концептуального рівня.*

Для будь-якої інформаційної моделі можна виділити три рівні організації даних: внутрішній рівень, зовнішній рівень та концептуальний рівень. *Внутрішній рівень* інформаційної моделі – це рівень структур даних певної СКБД: одна і та сама предметна область може бути реалізована через Access, Paradox, FoxPro або іншу систему. *Зовнішній рівень* – це рівень (точка зору) зовнішніх користувачів щодо організації даних. Між внутрішнім і зовнішнім рівнями знаходиться *концептуальний рівень*, який інтегрує різні точки зору зовнішніх користувачів, розв”язуючи суперечності між ними. Відносно внутрішнього рівня концептуальний рівень має єдину структурну мову. Всі СКБД, на яких реалізовано певну предметну область, повинні підтримувати концептуальний рівень.

Для опису інформаційної моделі концептуального рівня може бути вибрана *ER-модель*. Основними поняттями ER-моделі є: *об”єкти*, *зв”язки*, *атрибути* та *ключі. Об”єкт* – певна сукупність даних, які утворюють семантичне або логічне ціле. В реляційних базах даних відповідником об”екту є реляція – таблиця. Для даної предметної області можна виділити такі об”єкти: *читач, книга, абонемент, відділ, бібліотекар, видавництво, факультет, боржники, відділ-книга, видавництво-книга, бібліотекар-відділ.* Кожен з цих об”єктів складається з відповідних *атрибутів* – полів запису, що містять певну частину інформації про об”єкт. Для об”єкту “читач“ атрибутами виступають такі поля: код читача, прізвище, ім”я, код факультету, курс. Для об”єкту “відділ” атрибутами є: код відділу та його назва. Об”єкт “ бібліотекар” містить атрибути: код бібліотекаря, прізвище, освіта, вік, стаж роботи. “Видавництво” -: код видавництва, назва, індекс міста, назва міста, адреса. “Факультет” -: код факультету, назва, прізвище керівника, який очолює факультет. “Боржники”-: код читача, код книги, дата взяття книги, факт про повернення книги, дата повернення. “Книга” -: код книги, назва, автор. “Абонемент” -: код читача, код книги, дата взяття книги, факт про повернення книги. “Відділ-книга” -: код книги, код відділу, кількість однакових книжок у відділі. Видавництво-книга”-: код книги, код видавництва, рік видання книги та її тираж. “Бібліотекар-відділ”-: код відділу та код бібліотекаря.

*Ключ* – це атрибут або група атрибутів, який однозначно ідентифікує примірник об”єкту за умови мінімальності. Під умовою мінімальності тут слід розуміти той факт, що зі складу ключа не можна вилучити жодного атрибуту без втрати властивості ідентифікації. Для об”єкту “книга” ключем виступає атрибут “код книги”. За кодом книги можна однозначно визначити назву книги та її автора. Для об”єкту “читач” ключовим атрибутом є “код читача”, за яким можна знайти прізвище та ім”я читача, а також факультет та курс, на якому він навчається. Об”єкт “факультет” має ключ – “код факультету”. Кожен факультет має свій код, за яким знаходиться його назва та прізвище керівника, що його очолює. У об”єкта “боржники” ключем виступають два атрибути – код читача та код книги. Читачі-боржники є підмножиною більшої множини читачів, для яких також ведеться абонементний запис, де теж фіксується код читача та код книги, яку він тримає. Читачів-боржників доцільно виокремити з усього загалу читачів, не дивлячись на те, що у відповідних таблицях для читачів –“абонемент” та для читачів-боржників – “боржники” є схожі атрибути. По-перше це необхідно для того, щоб не перевантажувати базу даних. Крім того, для звичайних читачів не фіксується дата повернення книги, оскільки тут вона є неважливою. Навпаки, для читачів-боржників дата повернення книги обов”язково фіксується і запис про неї не видаляється ще певний час. Як було зазначено раніше, це пов”язано з тим, що читач-боржник переходить у спільний реєстр читачів тільки через певний час після здачі книги. Цей факт пов”язаний і з наступними міркуваннями: якщо для звичайного читача встановлення атрибуту “факт про здачу книги” в істину автоматично видаляє записи про нього з таблиці “абонемент” до наступного обслуговування, то для читача-боржника він означає відлік часу до повного відновлення своїх читацьких прав. Атрибут “дата взяття книги” є більш важливим для об”єкту “читач”, оскільки в бібліотеці постійно вираховується різниця між поточною датою та датою взяття книги. Якщо її різниця перевищує певні межі, читач автоматично зараховується в стан читачів-боржників.Встановлює всі ці обмеження сам бібліотекар, він може змінити їх як завгодно.

Для об”єкту “відділ” ключем є атрибут “код відділу”, за яким однозначно встановлюється його назва. За ключовим атрибутом “код бібліотекаря” для об”єкту “бібліотекар” однозначно визначаються прізвище бібліотекаря, його освіта, вік та стаж роботи. Ключ “код видавництва” для об”єкту “видавництво” визначає назву видавництва, індекс міста, в якому розташовано видавництво, назву цього міста та адресу, за якою можна знайти дане видавництво в цьому місті. Об”єкт “бібліотекар-відділ” фактично можна інтерпретувати як: “бібліотекар працює у відділі”. Він містить лише два атрибути, які є ключовими. За кодом бібліотекаря можна знайти відділ, де він працює, і навпаки. Об”єкт “відділ-книга” інтерпретується як “книга знаходиться у відділі”. Цей об”єкт має два ключових атрибути: код книги та код відділу. Оскільки предметною областю передбачено, що одна книга може знаходитися лише у одному відділі, то між цими атрибутами існує взаємна однозначність. І, нарешті, об”єкт “видавництво-книга” можна також інтерпретувати як “видавництво видає книгу”. Ключовими атрибутами є код книги та код видавництва.

*Зв”язок* – це відношення між примірниками об”єктів одного і того ж типів. Зв”язок характеризується арністю, яка визначається кількістю типів об”єктів, що включаються в зв”язок та типом відображення. Існують відображення: один-до-одного, один-до-багатьох, багато-до-багатьох. *Зв”язок “один-до-одного”* – один примірник одного типу об”єкта співвідноситься з одним примірником іншого типу об”єкта, або до цього ж самого. *Зв”язок “один-до-багатьох”* – одному примірнику деякого типу об”єкту відповідає декілька примірників іншого типу об”єкта. Зв”язок “багато-до-багатьох” – кільком примірникам одного типу об”єкта відповідає кілька примірників іншого.

