

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ МІСЬКОГО ГОСПОДАРСТВА

А.А. Євдокімов

МЕТОДИЧНІ ВКАЗІВКИ

до самостійної роботи та лабораторних занять з дисципліни

«Бази даних»

(для студентів 2 курсу денної і заочної форм навчання напряму підготовки 6.080101
«Геодезія, картографія та землеустрій»
спеціальності 6.070900 «Геоінформаційні системи і технології»)

Харків - ХНАМГ - 2009

Методичні вказівки до самостійної роботи та лабораторних занять з дисципліни «Бази даних» (для студентів другого курсу денної і заочної форм навчання напряму підготовки 6.080101 «Геодезія, картографія та землеустрій» спеціальності 6.070900 «Геоінформаційні системи і технології»). / Укл.: Євдокімов А.А.- Харків: ХНАМГ, 2009 - 60 с.

Укладач: А.А. Євдокімов

Рецензенти: к.т.н., доц. Л.К. Войславський

Рекомендовано кафедрою геоінформаційних систем і геодезії,
протокол № 9 від 19 травня 2009 р.

ЗМІСТ

Передмова.....	4
Вступ.....	5
Лабораторна робота №1.....	6
Лабораторна робота №2.....	11
Лабораторна робота №3.....	13
Лабораторна робота №4.....	16
Лабораторна робота №5.....	19
Лабораторна робота №6.....	22
Лабораторна робота №7.....	24
Лабораторна робота №8.....	26
Лабораторна робота №9.....	32
Самостійна робота.....	37
Рекомендована література.....	59

Передмова

Цей методичні вказівки призначені для вивчення СУБД (систем управління базами даних) на прикладі програми ACCESS, що входить у стандартний пакет Microsoft Office. Робота містить короткі теоретичні відомості, завдання до самостійної роботи і лабораторні роботи, які бажано виконувати поряд з лекційним вивченням дисципліни.

Основна мета роботи – дати знання і зміцнити навички, достатні для самостійного вивчення СУБД.

При роботі з даними вказівками необхідно знати і вміти:

- ✓ основні прийоми роботи з операційною системою Windows;
- ✓ текстовий редактор MS-Word (коротке знайомство);
- ✓ табличний процесор MS-Excel (коротке знайомство);
- ✓ основні прийоми роботи з графічними об'єктами Microsoft Office'97;
- ✓ упевнено використовувати маніпулятор «миша»;
- ✓ роботу з контекстним меню.

Вступ

У самому загальному значенні **база даних** – це один або кілька файлів, що містять зв'язану (за змістом) між собою інформацію. Припустимо, в одному файлі ви зберігаєте дати народження ваших друзів, в іншому - ваші спільні фотографії. Або можна створити кілька файлів електронних таблиць, де ви будете враховувати ваші доходи й витрати за певні періоди. Якщо ви організована людина, то, використовуючи спеціальну структуру каталогів і підкаталогів, ви справитеся із сотнею інших файлів. Однак рано, або пізно ви зіштовхнетеся з таким завданням, що буде занадто великим. Що робити? Що робити, якщо якісь дані повинні використовувати кілька додатків або людей, і при цьому не можна допустити, щоб ті самі дані коректували одночасно? Коли ви постаєте перед подібними проблемами, вам потрібна **система управління базами даних** (СУБД).

Майже всі сучасні системи засновані на реляційній (relational) моделі управління базами даних. Назва «реляційна» пов'язана з тим, що кожен запис містить інформацію, яка відноситься тільки до одного об'єкта. У таких базах дані не дублюються, а зв'язуються за визначеними полями. Наприклад, при вказівці товарів і їхніх категорій не доцільно згадувати всі відомості про постачальника, що зберігаються в окремій таблиці. Досить зв'язати ці товари з вище зазначеною таблицею, за номером (кодом) постачальника.

Можна виділити три основні функції СУБД:

♦ **визначення даних** (Data definition) – ви можете визначити, яка саме інформація буде зберігатися у вашій базі даних, задати структуру даних і їхній тип (наприклад, максимальна кількість цифр або символів), а також указати, як ці дані зв'язані між собою. У деяких випадках ви можете також задати формати й критерії перевірки даних;

♦ **обробка даних** (Data manipulation) – дані можна обробляти різними способами. Можна поєднувати дані з іншою зв'язаною з ними інформацією й обчислювати підсумкові значення;

♦ **управління даними** (Data control) – ви можете вказати, кому дозволено знайомитися з даними, коректувати їх або надавати нову інформацію. Можна також визначити правила колективного користування даними.

Лабораторна робота №1

Створення таблиць

Мета роботи:

*одержати поняття про шаблони в MS-ACCESS;
вивчити структуру об'єкта MS-ACCESS “таблиця”;
навчитися задавати полям різні типи даних.*

Вступ

СУБД MS-ACCESS має наступну структуру:

- таблиці – для постійного збереження даних;
- запити – для виклику даних з таблиць за визначеним шаблоном і їхнього подальшого оброблення;
- форми – для зручного введення і перегляду даних;
- звіти – для зручного виведення заданої інформації на друк;
- макроси – для завдання властивостей об'єктів, що відкриваються, і настроювання порядку їхньої роботи;
- модулі – для спрощення виклику раніше створених функцій.

Таблиці складаються з полів і записів. Полями називаються стовпці, а рядки – записами. Внести запис у таблицю означає заповнити даними який-небудь рядок. Щоб створити таблицю необхідно визначити її поля, типи даних цих полів і, іноді, деякі додаткові властивості цих полів. Не всі дані займають у комп'ютері однакове місце. Для їхнього компактного збереження необхідно чітко визначити: що це текст або число, дата або логічний символ. У базах даних під кожен тип даних резервується деякий простір, і якщо відомо наперед, що він не буде використано до кінця, його необхідно зменшити. Як це зробити, ви побачите по ходу виконання даної роботи.

Хід роботи

1. Запустіть програму **MS-ACCESS**.
2. У діалоговому вікні, що з'явилося, встановіть прапорець напроти **опції нова база даних** і підтвердіть операцію.
3. У наступному діалоговому вікні ви повинні вибрати папку для бази даних і задати її ім'я (**Європа**).

На екрані з'явиться вікно із шістьма вкладками, це і є ваша база даних (вона поки порожня).

4. Перейдіть на вкладку **таблиці**.
5. Натисніть мишею по кнопці **створити**
6. У діалоговому вікні, що з'явилося, виберіть режим **створення (конструктор)** і підтвердіть дану операцію.

*У стовпець **назва поля** ми будемо заносити імена стовпців нашої майбутньої таблиці (при цьому не можна використовувати деякі символи, у тому числі крапки і коми). У стовпці **тип даних** будемо вибирати (використовуючи кнопку виклику списку) тип даних. А те, що заноситься у стовпець **опис**, потім з'являється у вигляді коментарів, у рядку стану (для перевірки в одному з рядків цього стовпця напишіть фразу: моя перша база даних).*

З нижче перерахованих даних необхідно створити наступні поля:

Назва поля	Тип даних
Код країни	Лічильник
Країна	Текстовий
Столиця	Текстовий
Площа	Числовий
Населення	Числовий
Релігія	Текстовий
Гроші	Текстовий
Лад	Текстовий

Дані для введення:

Албанія

Площа - 28 748 кв. км
 Столиця - Тирана
 Число жителів - 3 149 000 чол.
 Основна релігія - Атеїзм
 Грошова одиниця - Лек
 Держ. Лад - Республіка

Андорра

Площа - 468 кв. км
 Столиця Андорра-ла-В'єха
 Число жителів - 51 400 чол.
 Основна релігія - Християнство
 (католики)
 Грошова одиниця - Франк
 Держ. лад Князівство

Бельгія

Площа - 30 518 кв. км
 Столиця - Брюссель
 Число жителів - 9 865 000 чол.
 Основна релігія - Християнство
 (католики)
 Грошова одиниця - Франк
 Держ. лад - Монархія

Франція

Площа -543 965 кв. км
 Столиця -Париж
 Число жителів -55 860 000
 Основна релігія -Християнство
 (католики)
 Грошова одиниця -Франк
 Держ. лад -Республіка

Болгарія

Площа - 110 994 кв. км
 Столиця - Софія
 Число жителів - 8 978 000 чол.
 Основна релігія - Християнство
 (православні)
 Грошова одиниця - Лев
 Держ. Лад – Республіка

Данія

Площа - 43 092 кв. км
 Столиця- Копенгаген
 Число жителів- 5 130 000 чол.
 Основна релігія- Християнство
 (протестанти)
 Грошова одиниця -Крона
 Держ. лад -Монархія

Фінляндія

Площа- 338 145 кв. км
 Столиця -Хельсінкі
 Число жителів -4 952 000 чол.
 Основна релігія -Християнство
 (протестанти)
 Грошова одиниця -Марки
 Держ. лад- Республіка

Югославія

Площа 255 804 кв. км
 Столиця -Белград
 Число жителів -23 591 000 чол.
 Основна релігія -Християнство
 (православні)
 Грошова одиниця -Динар
 Держ. лад -Республіка

Греція

Площа -131 957 кв. км

Столиця -Афіни

Число жителів -10 055 000 чол.

Основна релігія -Християнство
(православні)

Грошова одиниця -Драхма

Держ. лад -Республіка

Ірландія

Площа -70 285 кв. км

Столиця -Дублін

Число жителів -3 553 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Фунт

Держ. лад -Республіка

Ісландія

Площа- 103 000 кв. км

Столиця -Рейк'явік

Число жителів -248 000 чол.

Основна релігія -Християнство
(протестанти)

Грошова одиниця -Крона

Держ. лад -Республіка

Італія

Площа -301 277 кв. км

Столиця -Рим

Число жителів -57 401 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Ліра

Держ. лад -Республіка

Монако

Площа- 2 кв. км

Столиця -Монако

Число жителів -28 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Франк

Держ. лад -Князівство

Ліхтенштейн

Площа- 160 кв. км

Столиця -Вадуц

Число жителів -27 840 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Франк

Держ. лад -Монархія

Люксембург

Площа 2 586 кв. км

Столиця Люксембург

Число жителів -372 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Франк

Держ. лад -Герцогство

Мальта

Площа- 316 кв. км

Столиця -Валлетта

Число жителів -347 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Фунт

Держ. лад -Республіка

Румунія

Площа- 237 500 кв. км

Столиця -Бухарест

Число жителів -23 014 000 чол.