В даній предметній області існують такі зв”язки між об”єктами:

1. між “читачем” та “книгою”: зв”язок – багато-до-багатьох. Багато читачів можуть читати одну книгу, багато книг може читати один читач.
2. між “боржником” та “книгою” – аналогічна ситуація.
3. між “видавництвом” та “книгою”: зв”язок – один-до-багатьох. В одному видавництві може видаватися багато книг. За даною предметною областю одна і та сама книга не може видаватися у кількох видавництвах.
4. між “відділом” та “книгою”: зв”язок – один-до-багатьох. В одному відділі може знаходитися багато книг.Одна і та сама книга не може знаходитися у кількох відділах.
5. між “бібліотекарем” та “відділом”: зв”язок – один-до-одного. В одному відділі працює лише один бібліотекар.
6. між “читачем” та “факультетом”: зв”зок – один-до-багатьох. На одному факультеті може навчатися багато читачів, але один читач не може навчатися на кількох факультетах.

Остаточно ER-модель для даної предметної області можна представити у вигляді *схеми*.

читає знаходиться

**читач** m n **книга** n 1 **відділ**

n “абонемент” “книга-відділ” 1

начється тримає n n працює

видається

m “видавництво-книга” “бібліотекар-відділ”

**боржник**  1 1 1

**факультет**  **видавництво** **бібліотекар**

*Реалізація.*

Засобом для реалізації даної прикладної системи, як вже зазначалось раніше, стала система Microsoft Access’97.

Першим етапом по створенню реляційної бази дани була розробка необхідних реляцій (таблиць) та відповідне їх наповнення. В даній базі даних «Бібліотека» створено 11 таблиць: абонемент, книга, відділ-книга, читач, боржники, факультет, відділ, бібліотекар, бібліотекар-відділ, видавництво, видавництво-книга.

Таблиця “абонемент” має 4 поля, два з яких ключові – “код читача” та “код книги” і два інші – “дата взяття книги” з даними типу дата та “факт про повернення книги” з даними логічного типу.

|  |  |  |  |
| --- | --- | --- | --- |
| **Код читача** | **Код книги** | **Дата взяття книги** | **Факт сдачі** |
| 1 | 47 | 12.05.98 | Нет |
| 1 | 49 | 12.05.98 | Нет |
| 2 | 43 | 13.05.98 | Нет |
| 2 | 44 | 13.05.98 | Нет |
| 2 | 46 | 13.05.98 | Нет |
| 3 | 41 | 13.05.98 | Нет |
| 4 | 40 | 10.05.98 | Да |
| 6 | 31 | 05.05.98 | Нет |
| 6 | 33 | 05.05.98 | Нет |
| 7 | 40 | 10.05.98 | Да |
| 8 | 34 | 11.05.98 | Нет |
| 8 | 37 | 11.05.98 | Нет |
| 11 | 8 | 12.05.98 | Нет |
| 11 | 9 | 13.05.98 | Нет |
| 12 | 6 | 14.05.98 | Нет |
| 12 | 7 | 14.05.98 | Нет |
| 13 | 6 | 13.05.98 | Да |
| 15 | 19 | 12.05.98 | Нет |
| 15 | 20 | 12.05.98 | Нет |
| 15 | 22 | 12.05.98 | Нет |
| 16 | 22 | 10.05.98 | Нет |
| 17 | 24 | 10.05.98 | Нет |
| 17 | 25 | 10.05.98 | Нет |
| 17 | 27 | 10.05.98 | Нет |
| 17 | 28 | 12.05.98 | Нет |
| 17 | 29 | 12.05.98 | Нет |
| 18 | 6 | 10.05.98 | Нет |
| 18 | 13 | 10.05.98 | Нет |
| 20 | 16 | 11.05.98 | Нет |
| 20 | 31 | 11.05.98 | Нет |
| 20 | 33 | 12.05.98 | Нет |

Таблиця “книга” має три поля, ключове поле – “код книги” з числовими даними. Інші поля: “назва” з текстовими даними та “автор” – теж з текстовими даними.

|  |  |  |
| --- | --- | --- |
| **Код книги** | **Назва** | **Автор** |
| 2 | Макроэкономика | Мэнкью |
| 3 | Экономикс | Самюэльсон |
| 4 | Линейное программирование | Юдин |
| 5 | Вища математика | Дубовик |
| 6 | Першотвір і переклад | Коптілов |
| 7 | О всеобщем родстве языков мира | Мельничук |
| 8 | Экономикс | Кэмпбелл |
| 9 | Дослідження операцій | Варнер |
| 10 | Системний аналіз | Пономаренко |
| 11 | Теорія і практика перекладу | Коптілов |
| 12 | Реалія і переклад | Зорівчак |
| 13 | Слово живое и мертвое | Галь |
| 14 | Индоевропейский язык и индоевропейцы | Гамкрелидзе |
| 15 | Сучасне порівняльно-історичне мовознавство | Білецький |
| 16 | Праславянский язык | Хенрик |
| 17 | Избранные работы по языкознанию | Покровский |
| 18 | Сравнительная грамматика индоевропейских языков | Савченко |
| 19 | Лингвистический энциклопедический словарь | Ярцева |
| 20 | Політологія | Дзюбко |
| 21 | Основи загальної теорії права та держави | Рабінович |
| 22 | Основи держави і права | Котюк |
| 23 | The Social Experience | Vander Zanden |
| 24 | Українська мова | Мацько |
| 25 | Интенсивный курс турецкого языка | Щека |
| 26 | The Heinemann English Grammar | Beaumont |
| 27 | Введение в славянскую филологию | Супрун |
| 28 | Язык в пространстве и времени | Мартынов |
| 29 | Українська гіпотеза | Карпенко |
| 30 | Новый Навигатор Internet | Гилстер |
| 31 | Техніка програмування мовою Сі | Проценко |
| 32 | Дослідження операцій | Зайченко |
| 33 | Лінійні простори | Калужнін |
| 34 | IBM PC для пользователя | Фигурнов |
| 35 | Лекции по линейной алгебре | Гельфанд |
| 36 | Программирование на языке Turbo Pascal | Зубов |
| 37 | Учимся программировать на С++ | Джамса |
| 38 | Основы линейной алгебры | Мальцев |
| 39 | Сборник задач по линейной алгебре | Проскуряков |
| 40 | Комп"ютерні віруси в MS DOS | Касперський |
| 41 | Программирование в Paradox под Windows на примерах | Тиней |
| 42 | Пишем вирус и антивирус | Хижняк |
| 43 | Компьютерная вирусология | Безруков |
| 44 | Access 7.0 | Вейскас |
| 45 | Язык Си для профессионалов | Шилдт |
| 46 | Программирование видеоадаптеров | Фролов |
| 47 | Язык ассемблера для IBM PC | Абель |
| 48 | Турбо Паскаль | Поляков |
| 49 | Paradox-это очень просто | Сигель |
| 50 | Учебник японского языка | Головнин |
| 111 | Основи | Дзюбикyg8 |