Основна релігія -Християнство
(православні)

Грошова одиниця -Лей

Держ. лад -Республіка

Сан-Марино

Площа- 61 кв. км

Столиця -Сан-Марино

Число жителів -22 830 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Ліра

Держ. лад -Республіка

Нідерланди

Площа- 41 863 кв. км
Столиця -Амстердам
Число жителів -14 741 000 чол.
Основна релігія- Християнство
(католики)
Грошова одиниця -Флорин
Держ. лад -Монархія

Норвегія

Площа- 323 878 кв. км
Столиця -Осло
Число жителів -4 202 000 чол.
Основна релігія -Християнство
(протестанти)
Грошова одиниця -Крона
Держ. лад -Монархія

Польща

Площа- 312 683 кв. км
Столиця -Варшава
Число жителів -37 864 000 чол.
Основна релігія -Християнство
(католики)
Грошова одиниця -Злотий
Держ. лад -Республіка

Португалія

Площа- 92 389 кв. км
Столиця -Лісабон
Число жителів -10 349 000 чол.
Основна релігія -Християнство
(католики)
Грошова одиниця -Ескудо
Держ. лад -Республіка

Швеція

Площа- 449 964 кв. км
Столиця -Стокгольм
Число жителів -8 415 000 чол.
Основна релігія -Християнство
(протестанти)
Грошова одиниця -Крона
Держ. лад -Монархія

Швейцарія

Площа- 41 293 кв. км
Столиця -Берн
Число жителів -6 626 000 чол.
Основна релігія -Християнство
(католики)
Грошова одиниця -Франк
Держ. лад -Республіка

Іспанія

Площа- 504 783 кв. км
Столиця -Мадрид
Число жителів -38 996 000 чол.
Основна релігія -Християнство
(католики)
Грошова одиниця -Песета
Держ. лад -Монархія

Великобританія

Площа- 244 110 кв. км
Столиця -Лондон
Число жителів -57 006 000 чол.
Основна релігія Християнство
(протестанти)
Грошова одиниця -Фунт
Держ. лад -Монархія

Угорщина

Площа- 93 031 кв. км
Столиця -Будапешт
Число жителів -10 591 000 чол.
Основна релігія -Християнство
(католики)
Грошова одиниця -Форинт
Держ. лад –Республіка

Німеччина

Площа- 357 042 кв. км
Столиця -Бонн
Число жителів -77 370 000 чол.
Основна релігія -Християнство
(протестанти)
Грошова одиниця -Марка
Держ. лад -Республіка

Чехословакія

Площа- 127 900 кв. км

Столиця -Прага

Число жителів -15 604 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця -Крона

Держ. лад -Республіка

Австрія

Площа- 83 857 кв. км

Столиця -Вена

Число жителів- 7 557 000 чол.

Основна релігія -Християнство
(католики)

Грошова одиниця- Шилінг

Держ. лад -Республіка

Після введення полів і типів даних бажано задати ключове поле. Оскільки значення в ключовому полі повинні бути унікальними, тобто не повторюваними, то в цій якості слід вибрати поле **код країни**.

7. Для цього необхідно клацнути правою клавішею миші по заданому полю і у з'явившомуся меню виконати команду **ключове поле**.

8. Закрийте вікно конструктора і при запиті про збереження задайте ім'я Країни Європи.

9. Відкрийте таблицю для заповнення даними. (Помніть, що поле з типом даних «лічильник» буде заповнюватися самостійно).

10.Порахуйте максимальну кількість символів у кожному з полів з текстовими даними і, про всякий випадок, додайте до значень 5.

11.Відкрийте таблицю в режимі конструктора.

12.По черзі, переводячи текстовий курсор у кожне з текстових полів, задайте у вікні властивостей поля його розмір (значення з пункту 10).

13.Перегляньте, які ще налаштування можна задавати текстовим полям.

14.Перегляньте, які налаштування можна задавати іншим полям.

15.Закрийте таблицю зі збереженням.

16.Збережіть вашу базу даних для наступної лабораторної роботи за адресою - C:/Student/GICiT/GICiT-20XX-X (або за адресою, вказаною викладачем на дошці). На відміну, від раніше вивчених програм, для цього треба закрити ACCESS і скопіювати файл бази одним із засобів Windows.

Контрольні запитання

1. Що називається базою даних?
2. Чим поле відрізняється від запису?
3. Поясніть термін СУБД та основні функції СУБД?
4. Які структури ACCESS ви знаєте?
5. Для чого служить структура "таблиця"?
6. Для чого даним задаються різні типи?
7. Що можна налаштувати у властивостях поля таблиці?

Лабораторна робота №2

Реляційні бази даних

Мета роботи:

одержати поняття про реляційні бази і плюси, надані ними.

Вступ

У процесі введення даних часто виникає необхідність захистити оператора від помилки і розмежувати доступ до важливої інформації. Для цього дані різних категорій поділяють по різних таблицях, крім цього, як правило, такий поділ дозволяє більш компактно зберігати інформацію. Дані в головній таблиці індексують і, використовуючи ці індекси, зв'язують з підлеглими таблицями. Бази даних з такими зв'язками називають реляційними. У даній лабораторній роботі ми будемо приводити створену раніше базу даних до такого виду.

Хід роботи

Поділ даних на дві таблиці

1. Відкрийте базу даних **Європа**
2. Створіть таблицю **Релігія** з полями:

Назва поля	Тип даних
Код_релігії	Лічильник
Релігія	Текстовий

3. Поле **Код_релігії** зробіть ключовим
4. Створіть таблицю **Лад** з полями:

Назва поля	Тип даних
Код_ладу	Лічильник
Лад	Текстовий

5. Поле **Код_ладу** зробіть ключовим
6. Заповніть ці таблиці (таким чином навпроти кожної релігії і кожного ладу буде стояти його код)
7. Відкрийте таблицю **Країни Європи** і змініть назви в полях **Лад** і **Релігія** на відповідні їм коди в раніше створених таблицях (для автоматизації спробуйте використовувати команду заміни з меню виправлення)
8. Відкрийте таблицю **Країни Європи** в режимі конструктора
9. Для полів **Лад** і **Релігія** змініть тип даних на числовий
10. Одночасно установивши їм розмір поля **Довге ціле** (тому що поля, що зв'язуються, як правило, повинні мати однаковий тип даних)

Установка зв'язку між двома таблицями

1. Виконайте команду Схема даних з меню Сервіс
2. У діалоговому вікні додавання таблиць додайте в схему всі три таблиці
3. На полі Код_релігії таблиці Релігія натисніть ліву клавішу миші й, утримуючи її, перетягніть на поле Релігія таблиці Країни Європи

У діалоговому вікні, що з'явилося, необхідно установити потрібний зв'язок: прапорець навпроти **опції забезпечення цілісності** даних означає, що перед тим як занести дані в підлеглу таблицю, програма буде перевіряти їх на відповідність головній. (Таблиця **Країни Європи** є підлеглою для таблиць **Релігія** і **Лад**). Прапорець напроти опції **каскадне відновлення зв'язаних полів** означає, що зміни в головній таблиці автоматично будуть впливати на підлеглу. Прапорець навпроти опції **каскадне видалення зв'язаних полів** означає, що поля, вилучені в головній таблиці, будуть вилучені й у підлеглій.

4. Установіть всі ці прапорці
5. Аналогічну операцію проробіть з таблицями **Країни Європи** і **Лад** (Якщо зв'язок не встановлюється, ще раз перевірте типи даних полів, що зв'язуються)
6. Закрийте схему зі збереженням
7. Спробуйте в таблиці **Країни Європи** в полях **Лад** і **Релігія** поміняти коди на неіснуючі в головних таблицях. Вийшло? Чому?
8. Збережіть вашу базу даних за адресою - C:/Student/ГІСіТ/ГІСіТ-20XX-X (або за адресою, вказаною викладачем на дошці).

Контрольні запитання

1. Для чого дані розбивають на кілька таблиць?
2. Яка необхідна умова для створення зв'язку із забезпеченням цілісності даних?
3. Як при розбивці досягається компактність збереження інформації?

Лабораторна робота 3

Створення запитів на вибірку

Мета роботи:

*Вивчити структуру баз даних **Запит**;
навчитися користуватися найпростішими логічними операндами;
одержати практику вибірки заданої інформації.*

Вступ

Таблиці тільки зберігають дані, але необхідно мати можливість вибрати ці дані з декількох таблиць. Саме для цього служать **запити на вибірку**.

У **запитах на вибірку** дані можуть: відбиратися за багатьма критеріями; сортуватися; з ними можуть виконуватися обчислювальні операції.

Запит - це тимчасова таблиця. Це значить, що дані в них не зберігаються постійно, а тільки тимчасово викликаються з таблиць за заздалегідь заданим шаблоном у момент активізації запиту. Таким чином, у базі даних постійно зберігаються тільки шаблони виклику даних (тимчасові таблиці віддаляються після закриття запиту), а сама інформація не дублюється.

Під час створення у запиті необхідно залишити посилання на таблицю, з якої будуть викликатися дані (додавання таблиць у запит). Якщо потрібна таблиця не додана, то перед відкриттям запиту необхідні дані будуть запитуватися в діалозі.

Хід роботи

1. Відкрийте базу даних **Європа**
2. Перейдіть на вкладку **Запити**
3. Створіть запит, вибравши режим конструктора (з'явиться вікно конструктора запиту з діалоговим вікном додавання таблиць. Вікно додавання таблиць можна викликати командою **Додати таблицю** з меню **Запит**).
4. Додайте в запит усі три таблиці

Поле:	Имя таблицы:	Сортировка:	Вывод на экран:	Условие отбора:
			<input type="checkbox"/>	
			<input type="checkbox"/>	
			<input type="checkbox"/>	
			<input type="checkbox"/>	

5. Переконайтеся, що між доданими таблицями автоматично встановився зв'язок (див. рисунок)

Далі необхідно вибрати поля для запиту. З таблиці **Країни Європи** візьмемо поля: **Країна, Столиця, Площа, Населення, Гроші**, з таблиці **Лад** - поле **Лад**, а з таблиці **Релігія** - поле **Релігія**. Це викликано тим, що останні два поля в таблиці **Країни Європи** зберігаються в кодовому вигляді, а в запиті ми можемо дозволити собі зручний перегляд.

1. Перше поле додайте, зробивши подвійного кліка по ньому в таблиці
2. Друге перетягніть, використовуючи мишу (є і третій спосіб, але він менш зручний).
3. Інші поля додайте будь-яким способом.
4. Закрийте запит, при закритті задавши йому ім'я **Основний**
5. Відкрийте запит і перегляньте, які дані він виводить
6. Закрийте запит

Тепер наше завдання - настроїти запит на різні умови вибірки. Звичайно для цього використовують один запит, змінюючи ці умови. Але для перевірки виконаної роботи ми створимо кілька запитів (на кожен умову по запиту). Щоб не створювати заново майже однакові запити, зручно кілька разів скопіювати **Основний** через буфер обміну.

Для того, щоб вивести в запиті тільки країни з визначеною релігією, необхідно в поле запиту **Релігія** в термін **Умова добору** ввести цю релігію. Аналогічно для кожного поля.

Для того, щоб заборонити країнам з визначеною релігією виводитися в запиті, необхідно в даному полі і вже зазначеному рядку ввести задану релігію, але перед нею поставити оператор **Not**.

Для того, щоб вивести країни з населенням більше 1000000, досить у поле **Населення** в рядок **Умова добору** ввести **> 1000000**. До речі, для умов передбачено два рядки, тому для одного поля можна вводити дві умови.

Можна зробити так, щоб перед тим, як вивести запит програма просила ввести дані, рядки з якими будуть виведені. Для цього в потрібному полі у зазначеному рядку треба ввести [Введіть дані для добору]. Вираження в дужках може бути будь-яким, але саме воно з'явиться в діалоговому вікні.

Для того, щоб знаходити в полі не конкретне значення, а тільки його фрагмент, використовують оператор **Like**. Його ставлять перед заданим фрагментом а до або після фрагмента у цьому випадку можна використовувати зірочки маски. Цей оператор не завжди коректно працює, якщо офіс давно не переустановлювався.

7. Виділіть запит Основної (він повинний бути закрити)
8. Скопіюйте його в буфер обміну
9. Уставте його в базу 6 разів, задаючи імена співзвучні наступної нижче завданням.

10. Створіть запит, що виводить країни з Православ'ям
11. Створіть запит, що не виводить країни з Православ'ям
12. Створіть запит, що виводить країни з населенням більше 10000000 і менше 200000000
13. Створіть запит, що виводить країни з населенням більше 3000000 і площею менше 30000 кв. км.
14. Створіть запит, який питає: країни з якою грошовою одиницею вивести на екран?
15. Створіть запит, що виводить країни з грошовою одиницею, яка містить букву до
16. Збережіть вашу базу даних за адресою - C:/Student/GiCiT/GiCiT-20XX-X (або за адресою, вказаною викладачем на дошці).

Контрольні запитання

1. Для чого служать запити на вибірку?
2. Чому запити не займають у базі даних багато місця?
3. За якою кількістю критеріїв можна відбирати інформацію в одному полі запиту?
4. Для чого в запит уставляють посилання на таблиці?
5. Які способи вставки полів у запит ви знаєте?
6. Для чого служить оператор Like?

Лабораторна робота №4

Створення запитів з полями, що обчислюються

Мета роботи:

*Практика створення баз даних;
вивчення побудови математичного вираження у структурі запит;
вивчення групових операцій у запитах;
практика роботи з контекстним меню.*

Вступ

Як відомо, бази даних потрібні не тільки для збереження, але і для обробки даних. Одним із способів є створення полів, що обчислюються, у запитах на вибірку. На прикладі цієї лабораторної роботи ви побачите, що цього способу іноді буває достатньо для створення бази даних яка має практичне значення. Сьогодні ми будемо розраховувати і заносити в базу дані з продажу товарів і прибутку маленького магазину. У даній роботі опущені багато підказок з попередніх тем. Вам самим треба бути вирішувати, які поля робити ключовими і т.п.

Хід роботи

Частина 1 (підготовка)

1. Створіть базу даних **Магазин**
2. Створіть таблицю **Товар** за наведеному нижче прикладом.

код товару	товар	ціна постачання \$
1	монітор	150,00
2	вінчестер	120,00
3	CD-ROM	70,00
4	халява	1,00

Лічильник текстовий числовий ← типи даних

3. Створіть таблицю **Продаж товару по даті** за наведеним нижче прикладом

Примітка 1: Після задання типу даних для поля **націнка** переведіть текстовий курсор у поле **Розмір поля** і за допомогою **кнопки списку, що випадає**, виберіть режим з **плаваючою точкою, (4 байт)**.

Примітка 2: Після завдання типу даних для поля **дата** переведіть текстовий курсор у поле **Формат поля** і за допомогою **кнопки списку, що випадає**, виберіть режим **Короткий формат дати**.

код товару	кількість	націнка	дата
1	5	0,2	04.10.99
1	5	0,2	05.10.99
2	2	0,2	04.10.99
3	10	0,2	04.10.99
4	2	0,02	02.10.99

числовий числовий числовий дата/час ← типи даних

4. Створіть таблицю **Курс долара** за наведеним нижче прикладом

дата	курс \$
04.10.99	25р.
05.10.99	26р.
06.10.99	27р.
02.10.99	24р.
03.10.99	0р.

дата/час грошовий ← типи даних

5. Використовуючи команду **Схема даних** з меню **Сервіс** зв'яжіть таблиці **Товар** і **Продаж товару по даті** по полю код товару

6. Створіть у режимі Конструктора запит **прибуток по даті за вид товару** з полями **товар** з таблиці **Товар** і **дата** з таблиці **Курс долара**.

УВАГА: Переконаєтесь, що в запит уставлені саме ці таблиці.

Крім того, для подальшої роботи необхідно, щоб у цей запит була вставлена таблиця **Продаж товару за датою**.

7. Установіть (перетаскуванням миші) зв'язок між полями **Дата** таблиць **Продаж товару за датою** і **Курс долара**.

8. Збережіть запит під зазначеним ім'ям.

Частина 2 (побудова поля, що обчислюється,)

Наше завдання (як видно з назви запиту) побудувати в наступному стовпці даного запиту вираження для обліку прибутку за видом товару за визначену дату в карбованцях. Для цього необхідно помножити стовпець цін у доларах на курс долара за датами, помножити добуток, що вийшов, на стовпець коефіцієнтів націнки і помножити на кількість проданого товару за датою.

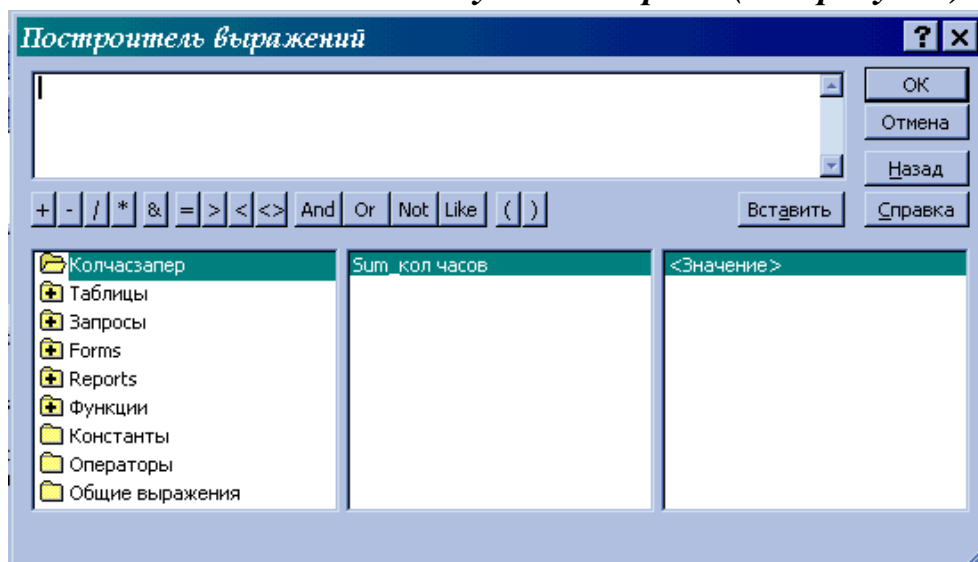
Вирішення задачі:

1. Відкрийте запит **прибуток за датою за вид товару** в режимі конструктора.

2. Установіть текстовий курсор у рядок **Поле** вільного стовпця і щикликом правої клавіші миші на ньому викличте контекстне меню.

3. Виконайте команду **Побудувати**

*Примітка: з'явиться діалогове вікно **Будівник виразів** (див. рисунок).*



4. У даному діалоговому вікні, відкриваючи папки Таблиці і Запити, а потім задані таблиці і запити, необхідно кнопкою вставляти у вікно потрібні поля (з цих таблиць і рядків), а також знаки необхідних математичних операцій.

5. Підготувавши вираз, як зазначено в описі до другої частини, натиснути підтверджуючу клавішу.

6. Установити прапорець (щигликом миші) у поле Висновок на екран створюваного стовпця.

7. Збережіть змінений запит (Файл – зберегти).

8. Переведіть текстовий курсор у початок рядка **Поле** створеного стовпця.

9. Видаліть назву поля (Вираження 1), зберігши двокрапку.

10. Введіть задану назву поля: **прибуток за датою за вид товару**

11. У меню Вид виконайте команду Режим таблиці

12. Якщо побудоване поле виводить дані, закрийте запит зі збереженням, інакше закрийте без збереження.

Ми домоглися обліку прибутку магазина за видом товару за певну дату. Тепер треба врахувати повний прибуток магазина за датою. Для цього застосовується функція угруповання у створюваному запиті. У такий запит звичайно вставляють поле або поля, по яких буде виконуватися угруповання і поле, яке по цих угрупованнях необхідно враховувати. У нашому випадку групувати ми будемо по полю **Дата**, а враховувати (застосовуючи функцію суми) поле, побудоване в попередньому запиті.

Вирішення задачі:

1. Створити запит, додавши в нього тільки попередній запит (з нього будемо брати дані).

2. Уставити з нього поля **Дата** і поле, побудоване в попередньому запиті.

3. Вибрати на панелі інструментів або в контекстному меню операцію угруповання (у запиті повинен додатися рядок **Групова операція:**)

4. Установити текстовий курсор в осередок на перетинанні цього рядка і стовпця, побудованого в попередньому запиті.

5. Клацнути в ній по кнопці списку, що випадає, і вибрати з нього функцію **sum**.

6. Закрити запит зі збереженням.

7. Збережіть базу даних на дискеті.

Контрольні запитання

1. Як викликається контекстне меню?

2. Для чого призначена операція угруповання?

3. Що значить зв'язати поля зі збереженням цілісності даних?

4. Чи можна установити такий зв'язок між полями з різними типами даних і якщо ні, то чому?*

Лабораторна робота №5 Створення і конструювання форм

Мета роботи:

*знайомство зі структурою **форма**;
практика створення форм;
практика конструювання форм;
знайомство з режимом **майстра**.*

Вступ

Форма- це структура бази даних, що дозволяє зручно вводити й обробляти задану інформацію. Дуже часто, для компактного збереження інформації, потрібно заносити в таблицю код об'єкта, а не його ім'я. Таким прикладом може служити таблиця Продаж товару за датою з попередньої лабораторної роботи. Заповнення таких таблиць, особливо для рядових операторів, було б зв'язане з великими проблемами. На щастя, форми дозволяють, вибираючи зі списку імена об'єктів, вставляти в таблиці їхні коди. Можливості форм неосяжні для даного практикуму. У ньому ми познайомимось лише з деякими з них. Крім того, в цій роботі ми познайомимось з деякими елементами візуального програмування, коли беруть готові елементи управління (поля, поля зі списками, кнопки і т.п.) і потім їм задають визначені властивості, а також зв'язку з даними, що знаходяться в таблицях і запитах. Дана тема буде продовжена й у наступних роботах.