Таблиця “читач” має 5 полів: ключове поле – “код читача” з числовими даними. Інші поля: “прізвище” – текстові дані, “ім”я” – текстові дані, “номер факультету” – числові дані, “курс” числові дані.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Код читача** | **Прізвище** | **Ім"я** | **Факультет** | **Курс** |
| 1 | Сігаєва | Марина | 1 | 3 |
| 2 | Фіялка | Світлана | 1 | 3 |
| 3 | Пархоменко | Олена | 1 | 3 |
| 4 | Комарницька | Людмила | 1 | 3 |
| 5 | Підгорний | Свєтослав | 1 | 3 |
| 6 | Федосова | Наталя | 1 | 3 |
| 7 | Тарнавська | Олена | 1 | 3 |
| 8 | Бубенщиков | Євген | 1 | 3 |
| 9 | Сиворакша | Сергій | 1 | 3 |
| 10 | Овдій | Юрій | 1 | 3 |
| 11 | Цап | Ольга | 2 | 3 |
| 12 | Дундій | Ганна | 4 | 3 |
| 13 | Медвєдєва | Соломія | 5 | 3 |
| 14 | Жилєвський | Олександр | 2 | 2 |
| 15 | Жарко | Тетяна | 3 | 2 |
| 16 | Іванова | Тетяна | 6 | 1 |
| 17 | Галабурда | Юля | 4 | 3 |
| 18 | Сосновська | Люба | 4 | 3 |
| 19 | Вахненко | Тетяна | 4 | 3 |
| 20 | Беднова | Ганна | 4 | 3 |

Таблиця “боржники” має 5 полів: ключові поля – “код читача” та “код книги”. Інші поля: “дата взяття книги” – дані формату дат, “факт про повернення книги” – логічні дані, “дата повернення книги” – дані формату дат.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Код читача** | **Код книги** | **Дата взяття** | **Повернення книги** | **Дата повернення** |
| 5 | 30 | 30.03.98 | Да | 01.04.98 |
| 9 | 39 | 30.04.98 | Нет |  |
| 10 | 10 | 30.03.98 | Да | 01.04.98 |
| 14 | 3 | 01.05.98 | Нет |  |
| 14 | 4 | 01.05.98 | Нет |  |
| 14 | 111 | 30.04.98 | Нет |  |
| 19 | 14 | 30.04.98 | Нет |  |
| 19 | 15 | 01.05.98 | Нет |  |

Таблиця “факультет” містить 3 поля: ключове поле – “номер факультету”.Поле “повна назва” – назва факультету, текстові дані. Поле “керівник підрозділу” – текстові дані.

|  |  |  |
| --- | --- | --- |
| **Номер факультету** | **Повна назва** | **Керівник підрозділу** |
| 1 | Департамент комп"ютерних технологій | Глибовець М.М. |
| 2 | Департамент економічної теорії | Ястремський О. |
| 3 | Факультет правничих наук | Сущенко В.М. |
| 4 | Факультет гуманітарних і суспільних наук | Гусєв В.І. |
| 5 | Факультет природничих наук | Безусько А.Г. |
| 6 | Школа соціальної роботи | Полтавець В.І. |

Таблиця “відділ” має 2 поля: ключове поле – “код відділу” з числовими даними та поле “назва” з текстовими даними.

|  |  |
| --- | --- |
| **Код відділу** | **Назва** |
| 1 | читальна зала для суспільнознавців |
| 2 | читальна зала для природничників |
| 3 | читальна зала для гуманітаріїв |

Таблиця “бібліотекар” має 5 полів: ключове поле – “код бібліотекаря” з числовими даними, поле “прізвище” з тектовими даними, поле “освіта” з текстовими даними, поле “вік“ з числовими даними, поле “стаж” з числовими даними.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Код бібліотекаря** | **Прізвище** | **Освіта** | **Вік** | **Стаж** |
| 1 | Ярошниченко Т.Я. | вища | 45 | 25 |
| 2 | Мирошниченко Я.Т. | вища | 40 | 15 |
| 3 | Фощенко Т.Т. | вища | 35 | 10 |

Таблиця “видавництво” має 5 полів: ключове поле – “код видавництва” з числовими даними, поле “назва” з текстовими даними, поле “індекс міста” з числовими даними, поле “місто” з текстовими даними, поле “адреса” з текстовими даними.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Код видавництва** | **Назва видавництва** | **Індекс міста** | **місто** | **Адреса** |
| 1000 | Просвіта | 252001 | Київ | вул.Шевченко,23 |
| 1001 | Світанок | 252123 | Київ | бул.Л.Українки,14 |
| 1002 | Либідь | 252001 | Київ | Хрещатик,10 |
| 1003 | Вища школа | 252054 | Київ | Гоголівська вул.,7 |
| 1004 | Дніпро | 252332 | Київ | вул.Милославська,10 |
| 1005 | Книжна фабрика | 252160 | Київ | вул.Перова,12 |
| 1006 | ЛДУ | 239170 | Львів | бул.Перемоги,27 |
| 1007 | Высшая школа | 101430 | Москва | Неглинная ул.,29/14 |
| 1008 | ТДУ | 108677 | Тбілісі | ул.Интернациональная,14 |
| 1009 | Наука | 128367 | Москва | 2-й Рижский пер.,3 |
| 1010 | Вентурі | 252003 | Київ | вул.Сосновська,6 |
| 1011 | Random House |  | New York |  |
| 1012 | Издательство Московского ууниверситета | 103009 | Москва | ул.Большая Никитинская,5/7 |
| 1013 | Heinemann Oublishers |  | Oxford |  |
| 1014 | Діалектика | 252022 | Київ | просп.Глушкова,6 |
| 1015 | Инфра.М | 127247 | Москва | Дмитровское шоссе,107 |
| 1016 | Филин | 128820 | Москва | Новопетровская ул.,1 |
| 1017 | Мир | 129820 | Москва | 1-й Рижский пер.,2 |
| 1018 | Бином | 103473 | Москва | ул.Новослободская,50 |
| 1019 | Питер | 197198 | Санкт-Петербург | ул.Благодатная,67 |
| 1020 | И.В.К.-Софт | 127147 | Москва | ул.Строительная,25 |
| 1021 | Диалог-Мифи | 126548 | Москва | ул.Закревского,9 |
| 1022 | БорАГ | 102111 | Москва | ул.Набережная,6 |
| 1023 | Лингва | 102110 | Москва | ул.Красноармейская,8 |

Таблиця “бібліотекар-відділ” має 2 ключових поля: “код відділу” та “код бібліотекаря”.

|  |  |
| --- | --- |
| **Код відділу** | **Код бібліотекаря** |
| 1 | 3 |
| 2 | 2 |
| 3 | 1 |