Хід роботи

Частина 1 (підготовка)

1. Відкрийте базу даних **Магазин**.
2. Переконайтеся в наявності наступних нижче таблиць.

Товар.

код товару	товар	ціна постачання \$
1	монітор	150,00
2	вінчестер	120,00
3	CD-ROM	70,00
4		1,00

лічильник текстовий числовий ← типи даних

Продаж товару за датою

код товару	кількість	націнка	дата
1	5	0,2	04.10.99
1	5	0,2	05.10.99
2	2	0,2	04.10.99
3	10	0,2	04.10.99
4	2	0,02	02.10.99

числовий числовий числовий дата/час ← типи даних

Курс долара

дата	курс \$
04.10.99	25р.
05.10.99	26р.
06.10.99	27р.
02.10.99	24р.
03.10.99	0р.

дата/час грошовий ← типи даних

1. Перевірте (у схемі даних) установку зв'язку між полями **код товару** в таблицях **Товар** і **Продаж товару за датою**.

Частина 2 (створення форми)

Форми і звіти є досить складними структурами баз даних, тому звичайно їх створюють у режимі **Майстра** і допрацьовують у режимі **конструктора**. Так зробимо і ми.

1. Перейдіть на вкладку **форми**.
2. Клацніть по кнопці **створити** і виберіть режим **майстер форм**.
3. У поле зі списком **Таблиці/запити** виберіть таблицю – джерело

Продаж товару за датою.

4. Використовуючи клавішу « >> », перенесіть усі поля даної таблиці з вікна **доступні поля** у вікно **обрані поля** і перейдіть до наступного діалогового вікна.

5. У ньому установіть прапорець навпроти опції **в один стовпець** і перейдіть до наступного діалогового вікна.

6. Виберіть у ньому оформлення і перейдіть до наступного діалогового вікна.

7. В останньому вікні задайте ім'я формі і натисніть на кнопку **Готово**.

Ви створили форму, що дозволяє більш зручно вводити дані в таблицю **Продаж товару за датою**.

Але для її заповнення необхідно пам'ятати коди всіх товарів. Тому наше наступне завдання - забезпечити зручне введення даних у поле **код товару** таблиці **Продаж товару за датою**.

Частина 3 (добробка форми)

Ми повинні забезпечити, щоб при введенні назви товару в нашій формі у таблицю заносився його код.

1. Відкрийте вашу форму в режимі **конструктора**.
2. Виділіть поле **код товару** і видаліть його, використовуючи клавішу

Delete.

3. Знайдіть у вікні програми панель елементів і перепишіть можливі для вставки об'єкти.

4. З них (щигликом миші) виберіть **поле зі списком**.

5. Підведіть курсор миші до передбачуваного місця вставки нового поля (врахуйте, що ліворуч від нього автоматично вставиться заголовок).

6. Натисніть ліву клавішу миші і, утримуючи її, розтягніть об'єкт як малюнок (після того, як ви відпустите її, у вас з'явиться перше вікно діалогу) .

7. Ознайомтеся з ним і установіть прапорець навпроти потрібної опції (перша).

8. Натиснувши клавішу **Далі**, перейдіть до наступного діалогового вікна.

9. У ньому виберіть таблицю, з якої буде братися найменування товару.

10. У наступному діалоговому вікні виберіть два поля (з одного будуть братися коди товарів, а завдяки іншому ви зумієте бачити при цьому назви цих товарів).

11. Підказка: ці поля **код товару** і **товар**.

12. З наступним діалоговим вікном просто ознайомтеся.

13. У наступному виберіть опцію **Зберегти в поле** і, використовуючи кнопку виклику списку, виберіть з нього **код товару**.

14. В останньому вікні задайте підпис створеного поля (товар).

15. Закрийте форму зі збереженням.

16. Відкрийте її і введіть новий запис.

17. Перевірте, що внеслося в таблицю **Продаж товару за датою** в поле **код товару**.

18. Скопіюйте базу даних на дискету.

Контрольні запитання

1. Для чого служить структура **форми**?
2. Які елементи можна вставляти у форму в режимі конструктора?
3. Як у формі здійснюється перехід між записами?
4. Подумайте, які недоліки є у створеної вами форми?

Лабораторна робота №6

Властивості форм

Мета роботи:

*знайомство зі структурою **форма**;
практика створення форм;
практика конструювання форм;
знайомство з режимом **майстра**.*

Вступ

У попередній лабораторній роботі одне з контрольних питань звучало так: «Подумайте, які недоліки є у створеної вами форми?»; відповідь проста:

- бажаючи додати нові записи, ви можете зіпсувати попередні;
- порядок введення полів запису порушений створенням нового поля;
- шрифт дрібний і невиразний.

Як же забрати ці й подібні їм недоліки?

У більшості об'єктів: Windows і програм цієї операційної системи, є можливість задання якихось параметрів (властивостей). Найпростіший спосіб для їхньої зміни - це викликати діалогове вікно щигликом правої клавіші миші на заданому об'єкті (контекстне меню).

Хід роботи

Частина 1 (Задання властивостей форми і її полів)

1. Зробіть дві копії своєї форми. Для цього:
 - 1.1) у меню **виправлення** виберіть команду **копіювати**;
 - 1.2) у меню **виправлення** виберіть команду **вставити**;
 - 1.3) у діалоговому вікні, що з'явилося, задайте **ім'я введення нових даних**;
 - 1.4) повторіть команду **вставити** і задайте ім'я **видалення записів**;
 - 1.5) своїй першій формі задайте ім'я **редагування даних**.
2. Відкрийте форму введення нових даних у режимі конструктора.
3. Клацніть правою клавішею миші на заголовку вікна форми.
4. Виберіть з контекстного меню команду **властивості**.
5. У діалоговому вікні, що з'явилося, перейдіть на **вкладку все**.
6. Переведіть текстовий курсор у рядок **дозволити зміни** і, використовуючи кнопку виклику списку, виберіть **немає**.
7. Аналогічну операцію проробіть у рядку **дозволити видалення**.
8. Закрийте форму зі збереженням, відкрийте її і спробуйте:
 - а) змінити запис;
 - б) видалити запис;
 - в) додати запис.

9. Подумайте, які зміни треба внести в дві інші форми.
10. Зробіть їх.
11. Відкрийте одну з форм у режимі конструктора.
12. Викличте на заголовку її вікна. контекстне меню.
13. Виконаєте команду **послідовність переходу**.
14. У діалоговому вікні, що з'явилося, змініть послідовність так, щоб поля йшли як вони розташовані усередині форми.
15. Перевірте зроблені зміни.
16. Відкрийте будь-як форму в режимі конструктора.
17. Викличте контекстне меню на одному з полів форми.
18. Виконаєте команду властивості і у вікні, що з'явилося, спробуйте самостійно змінити величину шрифту.

Контрольні запитання

1. Як викликати діалогове вікно властивостей об'єкта?
2. Як ви думаєте: які команди містить контекстне меню?
3. Які команди ви використовували при розмноженні форм?
4. Що ви ще змогли змінити у формі?

Лабораторна робота №7

Створення кнопочних форм

Мета роботи:

*одержати практику щодо створення таблиць;
одержати уявлення про створення оболонок для баз даних.*

Вступ

Після створення реальної бази даних грамотно користуватися нею може тільки творець. Тому для її експлуатації рядовим користувачем необхідно створити оболонку управління зі зрозумілим інтерфейсом (інтерфейс – засобу взаємодії). Для цієї мети підходять так звані кнопочні форми. Ідея в тому, щоб установити зв'язок між визначеною подією і щикликом миші по графічному зображенню кнопки. Однак не всі події можна викликати з використанням кнопочних форм, реалізованих у СУБД ACCESS. Тому для уникнення цієї перешкоди ми скористаємося створенням макросів. Макрос є програмою, що містить макрокоманди, тобто виклик визначених дій із заданими параметрами. Запуск макросу можна виконати з кнопочової форми.

Хід роботи

Частина 1 (підготовка)

1. Створіть базу даних – **Адресна книга**
2. Створіть таблицю – **Адресат** з полями: **код_адресата, ФІО, країна, місто, телефон, e-mail, адреса, примітка.**
3. Створіть форму для заповнення даної таблиці
4. Створіть запит – **пошук на прізвище**, що містить усі поля даної таблиці, за винятком поля **код_адресату**
5. Задайте в цьому запиті умову добору за прізвищем, що вводиться (на перетинанні стовпця **ФІО** і рядка умова добору набрати [Введіть ФІО])

Частина 2 (створення кнопочової форми)

1. У меню **сервіс** зі списку **надбудови** виберіть команду **Диспетчер кнопочних форм**
2. Підтвердить її створення (на екрані з'явилося діалогове вікно, в якому висвітилася назва кнопочової форми, дана їй за замовчуванням).
3. Для зміни даної кнопочової форми клацніть по кнопці **змінити**
4. У наступному діалоговому вікні у верхньому полі замініть назву кнопочової форми на назву бази даних (у вас є кнопочова форма, але немає кнопок)
5. Клацніть по кнопці **створити**
6. У вікні, що з'явилося (зміна елемента кнопочової форми), виберіть, використовуючи кнопки списку, що випадає:
7. задану команду (відкриття форми в режимі редагування);

8. форму, яку варто відкривати (вона у вас єдина);
9. і задайте (у верхнім полі) ім'я створюваної кнопки.
- 10.Клацніть ще раз по кнопці **створити**.
- 11.Переконаєтесь, що в полі команда немає команди відкриття запиту.
- 12.Натисніть клавішу скасування.
- 13.Закрийте вікна кнопкової форми.
- 14.Перейдіть на вкладку **Форми**.
- 15.Відкрийте кнопку форму.
- 16.Переконаєтесь, що створена вами кнопка працює.
- 17.Закрийте кнопку форму.

Частина 3 (Створення макросу, який відкриває запит)

1. Перейдіть на вкладку **Макроси**
2. Виконайте команду **Створити**
3. У діалоговому вікні, що з'явилося, в **полі** Макрокоманда, використовуючи кнопку виклику списку, виберіть команду **Відкрити запит**
4. У полі **Аргументи макрокоманди** виберіть ім'я запиту, що відкривається, його режим (таблиця), і режим даних (читання).
5. Закрийте макрос, зберігши його під ім'ям **Виклик**.

Частина 4 (створення кнопкової команди, що запускає макрос)

1. Виконаєте команду **диспетчер кнопкових форм** зі списку **надбудови** меню **сервіс**.
2. У діалоговому вікні, що з'явилося, натисніть **кнопку** змінити.
3. У наступному вікні виконаєте команду **створити**
4. Виконаєте налаштування аналогічно пунктам з частини 2.

Частина 5 (зміна параметрів запуску)

Тепер наше завдання зробити недоступне, при відкритті файлу, вікно бази даних, забезпечивши при цьому автоматичний запуск кнопкової форми. Для цього необхідно проробити наступні дії:

1. Виконати команду параметри запуску з меню **сервіс**
2. У діалоговому вікні, що з'явилося, забрати прапорець **навпроти вікна бази даних**.
3. А, у пункт меню **форма**, виставити кнопку форму.

Контрольні запитання

1. Для чого служать кнопкові форми?
2. Чи можна створити в базі дві кнопкові форми?
3. Для чого створювався макрос?
4. Які події можна задавати по натисканню кнопки в ACCESS?

Лабораторна робота №8

Створення запитів на мові SQL

Мета роботи:

Освоїти прийоми складання запитів з використанням різних операторів мови SQL.

Вступ

Запит SQL — це запит, створюваний за допомогою інструкцій SQL. Мова SQL (Structured Query Language) використовується при створенні запитів, а також для оновлення і управління реляційними базами даних, такими як бази даних Microsoft Access.

Коли користувач створює запит у режимі конструктора запиту, Microsoft Access автоматично створює еквівалентну інструкцію SQL. Фактично, для більшості властивостей запиту, доступних у вікні властивостей в режимі конструктора, є еквівалентні пропозиції або параметри мови SQL, доступні в режимі SQL. При необхідності користувач має нагоду переглядати й редагувати інструкції SQL в режимі SQL. Після внесення змін у запит в режимі SQL його вигляд в режимі конструктора може змінитися.

Деякі запити, які називають запитами SQL, неможливо створити у бланку запиту. Для запиту к серверу, керуючих запитів і об'єднання необхідно створювати інструкції SQL безпосередньо у вікно запиту в режимі SQL. Для підпорядкованого запиту користувач повинен ввести інструкцію SQL в рядок **Поле** або **Умова відбору** в бланку запиту.

Управляючий запит створює, видаляє або змінює таблиці або створює індекси в таблиці бази даних. Наприклад, в наступному управляючому запиті інструкція CREATE TABLE використовується для створення таблиці «Друзі». Наведена інструкція визначає імена і типи даних для полів таблиці і створює для поля «Код» індекс, що робить це поле ключовим.

Підлеглий запит є інструкцією SQL SELECT, вкладеною в запит на вибірку або запит на зміну. Щоб визначити нове поле, дану інструкцію можна ввести в рядок **Поле** у бланку запиту. Щоб вказати для даного поля умову відбору, введіть інструкцію в рядок **Умова відбору**. Підлеглі запити використовуються для наступного:

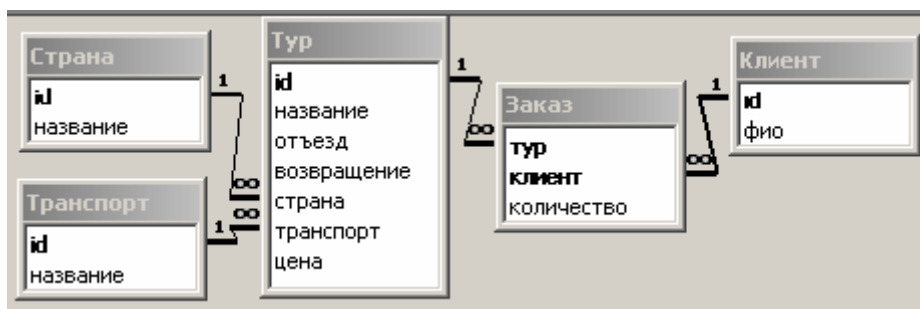
- перевірка наявності результатів підлеглого запиту (використовуються зарезервовані слова EXISTS або NOT EXISTS);
- пошук значень в основному запиті, які рівні перевищують або менше значень, що повертаються підлеглим запитом (використовуються зарезервовані слова ANY, IN або ALL);
- створення підлеглих запитів усередині підлеглих запитів (вкладені запити).

Хід роботи

Послідовність дій:

1. Запустіть Microsoft Access і створіть нову базу даних з ім'ям «ТурАгентство».

База даних повинна містити дані про туристичні поїздки, клієнтів і замовлення. Таблиці бази даних слід створювати засобами мови SQL, без використання візуальних інструментів. Для того, щоб створити таблицю, необхідно спочатку створити запит SQL на створення таблиці. Звичайно при створенні бази даних немає необхідності створювати всі таблиці за допомогою SQL, але іноді оператори SQL можуть бути використані в додатках для створення або зміни структури таблиць автоматично. Структура створюваної бази даних наведена на рисунку.



2. Створіть запит з ім'ям «СтворитьТблТранспорт». У режимі SQL введіть текст запиту на створення таблиці:

```
CREATE TABLE Транспорт (  
    id COUNTER PRIMARY KEY  
    назва TEXT(30)  
);
```

Цей запит створює таблицю «Транспорт», що містить два поля:

id – унікальний числовий ідентифікатор (первинний ключ) запису, значення якого призначаються автоматично (в режимі конструктора таблиць цей тип даних називається «Лічильник»);

назва – текстове поле, яке може містити не більше 30 символів. Таблиця містить назви видів транспорту (літак, автобус і т.п.), вживаних при організації турів.

3. Створіть запит з ім'ям «СтворитьТблКраїна», що містить текст запиту на язиці SQL:

```
CREATE TABLE Країна (  
    id COUNTER PRIMARY KEY  
    назва TEXT(30)  
);
```

Цей запит створює таблицю «Країна». Таблиця міститиме список країн, в які можуть організовуватися тури.

4. Створіть запит з ім'ям «СоздатьТблТур». В режимі SQL введіть текст запиту:

```
CREATE TABLE Тур (  
  id COUNTER PRIMARY KEY  
  назва TEXT(30)  
  від'їзд DATE  
  повернення DATE  
  країна INTEGER REFERENCES Країна (id)  
  транспорт INTEGER REFERENCES Транспорт (id)  
  ціна CURRENCY NOT NULL  
);
```

Запит створює таблицю «Тур». Таблиця міститиме відомості про конкретні туристичні поїздки.

Поля «від'їзд» і «повернення» міститимуть дати початку і закінчення туру відповідно. Для поля «країна» за допомогою ключового слова «**REFERENCES**» («посилається») визначається зв'язок з полем «**id**» таблиці «Країна». Аналогічно, поле «транспорт» зв'язується з полем «**id**» таблиці «Транспорт». Поле ціна (має грошовий тип даних) призначена для зберігання інформації про вартість туру і не може бути порожнім.

5. Створіть запит з ім'ям «СтворитьТблКлієнт», що містить текст запиту:

```
CREATE TABLE Клієнт (  
  id COUNTER PRIMARY KEY  
  фіо TEXT(50)  
);
```

Таблиця «Клієнт» міститиме імена клієнтів, які зверталися в дане туристичне агентство. Текстове поле «**фіо**» може містити до 50 символів.

6. Створіть запит з ім'ям «СтворитьТблЗаказ»:

```
CREATE TABLE Замовлення (  
  тур INTEGER REFERENCES Тур (id)  
  клієнт INTEGER REFERENCES Клієнт (id)  
  кількість INTEGER NOT NULL  
  PRIMARY KEY (тур, клієнт)  
);
```

Таблиця «Замовлення» міститиме відомості про замовлення, оформлені клієнтами даного турагентства на організовані їм туристичні поїздки. Поле «тур» зв'язується з полем «**id**» таблиці «Тур», поле «клієнт» - з полем «**id**» таблиці «Клієнт». Поле «кількість» показує, скільки квитків на вибраний тур замовив даний клієнт. Пара полей «тур» і «клієнт» утворює первинний ключ таблиці. Це означає, що один клієнт може замовити декілька квитків на один і той же тур, але всі вони повинні входити в одне замовлення (неможливо створити інше замовлення на ім'я того ж клієнта на той самий тур).

7. Виконайте створені запити в наступній послідовності: «СтворитьТблТранспорт», «СтворитьТблСтрана», «СтворитьТблТур», «СтворитьТблКлієнт», «СтворитьТблЗаказ». Послідовність створення таблиць не може бути довільною, оскільки поля таблиці «Тур» посилаються на поля таблиць «Транспорт» і «Країна», а поля таблиці «Замовлення» - на поля таблиць «Тур» і «Клієнт».

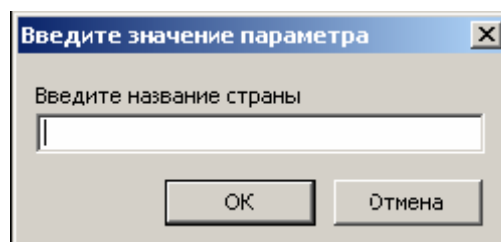
8. Переконайтеся, що всі необхідні таблиці були створені. Дослідіть кожну з таблиць за допомогою конструктора і з'ясуйте, як вживання ключових слів мови SQL (наприклад, «NOT NULL» або «TEXT(30)») вплинуло на властивості полів таблиць.

9. Відкрийте вікно «Схема даних» і переконайтеся, що між таблицями були встановлені всі зв'язки, задані в запитах SQL ключовим словом «REFERENCES».

10. Створіть запит з ім'ям «ДобавитиКраїну». В режимі SQL введіть текст запиту на додавання даних в таблицю:

```
INSERT INTO Країна  
(назва) VALUES ([Введіть назву країни])  
;
```

Цей запит додає в таблицю «Країна» новий запис. Запит містить параметр, тому при його запуску буде видано вікно для введення додаткової інформації.



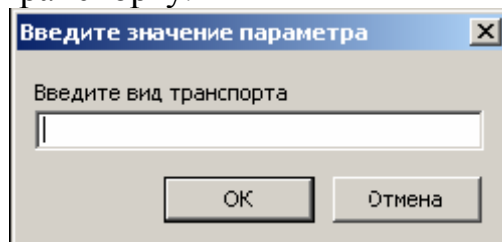
11. Використовуючи створений запит, додайте назви наступних країн:

- Росія
- Туреччина
- Індія
- Єгипет
- Навколоземна орбіта

12. Створіть запит з ім'ям «ДобавитиТранспорт». У режимі SQL введіть текст запиту:

```
INSERT INTO Транспорт  
(назва) VALUES ([Введіть вид транспорту])  
;
```

Запит додаватиме нові записи в таблицю «**Транспорт**», запрошуючи у користувача назва виду транспорту.