Таблиця “відділ-книга” має 3 поля: ключові поля – “код книги” та “код відділу”, інше поле – “кількість” містить числові дані.

|  |  |  |
| --- | --- | --- |
| **Код книги** | **Код відділу** | **Кількість** |
| 2 | 1 | 10 |
| 3 | 1 | 10 |
| 4 | 2 | 10 |
| 5 | 2 | 20 |
| 6 | 3 | 5 |
| 7 | 3 | 5 |
| 8 | 1 | 25 |
| 9 | 2 | 5 |
| 10 | 2 | 4 |
| 11 | 3 | 8 |
| 12 | 3 | 6 |
| 13 | 3 | 4 |
| 14 | 3 | 5 |
| 15 | 3 | 8 |
| 16 | 3 | 5 |
| 17 | 3 | 5 |
| 18 | 3 | 10 |
| 19 | 3 | 20 |
| 20 | 1 | 25 |
| 21 | 1 | 25 |
| 22 | 1 | 25 |
| 23 | 1 | 30 |
| 24 | 3 | 30 |
| 25 | 3 | 5 |
| 26 | 3 | 5 |
| 27 | 3 | 6 |
| 28 | 3 | 6 |
| 29 | 3 | 5 |
| 30 | 2 | 15 |
| 31 | 2 | 15 |
| 32 | 2 | 5 |
| 33 | 2 | 30 |
| 34 | 2 | 10 |
| 35 | 2 | 25 |
| 36 | 2 | 25 |
| 37 | 2 | 10 |
| 38 | 2 | 15 |
| 39 | 2 | 10 |
| 40 | 2 | 10 |
| 41 | 2 | 10 |
| 42 | 2 | 10 |
| 43 | 2 | 10 |
| 44 | 2 | 10 |
| 45 | 2 | 10 |
| 46 | 2 | 15 |
| 47 | 2 | 15 |
| 48 | 2 | 10 |
| 49 | 2 | 5 |
| 50 | 3 | 5 |
| 111 | 1 | 5 |

Таблиця “видавництво-книга” має 4 поля: з них два ключових – “код книги” та “код видавництва”, та інші два поля – “рік видання” – дані формату дат, “тираж” – числові дані.

Наступним важливим кроком в розробці бази даних було встановлення зв”язків між таблицями з врахуванням вимог третьої нормальної форми.

*Перша нормальна форма*: всі атрибути реляції мають бути атомарними, тобто вони повинні бути неподільними, не мати власної структури, не бути самі по собі реляціями. Кожне поле даних повинно містити унікальні елементи даних і жоден окремий елемент даних не повинен повторюватися в таблиці.

Розроблена база даних задовільняє вимогам першої номальної форми.

*Друга нормальна форма*: Вимоги для першої нормальної форми + : Кожна таблиця повинна мати унікальний ідентифікатор (первинний ключ). Всі

неключові поля таблиці мають знаходитися в функціонально повній залежності від цього ключа, тобто повністю ним визначатися.

Дана база даних задовільняє умовам другої нормальної форми. Кожна реляція в ній має ключове поле, яке однозначно визначає всі інші атрибути.

*Третя нормальна форма*: Вимоги для другої нормальної форми +: Всі поля, що не входять в первинний ключ, повинні бути взаємнонезалежними, Тобто повинна існувати можливість змінювати значення одного неключового поля, не змінюючи при цьому значення будь-якого іншого поля бази даних. Не повинно існувати транзитивної залежності вторинних атрибутів (тіх, що не входять до складу жодного квазі-ключа) від від кожного квазі-ключа.

A1 A2 A3

Дана база даних задовільняє вимогам третьої нормальної форми.

Останній етап проектування бази даних – це створення зв”язків між таблицями. Оскільки Access не дозволяє визначати прямий зв”язок “багато-до-багатьох”, то треба створювати додаткову таблицю перетину, за допомогою якої один зв”язок “багато-до-багатьох” зводиться до двох зв”язків типу “один-до-багатьох”. Саме з таких міркувань була створена таблиця “абонемент” , вона розбиває прямий зв”язок типу “багато-до-багатьох” між “читачами” та “книгами” на два зв”язки типу “один-до-багатьох”.

Схема зв”язків між таблицями в базі даних “Бібліотека”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Книга | | | | |
| *Код книги* | | | | |
| Назва | | | | |
| Автор | | | | |
| Абонемент | |
| Код читача | |
| Код книги | |
| Дата взяття книги | |
| Факт про повернення | |
| Читач | | | | |
| Код читача | | | | |
| Прізвище | | | | |
| Ім”я | | | | |
| Номер факультету | | | | |
| Курс | | | | |
| Боржники | |
| Код читача | |
| Код книги | |
| Дата взяття | |
| Факт про повернення | |
| Дата повернення | |
| Бібліотекар | | | | | |
| Код бібліотекаря | | | | | |
| Прізвище | | | | | |
| Освіта | | | | | |
| Вік | | | | | |
| Стаж | | | | | |
| Факультет | | |
| Номер факультету | | |
| Повна назва | | |
| Керівник підрозділу | | |
| Відділ | | |
| Код відділу | | |
| Назва | | |
| Бібліотекар-відділ | | | |
| Код відділу | | | |
| Код бібліотекаря | | | |
| Відділ-книга | |
| Код книги | |
| Код бібліотекаря | |
| Кількість | |
| Видавництво | | | | | | |
| Код видавництва | | | | | | |
| Назва | | | | | | |
| Індекс міста | | | | | | |
| Місто | | | | | | |
| Адреса | | | | | | |
| Видавництво-книга |
| Код книги |
| Код видавництва |
| Рік видання |
| Тираж |

Вся робота з базою даних відбувається через запити. За їх допомогою можна взнати будь-яку інформацію про читача, книгу, видавництво, відділ, бібліотекаря та ін. За допомогою запитів можна змінювати, видаляти будь-яку інформацію в базі даних. Через запити проходить і процес роботи з читачами. Створені запити при їх запуску згідно з усіма раніше описаними принципами заносять, наприклад, студентів у список читачів-боржників, або видаляють їх звідти. Деякі запити були написані з використанням графічних можливостей QBE (запита за зразком), інші – безпосередньо мовою SQL, стандартною мовою якою користується Microsoft Access для роботи з даними. Для написання складних запитів з множинними порівняннями використовувався принцип підпорядкованих запитів, результати роботи яких використовувалися в якості умов відбору в інших запитах. Майже всі запити реалізовані через параметри, тобто при їх запуску система просить ввести значення якогось параметра, на основі якого потім буде відбуватися пошук необхідної інформації.