13. Використовуючи створений запит, додайте наступні назви видів транспорту:

- Літак
- Потяг
- Автобус
- Теплохід
- Собача упряжка
- Слон
- Велорикша
- Верблюд
- Космічний корабель

14. Відкрийте таблицю «**Тур**» в режимі конструктора і визначте властивості підстановки стовпців для полів «**країна**» і «**транспорт**», використовуючи дані таблиць «**Країна**» і «**Транспорт**» відповідно.

15. Відкрийте таблицю «**Тур**» в режимі таблиці і додайте не менше десять записів про тури в різні країни і на різних видах транспорту, задаючи різні значення для часу від'їзду і повернення, а також ціни туру. Кожному туру повинна бути привласнена назва (по можливості, романтична).

16. Введіть у таблицю «**Клієнт**» інформацію не менше ніж про десять клієнтів.

17. Відкрийте таблицю «**Замовлення**» в режимі конструктора і визначте властивості підстановки стовпців для полів «**тур**» і «**клієнт**», використовуючи назву туру з таблиці «**Тур**» і поле «**фіо**» з таблиці «**Клієнт**» відповідно.

18. Відкрийте таблицю «**Замовлення**» в режимі таблиці і додайте не менше п'ятнадцять записів про замовлення. Для кожного замовлення повинна бути вказана кількість квитків.

19. Самостійно створіть запит «**Суми по замовленнях**». Запит повинен виводити список усіх замовлень (назва туру, фіо клієнта) і для кожного із замовлень його вартість (твір ціни туру на кількість замовлених квитків).

20. Самостійно створіть запит «**Суми по турах**». Запит повинен виводити список усіх турів і для кожного туру – вартість всіх проданих на нього квитків (обчислюється як твір ціни туру на суму поля «кількість» всіх замовлень, пов'язаних з даним туром).

21. Створіть запит «**Суми замовлень по клієнтах**». Введіть в режимі SQL наступний код:

```
SELECT Клієнт.id, Клієнт.фіо, SUM(Вартість) AS Сума  
FROM Клієнт LEFT JOIN (  
SELECT Тур.цена*Заказ.количество AS Вартість  
Заказ.клієнт  
FROM Тур INNER JOIN Замовлення ON Тур.id = Заказ.тур  
) AS ТурЗаказ ON Клієнт.id=ТурЗаказ.клієнт  
GROUP Клієнт.id, Клієнт.фіо;
```

Запит формує список клієнтів і для кожного клієнта указує сумарну вартість всіх його замовлень

22. Самостійно створіть запит **самий «поважний клієнт»**, що виводить ім'я клієнта, який оформив замовлень на найбільшу суму.

23. Самостійно створіть запит «**Елітні тури**», що виводить список турів, ціна яких вище за середню ціну по всіх наявних турах.

Контрольні запитання

1. Що таке запит SQL?
2. Які оператори мови SQL ви знаєте?
3. Як створити запит за допомогою SQL?

Лабораторна робота №9

Проектування бази даних

Мета роботи:

Освоїти методи проектування баз даних. Пройти всі етапи проектування – від аналізу документів з наочної області до реалізації бази даних в MS Access 2000.

Вступ

Розробка бази даних. Вдала розробка бази даних забезпечує простоту її підтримки. Дані слід зберігати в таблицях, причому кожна таблиця повинна містити інформацію одного типу, наприклад, відомості про замовників. Тоді достатньо буде відновити конкретні дані, такі як адреса, тільки в одному місці, щоб оновлена інформація відображалася у всій базі даних.

Правильно спроектована база даних звичайно містить різноманітні запити, що дозволяють відображати потрібну інформацію. У запитах може виводитися підмножина даних, наприклад, перелік замовників з Петербурга, або комбіновані дані з декількох таблиць, наприклад, відомості про замовлення сумісно з відомостями про замовників.

Хід роботи

Легенда: Комітет з освіти Вашого міста проводить експеримент, в якому беруть участь декілька шкіл з різних районів міста. Метою експерименту є вивчення можливостей сучасних інформаційних технологій стосовно завдань моніторингу і централізованого управління навчальним процесом.

Всі школи, які беруть участь в експерименті, зв'язані між собою швидкодіючого комп'ютерною мережею. Вимагається створити базу даних, яка забезпечуватиме зберігання і обробку даних, що відносяться до навчального процесу.

Вам доручено виконати проектування структури бази даних і реалізацію пілотної версії з використанням СУБД MS Access.

Початкові документи:

Документ 1. Школи, які беруть участь в експерименті:

Район	Школа №	Адреса	Директор
Московський	Гімназія № 303	пр. Гагаріна, 14	Петров В.П.
Дзержинський	Гімназія № 116	вул. Культури, 1	Романова Л.Н.
Дзержинський	СШ № 131	вул. Данилевського, 21	Іванова О.Ю.
Дзержинський	СШ № 50	вул. Єсеніна, 4	Сидорова У.Г.
Київський	СШ № 100	вул. Шевченка, 52	Дьомін В.Ф.
Червонозаводський	СШ № 2	вул. Пожежна, 23	Войлар П.П.

Документ 2. Список класів школи № 519

Класний керівник	Клас	Кількість учнів
Супрін Ф.Ф.	10-г	23
Камитін О.Ж.	11-е	32
Крилова Д.Л.	7-же	33

Документ 3. Список учнів 10-а класу 519-ї школи (класний керівник Дуров Д.М.)

Прізвище	Ім'я	По батькові	Пол
Іванов	Олександр	Володимирович	чол.
Петров	Андрій	Юрійович	чол.
Александров	Сергій	Георгійович	чол.
Сидоров	Костянтин	Володимирович	чол.
Ларіна	Олександра	Михайлівна	жін.
Тріпелець	Володимир	Сергійович	м чол.

Документ 4. Предмети, що викладаються у школах

Предмет	Випускний іспит
Математика	Обов'язковий
Фізика	Обов'язковий
Хімія	-
Іноземна мова	За бажанням
Фізкультура	За бажанням
Література	Обов'язковий
Біологія	За бажанням
Географія	За бажанням

Документ 5. Викладання предметів у 10-а класі 519-ї школи

Предмет	Уроків за тиждень	Викладач	Річний іспит
Математика	3	Петров І.П.	+
Фізика	2	Романов Л.З.	+
Хімія	2	Іванова О.Ю.	+
Іноземна мова	2	Сидоров Н.Г.	+
Фізкультура	1	Депутат В.Ф.	+
Література	2	Ляпунова П.П.	-
Біологія	1	Семенов В.П.	-
Географія	2	Цапишев П.П.	-

Документ 6. Відомість. Річні оцінки з фізики учнів 10-го класу 519-ї школи. (Викладач Дуров Д.М.)

Учень	Оцінка	Примітка
Іванов А.Ф.	3	
Петров М.И.	4	
Александров Т.Б.	-	не атестований
Сидоров П.Р.	5	
Ларіна Е.Х.	4	

Послідовність дій:

1. Виконайте аналіз початкових документів, що складають опис предметної області. Виділіть характерні сутності.
2. Виявіть атрибути для кожної сутності з предметної області. Складіть графічне зображення сутностей і їх атрибутів у компактній формі. Приклад графічного представлення сутностей наведений на рисунку.



3. Виявіть зв'язки між сутностями, визначте типи зв'язків (один до одного, один до багатьох або багато хто до багатьох). Проаналізуйте властивості зв'язків і при необхідності введіть в модель додаткові сутності або об'єднайте існуючі.

Оскільки база даних будуватиметься з використанням реляційної моделі, всі зв'язки вигляду «багато хто до багатьох» слід перетворити в зв'язку «один до багатьох». Зв'язки вигляду «один до одного» у багатьох випадках можна усунути, об'єднавши сутності.

4. Складіть діаграму «Єство-зв'язок (ER-діаграму)», що представляє зв'язки (взаємостосунки) між різними сутностями. Якщо присутні «слабкі» сутності, вони повинні бути виділені особливо.

Приклад ER-діаграми наведений нижче. Діаграма відноситься до предметної області, яка близька до розглянутої.



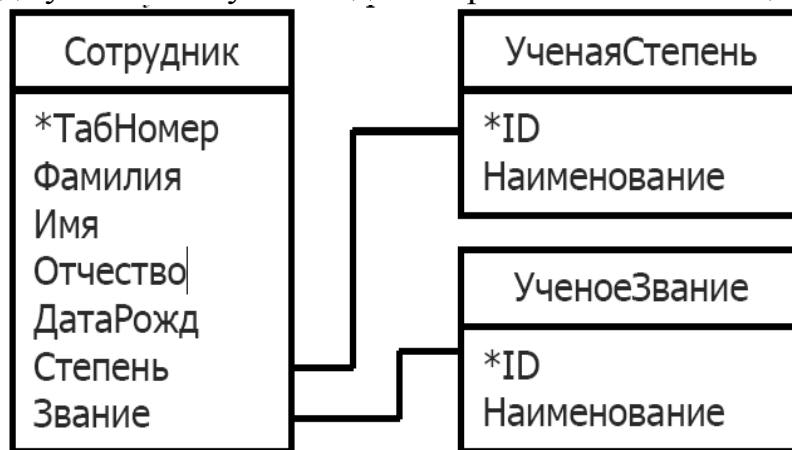
5. Для кожної сутності визначить спосіб унікальної ідентифікації екземплярів. Як ідентифікатор можуть бути використані вже існуючі атрибути, якщо для них дотримано вимогу унікальності і їх значення не схильні до змін. У багатьох випадках може виявитися більш переважним введення нового числового атрибуту як ідентифікатора. Для складових сутностей унікальний ідентифікатор повинен бути складовим і включати ідентифікатори сутностей, які пов'язані з даною.

Внесіть доповнення у графічне зображення сутностей і їх атрибутів, позначивши ідентифікатори сутностей. Приклад графічного представлення сутностей з ідентифікаторами наведений нижче.



6. Визначте домени для атрибутів сутностей. Якщо домен деякого атрибуту достатньо специфічний, може бути потрібно утворення нової сутності для зберігання значень даного домену.

У прикладі на рисунку нижче показано, як домени атрибутів «Ступінь» і «Звання» породжують нові сутності для зберігання значень з цих доменів.



7. На основі одержаної моделі «сутність - зв'язок» складіть реляційну модель, що описує таблиці і їх первинні ключі, які забезпечують унікальність рядків у таблицях. Визначте зовнішні ключі, що забезпечують зв'язки між таблицями.

8. Виконайте нормалізацію таблиць. Перевірте структуру кожної таблиці на відповідність першої, другої і третьої нормальним формам. Усуньте невідповідності.

9. Запустіть Access, створіть базу даних з ім'ям «Школи».

10. Реалізуйте створену раніше реляційну модель, склавши відповідні таблиці. Створіть всі необхідні зв'язки між первинними і зовнішніми ключами таблиць і визначте для зв'язків властивості забезпечення цілісності даних.

11. У тих таблицях, де це необхідно, визначте властивості підстановки значень.

12. Внесіть дані в таблиці. Повинно бути представлено не менше 3 шкіл, в кожній школі не менше 3 класів і в класі не менше 4 учнів.

14. Створіть об'єкти-запити, що формують набори даних, які відповідають початковим документам, використаним при проектуванні (список шкіл, список класів, список учнів, список предметів, предмети, що викладаються у вказаному класі, відомість класу по предмету). Запити можуть містити параметри.

15. Створіть звіт «Зведена відомість школи», що виводить список учнів усіх класів школи з їх річними оцінками з усіх предметів. Список групують за назвами предметів і номерами класів. Для зручності реалізації даного звіту може знадобитися створити запит, що формує набір усіх необхідних даних.

Контрольні запитання

1. Що називається базою даних?
2. Чим поле відрізняється від запису?
3. Які основні функції СУБД і що це таке?
4. Для чого служать кнопкові форми?

САМОСТІЙНА РОБОТА

За варіантами індивідуальних завдань (згідно з номером у журналі) складіть власну базу даних.

Варіанти індивідуальних завдань:

ВАРІАНТ 1

Фірма-постачальник добрив уклала угоди з певними замовниками. Фірма надає такі пільги: замовлення на 50 тонн і вище –90% вартості, на 100 тонн і вище-85%, на 200 тонн і вище-83%. Для реєстрації замовлень, збереження інформації про замовників, про асортимент добрив і обробки цієї інформації необхідно розробити базу даних, що складається з трьох таблиць.

“Замовлення”. Поля: код замовлення, дата заповнення, код замовника, назва добрива, код добрива, категорія пільг, площа для обробки, дата постачання.

“Добрива”. Поля: назва добрива, код добрива, виробник, норма використання на один гектар, вартість однієї тонни, термін зберігання.

“Господарства”. Поля: код замовника, назва господарства, область, район, індекс, телефон, голова господарства.

Завдання для побудови запитів:

знайти на певну дату всі замовлення постачання (дату ввести як параметр);

збільшити вартість добрива на10% для певного виробника;

для кожного замовника розрахувати необхідну кількість добрива, вартість цієї маси, суму до сплати (вартість замовлення-пільги+20%ПДВ);

знайти суму замовлень за кожен місяць поточного року;

які добрива не користуються попитом(замовлення на них відсутні протягом попереднього й поточного років)?

Завдання для створення форм і обробки даних:

розробити систему реєстрації замовлень з підрахуванням необхідної кількості добрив, вартості їх маси, суми до сплати (вартість замовлення-пільги+20% ПДВ). Треба передбачити можливість вибору добрив, що мають однакові назви, але відрізняються за іншими атрибутами. Треба також передбачити можливість оновлення таблиці «Добрива», коли термін зберігання певного виду добрива закінчився (видалення цих записів).

Завдання для побудови звітів:

кількість поставлених добрив за місяцями для кожної області з підрахунком відсоткового входження для кожної області до загальної суми поставлених добрив;

суму замовлень за кожен квартал поточного року за кожним господарством і загальну суму.

ВАРІАНТ 2

Для ведення підрахунків сплат абонентів телефонної мережі розробити базу даних “Сплати за телефон”, що містить три таблиці :

”Абоненти”, “Тарифи”, “Платежі”. Таблиця “Абоненти ” містить інформацію про абонентів телефонної мережі. Таблиця “Тарифи” відображає пільги, які надаються при сплаті абонентам деяких категорій(ветерани ВВВ, інваліди, чорнобильці і т. ін.). Таблиця “Платежі” – це журнал сплат абонентів, який повинен оновлюватись щомісячно. Вважаємо, що кількість хвилин розмов додається до значень “кількість несплачених хвилин ” першого числа наступного місяця .

Плата за міжміські й міжнародні розмови повинна вноситися до 15-го числа наступного місяця. Пеню в розмірі 1% від вартості розмови стягують, якщо сплачено не вчасно.

“Абоненти”. Поля: прізвище, ім’я та по батькові, № телефону, код тарифу, кількість несплачених хвилин за міські розмови, кількість несплачених хвилин за міжміські розмови, кількість несплачених хвилин за міжнародні розмови.

“Тарифи”. Поля: код тарифу, назва пільгової категорії, розмір сплати.

“Платежі”. Поля: № телефону, дата платежу, внесена сума за міські розмови, внесена сума за міжміські розмови, внесена сума за міжнародні розмови, розрахунковий рахунок.

Завдання для побудови запитів:

для кожного абонента на певну дату знайти загальну внесену суму;

для кожного абонента знайти дати платежів, суми до сплати, фактичні платежі, їх різницю й кількість несплачених хвилин за кожним видом розмов;

для певної пільгової категорії зменшити тариф на 10%;

для кожного абонента знайти суму всіх платежів за кожним видом розмов і їх загальну суму за поточний рік;

які абоненти зовсім не сплачували за телефон?

Завдання для створення форм і обробки даних:

розробити систему, що дає можливість реєструвати платежі, підраховувати суми сплат за кожен вид розмов, підраховувати загальну суму платежу. Надати можливість підраховувати заборгованість загальну й за кожним видом сплат. Передбачити можливість оновлення таблиці “Абоненти” наприкінці кожного місяця, вносячи дані у поля “ кількість несплачених хвилин” і наприкінці року “Платежі”, видаливши всі записи за попередній рік з паралельним внесенням кількості несплачених хвилин до таблиці “Абоненти”.

Завдання для побудови звітів:

для кожного абонента знайти суму всіх платежів за кожним видом розмов за кожен місяць і загальну суму за поточний рік;

підрахувати тривалість розмов кожного виду у хвилинах за місяцями, за рік, знайти загальну суму і частку (відсоток) кожного виду розмов у ній.

ВАРІАНТ 3

Туристична фірма розміщує гостей міста у готелях. Для отримання інформації про вільні місця у готелях міста й обробки інформації розробити базу даних “Розміщення гостей” з трьох таблиць: ”Готелі міста”, ”Замовлення”, ”Гості міста”. Фірма надає пільги при розміщенні для гостей з дітьми. При поселенні їх у дво- або тримісних номерах сплачують 80% від вартості проживання.

“Готелі міста”: включає інформацію про готелі міста. Поля: назва, код готелю, рівень сервісу (вибір зі списку від 1 до 5 зірок), адреса, район міста, кількість номерів люкс, кількість номерів одномісних, кількість номерів двомісних, кількість номерів тримісних, вартість проживання у номерах люкс, вартість проживання у номерах одномісних, вартість проживання у номерах двомісних, вартість проживання у номерах тримісних.

“Замовлення”: заповнюють на кожний замовлений номер. Поля: код замовлення, код готелю, код гостя, дата заселення, кількість днів, тип номера.

“Гості міста”: заповнюють при реєстрації гостей у туристичній фірмі. Поля: код, прізвище, ім’я, країна проживання, з дітьми або ні.

Завдання для побудови запитів:

для кожного району міста зайти готелі з певним рівнем сервісу (рівень вводиться як параметр);

для кожного гостя підрахувати загальну вартість проживання (вартість за одну добу*термін проживання*пільги+20%ПДВ);

для вказаного готелю збільшити вартість проживання на 5%;

знайти кількість вільних місць одного типу на певну дату по всіх готелях міста;

показати, які готелі не користуються попитом (не мають жодного замовлення за квартал)?

Завдання для створення форм і обробки даних:

розробити систему реєстрації гостей. Надати можливість вибору готелю (за районом міста, тарифами, рівнем сервісу) на певну дату. При реєстрації обчислити дату від’їзду, вартість проживання (вартість за одну добу*термін проживання*пільги+20%ПДВ)

Завдання для побудови звітів:

надрукуйте загальну кількість замовлень за поточний рік за районами міста і рівнем сервісу;

для кожного готелю покажіть суму оплати за проживання за місяцями.

ВАРІАНТ 4

Розробити базу даних для обслуговування потреб будівельного майданчику, що складається з таблиць “Матеріали”, ”Постачальники”, ”Замовлення”. Таблиця “Матеріали” містить дані про будівельні матеріали, що можуть бути поставлені на майданчик. Передбачити, що матеріали можуть мати однакові назви, але відрізняються за іншими атрибутами. Поля: назва матеріалу,

код матеріалу, виробник, вартість одиниці продукції, мінімальна партія, термін зберігання, код постачальника.

Таблиця «**Замовлення**» містить дані про будівельні матеріали, що необхідно поставити на будівельний майданчик. Поля: код замовлення, дата заповнення, назва матеріалу, код матеріалу, кількість, одиниця виміру, дата постачання, на яку потрібно отримати матеріали.

Таблиця «Постачальники» містить дані про постачальників будівельних матеріалів, з якими укладені угоди. Поля: код постачальника, назва постачальника, адреса, телефон, прізвище начальника.

Завдання для побудови запитів:

знайдіть всі замовлення на певну дату постачання (ввести як параметр);

для певного виробника збільшити вартість продукції на 15%;

для кожного замовлення розрахувати вартість і суму до сплати (вартість замовлення +20%ПДВ);

знайти суму замовлень для кожного постачальника на певну дату;

які матеріали зовсім не використовувались під час будівництва (замовлення на них відсутні)?

Завдання для створення форм і обробки даних:

розробити систему, що реєструє замовлення з підрахуванням їх вартості, суми до сплати (вартість замовлення+20%ПДВ). Передбачити можливість вибору матеріалу з тих, що мають однакові назви, але відрізняються за іншими атрибутами. Передбачити можливість оновлення таблиці “Матеріали”, коли термін зберігання певної продукції закінчився (видалити ці записи).

Завдання для побудови звітів:

для кожного з матеріалів показати кількість постачань, загальну вартість з підрахунком відсотку кожного з матеріалів у загальній сумі постачань;

показати суму замовлень для кожного постачальника на певну дату.

ВАРІАНТ 5

Для ведення підрахунків сплат за використану електроенергію споживачами розробити базу даних “Сплати за електроенергію”, що містить три таблиці: “Споживачі”, “Тарифи”, “Платежі”. Таблиця “Споживачі” містить інформацію про споживачів електроенергії. Таблиця “Тарифи” відображає пільги, що надаються при сплаті споживачами деяких категорій(ветерани ВВВ, інваліди, чорнобильці та т. ін.). Таблиця “Платежі” – це журнал сплат споживачів, який повинен оновлюватись щорічно. Вважаємо, що кількість кВт/год вноситься на початку наступного року.

Таблиця «**Споживачі**» має поля: прізвище, ім'я та по батькові, код споживача, код тарифу, показання лічильника на кінець попереднього року, заборгованість за попередній рік.

Таблиця “**Тарифи**” має поля: код тарифу, назва пільгової категорії, розмір сплати.