Найтиповіші запити по базі даних “Бібліотека”:

1. Знайти факультет, на якому навчається студент. SQL-запит:

SELECT DISTINCTROW факультет.[Повна назва], факультет.[Керівник підрозділу], читач.[Прізвище]&” “&[Ім”я] AS Выражение1

FROM факультет INNER JOIN [читач] ON факультет.[Номер факультету]=читач.Факультет

WHERE ((([читач].[Прізвище]&” “&[Ім”я])=[Введіть прізвище та ім”я

читача]);

2) Знайти прізвище читача, який читає таку-то книгу.

SELECT DISTINCTROW читач.Прізвище, читач.[Ім”я], книга.Назва

FROM [читач] INNER JOIN (книга INNER JOIN абонемент ON книга.[Код книги]=абонемент.[Код книги]ON читач.[КОд читача]=абонемент.[Код читача]

WHERE (((книга.Назва)=[Введіть назву книги]));

3) Який факультет очолює такий-то декан ?

SELECT DISTINCTROW факультет.[Повна назва], факультет.[Керівник підрозділу]

FROM факультет

WHERE (((факультет.[Керівник підрозділу])=[Введіть прізвище декану]));

1. Знайти всіх студентів, які навчаються на такому-то курсі.

SELECT DISTINCTROW читач.[Код читача], читач.Прізвище, читач.[Ім”я], читач.Курс

FROM [читач]

WHERE (((читач.Курс)=[Введіть курс]));

1. Знайти назви книг та місця, де вони були видані, якщо роки їх видання з

такого-то по такий-то рік.

SELECT DISTINCTROW [видавництво-книга].[Рік видання], книга.Назва, книга.Автор, видавництво.[Назва видавництва], видавництво.місто

FROM книга INNER JOIN (видавництво INNER JOIN [видавництво-книга] ON видавництво. [Код видавництва]=[видавництво-книга].[Код видавництва] ON книга.[Код книги]=[видавництво-книга].[Код книги]

WHERE ((([видавництво-книга].[Рік видання]) BETWEEN [Введіть перший рік випуску] AND [Введіть останній рік випуску]));

1. За номером відділу знайти його повну назву та бібліотекаря, який там

працює.

SELECT DISTINCTROW бібліотекар.Прізвище, відділ.[Код відділу], відділ.Назва

FROM відділ INNER JOIN (бібліотекар INNER JOIN [бібліотекар-відділ] ON бібліотекар.[Код бібліотекаря]=[бібліотекар-відділ].[Код бібліотекаря]) ON відділ.[Код відділу]=[бібліотекар-відділ].[Код відділу]

WHERE ((( відділ.[Код відділу])=[Введіть номер відділу]));

1. Знайти всі книжки, які не знаходяться у такому-то відділі.

SELECT DISTINCTROW книга.Назва, книга.Автор

FROM книга INNER JOIN (відділ INNER JOIN [відділ-книга] ON відділ.[Код відділу]=[відділ-книга].[Код відділу]) ON книга.[Код книги]=[відділ-книга].[Код книги]

WHERE (((відділ.Назва)<>[Введіть назву відділу]))

GROUP BY книга.Назва, книгаюАвтор;

1. Знайти коди всіх тих читачів, які прнаймі читали книгу з кодом 6.

# SELECT DISTINCTROW читач.[Код читача]

FROM абонемент INNER JOIN [читач] ON читач.[Код читача]=абонемент.[Код читача]

WHERE EXISTS

(SELECT книга.[Код книги]

FROM абонемент книга

WHERE абонемент.[Код книги]=книга.[Код книги]

AND книга.[Код книги]=6);

1. Знайти всіх читачів, що на даний момент тримають 5 книжок.

# SELECT DISTINCTROW читач.Прізвище, читач.[Ім”я], факультет.[Номер факультету], COUNT (абонемент.[Код книги]) AS [COUNT\_Код книги]

FROM (факультет INNER JOIN [читач] ON факультет.[Номер факультету]=читач.Факультет) INNER JOIN (книга INNER JOIN абонемент ON книга.[Код книги]=абонемент.[Код книги] ON читач.[Код читача]=абонемент.[Код читача]

GROUP BY читач.Прізвище, читач.[Ім”я], факультет.[Номер факультету]

HAVING (((COUNT (абонемент.[Код книги]))=5));

1. Знайти всі книги, що були видані у видавництві такому-то.

SELECT DISTINCTROW книга.Назва, книга.Автор

FROM книга INNER JOIN (видавництво INNER JOIN [видавництво-книга] ON видавництво.[Код видавництва]=[видавництво-книга].[Код видавництва]) ON книга.[Код книги]=[видавництво-книга].[Код книги]

WHERE (((видавництво.[Назва видавництва])=[Введіть назву видавництва]))

GROUP BY книга.Назва, книга.Автор;

11) Запит на видалення. З таблиці “боржники” видаляються записи про читачів,

якщо відмічено, що вони повернули взяті книги та від дня їх повернення до

поточного дня пройшло більше 20 днів, тобто, як зазначалось раніше, ще 20

днів після повернення книги бувший боржник обслуговуватись не буде.

DELETE DISTINCTROW DATE()-[Дата повернення] AS Вираз1, боржники.[Код читача] AS Вираз2, боржники.[Код книги] AS Вираз3, боржники.[Дата взяття] AS Вираз4, боржники.[Дата повернення] AS Вираз5, боржники.[Повернення книги]

FROM боржники

WHERE (((DATE()-[Дата повернення])>20 AND ((боржники.[Повернення книги])=YES));

1. Запит на додавання. Перевіряється таблиця “абонемент”. Якщо там є читачі, які тримають книгу більше 13 днів, то вони заносяться список боржників.

INSERT INTO боржники([Дата взяття книги],[Код книги],[Код читача])

SELECT DISTINCTROW абонемент.[Дата взяття книги] AS Вираз1, абонемент.[Код книги] AS Вираз2, абонемент.[Код читача] AS Вираз3

FROM абонемент

WHERE ((( DATE()-[Дата взяття книги])>13));

1. Запит на оновлення. Для боржника, який здав книгу, робиться помітка про здачу.

UPDATE DISTINCTROW [читач] INNER JOIN боржники ON читач.[Код читача]=боржники.[КОд читача]

SET боржники.[Повернення книги]=YES

WHERE ((([читач].[Прізвище]&” “&[Ім”я]=[Введіть прізвище та ім”я]));

1. Запит на створення таблиці. Створюється таблиця”куча”, в яку заносяться дані про назву книги, яка знаходиться у другому відділі і випущена в 1992 або 1993 роках.

SELECT DISTINCTROW відділ.[Код відділу], книга.Назва, [видавництво-книга].[Рік видання] INTO [куча]

FROM [книгаї INNER JOIN [видавництво-книга] ON кинга.[Код кинги]=[видавництво-книга].[Код книги]) INNER JOIN (відділ INNER JOIN [відділ-книга] ON відділ.[Код відділу]=[відділ-книга].[Код відділу]) ON книга.[Код книги]=[відділ-книга].[Код книги]

WHERE (((відділ.[Код відділу])=2 AND (([видавництво-книга].[Рік видання])=1992 OR ([видавництво-книга].[Рік видання])=1993));

1. Знайти всіх читачів, що читали принаймі всі ті книжки, що і читач Федосова.

Спочатку створюється підзапит, який видає множину всіх книг, що читає Федосова.

SELECT DISTINCTROW читач.Прізвище, книга.Назва

FROM [читач] INNER JOIN ( книга INNER JOIN абонемент ON книга.[Код книги]=абонемент.[Код книги]) ON читач.[Код читача]=абонемент.[Код читача]

WHERE (((читач.Прізвище)=”Федосова”));

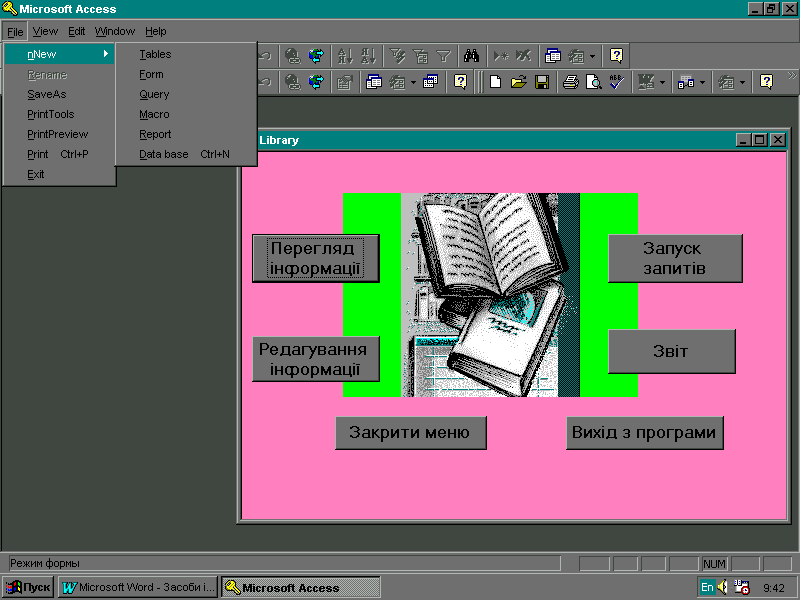
Тепер створюється головний запит на основі підзапиту (sub1).

SELECT DISTINCTROW читач.Прізвище, читач.[Ім”я], книга.Назва

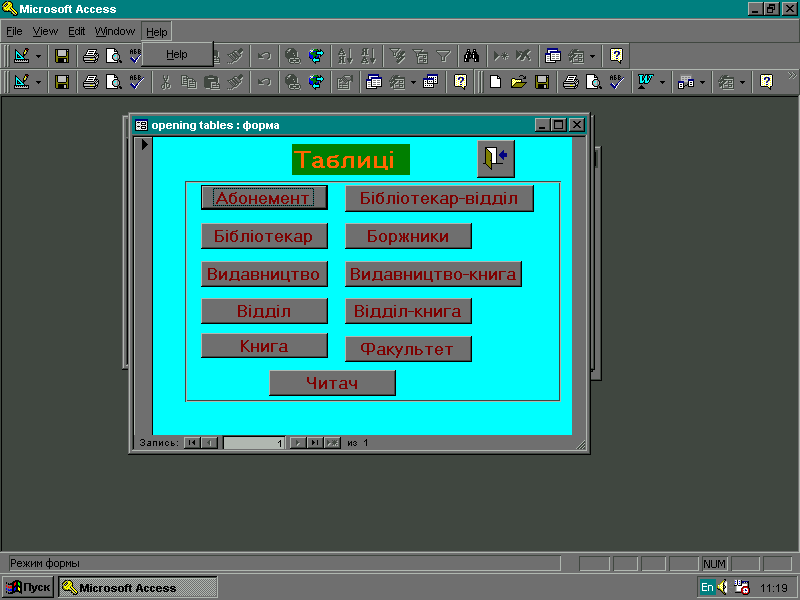
FROM [читач] INNER JOIN (( SUB1 INNER JOIN книга ON SUB1.Назва=книга.Назва) INNER JOIN абонемент ON книга.[Код книги]=абонемент.[Код книги]) ON читач.[Код читача]=абонемент.[Код читача]

*Засоби інтерфейсу.*

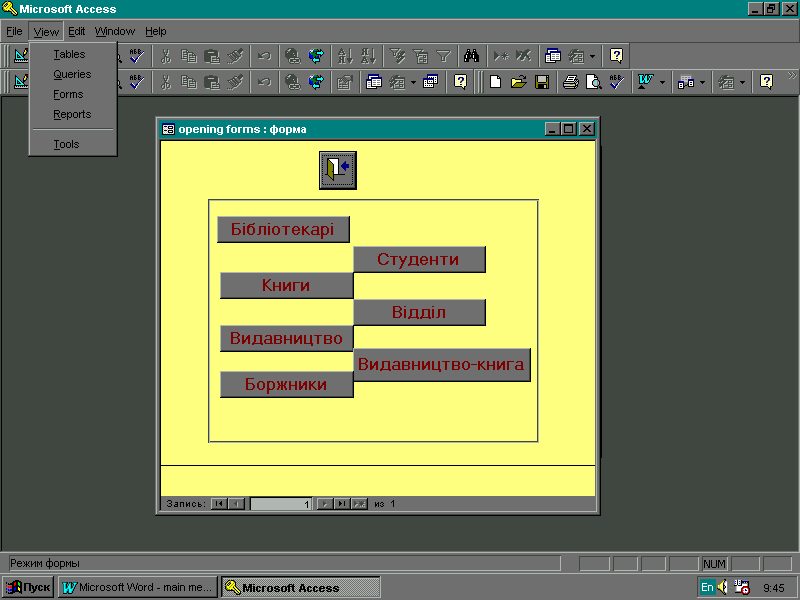
Для створеної бази даних розроблені інтефейсні засоби, які дозволяють працювати з нею і звичайному користувачеві. При запуску програми на екрані перед користувачем в першу чергу з”являється головна форма з шістью управляючими кнопками. Також вгорі екрану замість стандартного меню з”являється власноручно створене меню. В деяких моментах воно дублює форму, тобто з нього можна безпосередньо вибирати ті ж дії, що і по натисненню кнопки в формі, але має і свої відмінності, завдяки чому ним можна порадити користуватися більш досвідченим користувачам.



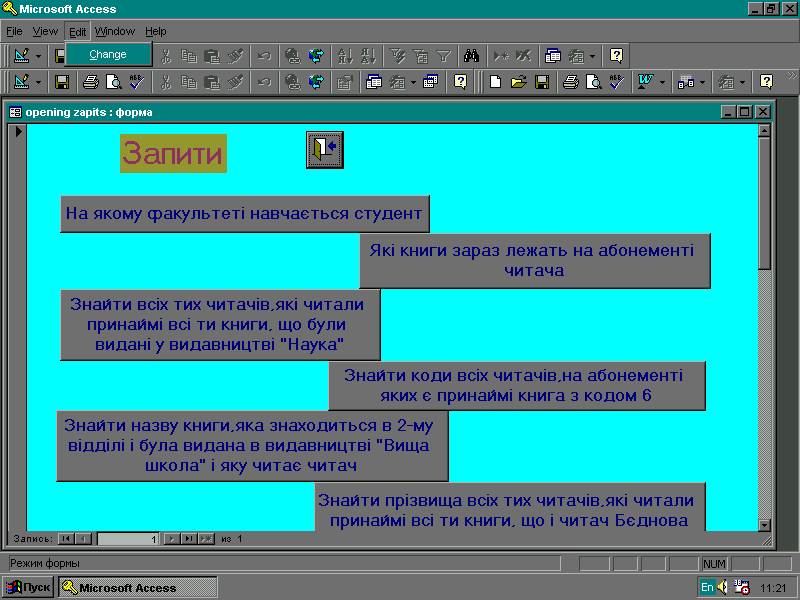
Отже, головне меню має 5 пунктів – File, View, Edit, Window, Help. Меню File має такі підменю: nNew, SaveAs, PrintTools, PrintPreview, Print. Підменю nNew має ще один рівень вкладеності – ми можемо вибрати, що створювати: нову таблицю, форму, запит, макрос, звіт, або, взагалі, нову базу даних. Велика увага приділяється роздруковуванню матеріалів. В пункті PrintTools можна встановити параметри друку. Інший пункт меню – View.Можна вибрати, що продивлятись: таблиці, форми, запити, звіти. Після вибору відповідного пункту відкривається форма, яка дозволяє робити наступний вібір, подаючи відповідно список існуючих таблиць, форм, запитів, звітів.Крім форми, яка виводить список таблиць для подальшого перегляду,



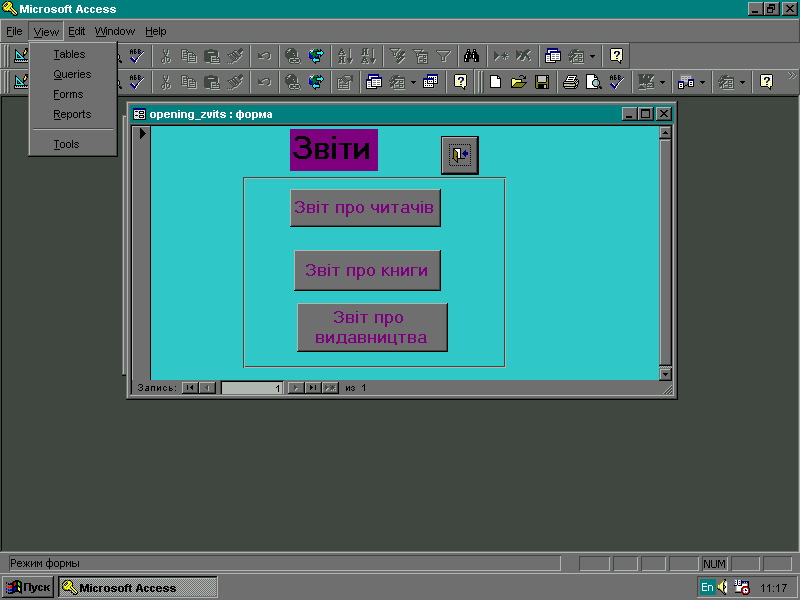
всі інші форми можна викликати натиснувши відповідні кнопки на меню-формі– “Перегляд інформації” – виводиться список існуючих форм,



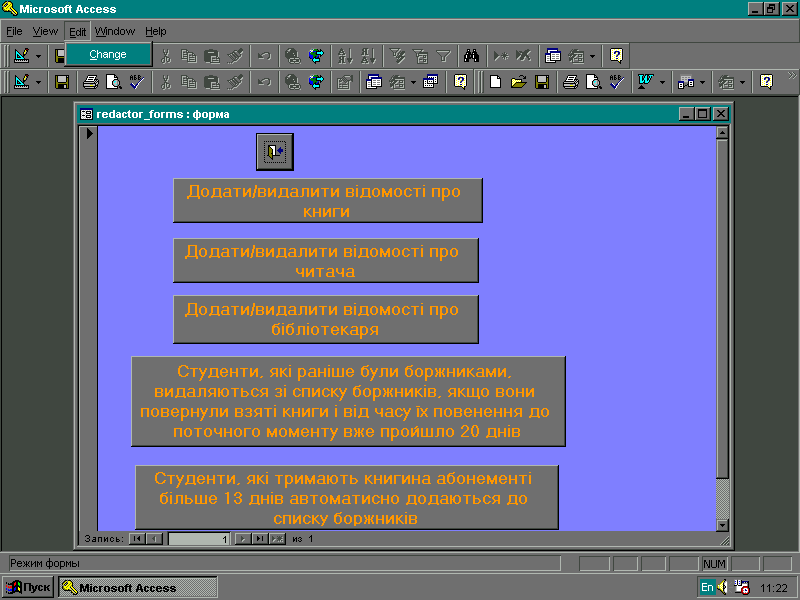
“Запити”- список існуючих запитів,



“Звіт” – список звітів.



В меню View є ще один пункт Tools для встановлення різноманітних параметрів системи. Наступний пукт меню Edit повністю аналогічний за результатом вибору натисненню на кнопку меню-форми “Редагування інформації”. Після його вибору з”являється інша форма, яка пропонує певні розділи інформації для редагування.



За власним вибором можна занести або вилучити якісь відомості про читачів, книги, працівників бібліотеки та ін. Наступний пункт меню – Window. За його допомогою можна поміняти розташування вікон на екрані: Tile, Cascade, Hide, Unhide. Останній пукт меню – Help – при його виборі запускається стандартна допога.

Меню-форма є, можливо, більш зручнішою для користування звичайними користувачами, оскільки вона є більш наочною. Так, натискаючи послідовно кнопки “Перегляд інформації” та наступні запропоновані розділи, можна продивитись майже всю інформацію, яка є в таблицях. Наприклад:

“Перегляд інформації” -> “бібліотекарі”:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Поле6** | **Поле8** | **Поле10** | **Поле12** | **ПолеСоСписком14** |
| Ярошниченко Т.Я. | 45 | 25 | вища | 1 |
| Мирошниченко Я.Т. | 40 | 15 | вища | 2 |
| Фощенко Т.Т. | 35 | 10 | вища | 3 |

Поле для кода бібліотекаря є списком, вибираючи з якого певний код можна відразу ж побачити всі записи про даного бібліотекаря (це зроблено за допомогою функції DlookUp). Дані в цю форму заносяться з таблиці, проте в більшість розроблених форм дані заносяться із запитів.

|  |  |  |  |
| --- | --- | --- | --- |
| **Прізвище** | **Ім"я** | **Повна назва** | **Курс** |
| Пархоменко | Олена | Департамент комп"ютерних технологій | 3 |
| Комарницька | Людмила | Департамент комп"ютерних технологій | 3 |
| Підгорний | Свєтослав | Департамент комп"ютерних технологій | 3 |
| Федосова | Наталя | Департамент комп"ютерних технологій | 3 |
| Тарнавська | Олена | Департамент комп"ютерних технологій | 3 |
| Бубенщиков | Євген | Департамент комп"ютерних технологій | 3 |
| Сиворакша | Сергій | Департамент комп"ютерних технологій | 3 |
| Овдій | Юрій | Департамент комп"ютерних технологій | 3 |
| Сігаєва | Марина | Департамент комп"ютерних технологій | 3 |
| Фіялка | Світлана | Департамент комп"ютерних технологій | 3 |
| Цап | Ольга | Департамент економічної теорії | 3 |
| Жилєвський | Олександр | Департамент економічної теорії | 2 |
| Жарко | Тетяна | Факультет правничих наук | 2 |
| Дундій | Ганна | Факультет гуманітарних і суспільних наук | 3 |
| Галабурда | Юля | Факультет гуманітарних і суспільних наук | 3 |
| Сосновська | Люба | Факультет гуманітарних і суспільних наук | 3 |
| Вахненко | Тетяна | Факультет гуманітарних і суспільних наук | 3 |
| Беднова | Ганна | Факультет гуманітарних і суспільних наук | 3 |
| Медвєдєва | Соломія | Факультет природничих наук | 3 |
| Іванова | Тетяна | Школа соціальної роботи | 1 |

Дана форма дає відомості про студентів. За допомогою кнопок на формі можна передивлятися всі записи. В таблиці “читач” не має запису про назву факультету, де навчається студент, а лише номер факультету, тому для виведення його в форму разом з іншими відомостями був написаний запит.

Форми на редагування інформації в основному використовують макроси.

|  |  |  |
| --- | --- | --- |
| **Назва** | **Автор** | **Код книги** |
| Макроэкономика | Мэнкью | 2 |
| Экономикс | Самюэльсон | 3 |
| Линейное программирование | Юдин | 4 |
| Вища математика | Дубовик | 5 |
| Першотвір і переклад | Коптілов | 6 |
| О всеобщем родстве языков мира | Мельничук | 7 |
| Экономикс | Кэмпбелл | 8 |
| Дослідження операцій | Варнер | 9 |
| Системний аналіз | Пономаренко | 10 |
| Теорія і практика перекладу | Коптілов | 11 |
| Реалія і переклад | Зорівчак | 12 |
| Слово живое и мертвое | Галь | 13 |
| Индоевропейский язык и индоевропейцы | Гамкрелидзе | 14 |
| Сучасне порівняльно-історичне мовознавство | Білецький | 15 |
| Праславянский язык | Хенрик | 16 |
| Избранные работы по языкознанию | Покровский | 17 |
| Сравнительная грамматика индоевропейских языков | Савченко | 18 |
| Лингвистический энциклопедический словарь | Ярцева | 19 |
| Політологія | Дзюбко | 20 |
| Основи загальної теорії права та держави | Рабінович | 21 |
| Основи держави і права | Котюк | 22 |
| The Social Experience | Vander Zanden | 23 |
| Українська мова | Мацько | 24 |

За допомогою цієї форми можна заносити запис про нову книгу до таблиці “книги” або вилучати якісь записи. Це робиться відповідними кнопками , які запускають в роботу необхідний макрос.

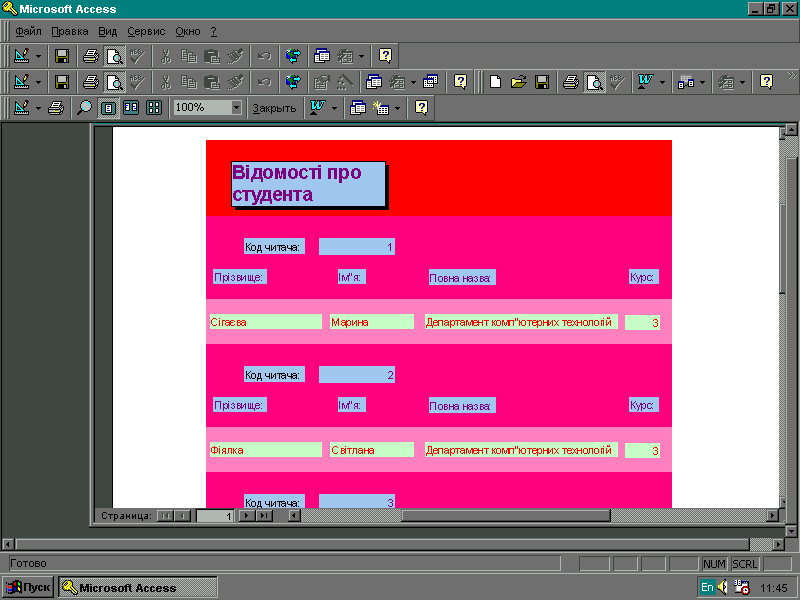
Натиснувши кнопку “Запити” можна запустити будь-який запропонований запит на виконання. Більшість запитів використовують параметри, але їх можна вводити не тільки через стандартне діалогове вікно, а й через форму.

|  |  |  |
| --- | --- | --- |
| **publish** | **year1** | **year2** |
|  |  |  |

Наприклад, ми вибираємо запит –“За назвою видавництва знайти книги, які були там видані між таким-то і таким-то роками. Після натиснення на кнопці з умовою цього запиту, з”являється форма, яка бере на себе обов”язки діалового вікна і пропонує ввести необхідні параметри.

Натискаючи кнопку “Звіт” на формі-меню можна вибрати певний звіт і продивитись його.

**Відомості про**



**студента**

Код 1

Прізвище: Ім"я: Повна Курс:

Сігаєва Марина Департамент комп"ютерних 3

В говному меню в File є також пункт Exit, який завершує роботу з системою ї виходить з Access. Аналогічну дію виконує кнопка “Вихід з програми” на меню-формі”, відмінність лише в тому, що перед виходом вона запрошує підтвердження.