Таблиця “**Платежі**” має поля: код споживача, дата платежу, внесена сума, розрахунковий рахунок, попередній показник лічильника, теперішній показник лічильника.

Завдання для побудови запитів:

для кожного споживача знайти всі сплати за поточний рік і кількість кВт/год, що сплачують;

для певної пільгової категорії зменшити тариф на 10%;

знайти для кожного споживача дати платежів, суми до сплати, фактичні платежі, різницю між ними;

для кожного споживача підрахувати заборгованість за сплатами протягом року;

які споживачі зовсім не сплачували за електроенергію?

Завдання для створення форм і обробки даних:

розробити систему, що дає можливість реєструвати платежі, підраховувати суми сплат, їх підсумок за поточний рік, заборгованість на дату перевірки.

Необхідно надати можливість підраховувати заборгованість за попередній рік.

Необхідно також передбачити можливість оновлення таблиці “Споживачі” й “Платежі” наприкінці року. Для таблиці “Споживачі” підсумувати заборгованість кожного споживача (з урахуванням і тих, що зовсім не сплачували за спожиту електроенергію, вважаючи, що показання лічильника на кінець року відомі). До таблиці “Платежі” занести записи за кожним споживачем за датою 1 січня, нульовою сумою внеску й показанням лічильника на кінець попереднього року.

Завдання для побудови звітів:

для кожного споживача знайти суму всіх платежів;

для кожного споживача показати заборгованість й загальну суму боргу всіх споживачів за місяцями, їх відсоток (частку) у загальній сумі.

ВАРІАНТ 6

Міський молокозавод виконує замовлення магазинів на постачання продукції. Для реєстрації замовлень, збереження інформації про замовників, про асортимент продукції, що виробляється, та організації поставок необхідно розробити базу даних з трьох таблиць.

“**Замовлення магазинів**”. Поля: код замовлення, дата заповнення, код магазину-замовника, назва продукції, код продукції, обсяг замовленої партії, обсяг поставленої партії, дата постачання.

“**Продукція**”. Поля: назва продукції, жирність, код продукції, вартість одиниці продукції, мінімальна партія, термін зберігання.

“**Магазини**”. Поля: код магазину-замовника, назва магазину, адреса, район міста, телефон, директор магазину.

Завдання для побудови запитів:

на певну дату постачання (яка може бути введена як параметр) показати замовлення для кожного магазину;

для продукції певної назви збільшити вартість на 10%;

на кожну дату для кожного замовника і його замовлень показати вартість замовленої партії продукції, суму до сплати (вартість замовлення + 20% ПДВ);

знайти суму замовлень за останній квартал поточного року за кожним магазином;

знайти продукцію, що не користуються попитом (замовлення на неї відсутні протягом останнього місяця)?

Завдання для створення форм й обробки даних:

розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовленої продукції, суми до сплати (вартість замовлення + 20% ПДВ), обсягу продукції, що необхідно допоставити, її вартості. Передбачити можливість вибору продукції, що має однакові назви, але відрізняється за іншими атрибутами. Забезпечити можливість оновлення таблиці «Продукція», коли термін зберігання певного виду продукту закінчився (видалити такі записи).

Завдання для побудови звітів:

показати кількість поставленої продукції за місяцями по кожному району міста й загальну суму;

показати обсяг продукції, що була недопоставлена за кожен місяць поточного року, підрахувати відсоток входження кожного виду продукції до загальної суми.

ВАРІАНТ 7

Автомобільний завод випускає різні моделі автомобілів. Для забезпечення потреб складального цеху автозаводу розробити базу даних з трьох таблиць: “Вузли”, “Постачальники”, “Замовлення”.

Таблиця “**Вузли**” містить дані про вузли автомобіля, що поставляються до складального цеху. Необхідно передбачити, що вузли можуть мати однакові назви, але відрізнятися за іншими атрибутами. Вона має поля: назва вузла, код вузла, виробник, вартість одиниці продукції, мінімальна партія, код постачальника (вибір зі списку).

Таблиця “**Замовлення**” містить дані про вузли автомобіля, які необхідно поставити до цеху. Поля: код замовлення, дата заповнення, назва вузла, код вузла, кількість, дата постачання, на яку потрібно отримати замовлення.

Таблиця “**Постачальники**” містить дані про постачальників автомобільних вузлів, з якими укладені угоди. Поля: код постачальника, назва постачальника, адреса, телефон, прізвище начальника.

Завдання для побудови запитів:

на певну дату постачання (ввести як параметр) показати всі замовлення;

для певного виробника збільшити тариф на 10%;

для кожного замовлення розрахувати вартість й суму до сплати (вартість замовлення +20%ПДВ);

знайти суму замовлень для кожного постачальника на певну дату;
які вузли зовсім не використовувались за попередній квартал?

Завдання для створення форм й обробки даних:

розробити систему, що реєструє замовлення, підраховує вартість замовлення, суму до сплати (вартість замовлення+20%ПДВ). Передбачити можливість вибору вузла відповідно до виробника.

Завдання для побудови звітів:

для кожного з вузлів й для кожного постачальника зайти кількість поставок, загальну вартість;

знайти суму замовлень для кожного постачальника на певну дату.

ВАРІАНТ 8

Для автоматизації роботи в касах аеропорту розробити базу даних “Продаж авіаквитків”, що має три таблиці: “Авіалайнери”, “Рейси”, “Продаж”. При сплаті за квитки діють такі пільги: ветеранам війн-20% від вартості квитка; дітям-50%; працівникам авіасервісу – безкоштовно(100%).

Таблиця “Авіалайнери” містить інформацію про літаки. Вона має поля: бортовий номер; тип літаку, дата останнього техогляду, кількість місць бізнес-класу, вартість квитків у цьому салоні, кількість місць салону першого класу, вартість квитків у цьому салоні, кількість місць другого класу, вартість квитків у цьому салоні.

Таблиця “Рейси” відображає інформацію про рейси аеропорту. Вона має поля: № рейсу, бортовий номер, дні вильоту(кожен день, по парним чи непарним числам), пункт вильоту, пункт призначення, пункти поміж ними.

Таблиця “Продаж” – це журнал продажу квитків у касі. Вона має поля: № рейсу, дата продажу, тип салону(вибір зі списку), кількість квитків, категорія пільг, дата вильоту (необхідно перевіряти за полем “Дні вильоту” з попередньої таблиці).

Завдання для побудови запитів:

показати усі рейси до певного пункту (ввести як параметр);

для певного типу літака збільшити тариф на 10%;

для кожного зареєстрованого продажу авіаквитків показати номер рейсу, дату продажу, тип салону, вартість квитків у цьому салоні, кількість квитків, категорію пільг, дату вильоту, підрахувати суму до сплати (вартість квитка* кількість квитків*пільги +20%ПДВ) ;

для одного з рейсів за певною датою підрахувати кількість проданих квитків, їх загальну вартість для кожного типу салону;

який рейс не користується попитом(квитків на нього продавались впродовж трьох діб до вильоту)?

Завдання для створення форм й обробки даних:

розробити систему, що реєструє у касах аеропорту продаж авіаквитків з підрахуванням суми до сплати (вартість квитка* кількість квитків*пільги

+20%ПДВ), загальної вартості квитків. Перевірити за полем “Дні вильоту” з таблиці “Рейси” поле “Дата вильоту”. Передбачити можливість отримання довідки про кожен рейс з підрахуванням кількості вільних місць у кожному салоні. Довідка про рейс: № рейсу, бортовий номер, дні вильоту(кожен день, по парним чи непарним числам), пункт вильоту, пункт призначення, пункти поміж ними, кількість місць бізнес-класу, вартість квитків у цьому салоні, кількість місць салону першого класу, вартість квитків у цьому салоні, кількість місць другого класу, вартість квитків у цьому салоні.

Завдання для побудови звітів:

знайти загальну вартість проданих квитків за кожен місяць з початку поточного року;

для кожного рейсу показати загальну кількість проданих квитків з підрахунком їх відсотку (частки) в загальній кількості проданих квитків на всі рейси.

ВАРІАНТ 9

Розробити базу даних “ Комерційна лікарня”, що складається з трьох таблиць “Відділення”, “Лікарі”, “Пацієнти”.

Таблиця “Відділення” має поля: назва відділення, код відділення, прізвище завідуючого, телефон, вартість лікування за добу у відділенні, кількість місць у відділенні.

Таблиця “Лікарі” має поля: код лікаря, прізвище, ім’я та по батькові, дата народження, категорія, стать, код відділення, де він працює, домашній телефон.

Таблиця “Пацієнти” має поля: прізвище, ім’я та по батькові, дата народження, стать, категорія пацієнта, дата надходження у лікарню, термін лікування, код лікаря, що спостерігає.

Завдання для побудови запитів:

для кожного лікаря показати список пацієнтів, які перебували під його доглядом на певний період(ввести як параметр);

для певного відділення збільшити вартість лікування за добу на 5%;

для кожного пацієнта знайти вартість лікування й сума до сплати;

обчислити суму премії для кожного лікаря (по місяцях);

який лікар не займався лікувальною практикою протягом поточного року?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації пацієнтів, що надходять до лікарні, їх розміщення по відділенням, а також для збереження інформації про лікарів цієї лікарні. Система мусить обчислювати вартість лікування, суму до сплати й премію лікаря. Розрахунки робити за такими правилами. Усіх пацієнти поділяють на категорії:

перша – пільгова, пацієнтів обслуговують безкоштовно (ветерани війн, діти до 7 років, чорнобильці);

друга – 50% сплати (пенсіонери, діти до 16 років, інваліди);

третя – повна сплата.

Лікарі отримують такі премії: 10% від суми до 300 грн., 5% - до 1000 грн., 2,5% - від суми понад 1000 грн. (від наданого обсягу медичних послуг).

Завдання для побудови звітів:

для кожного відділення показати суму, що сплачена пацієнтами минулого року, порахувати її відсоток у загальному обсязі наданих послуг, результат показати за місяцями;

знайти загальну суму премій для кожного лікаря за поточний рік.

ВАРІАНТ 10

Видавництво виконує замовлення магазинів на поставку книг для продажу. Від кількості замовленої партії книг залежить знижка, що надає видавництво: починаючи з 1000 примірників і більше - 30% від вартості, починаючи з 2000 примірників - 35%, понад 3000 примірників - 40%. Для реєстрації замовлень, збереження інформації про замовників, про книжкові видання та організацію поставок розробить базу даних, яка складається з трьох таблиць.

Таблиця “Замовлення магазинів”. Має такі поля: «№ замовлення», «дата заповнення», «код магазину-замовника», «код книги», «замовлена кількість примірників», «відправлена кількість», «дата постачання».

Таблиця “Книжкові видання”. Має такі поля: «код книги», «автор», «назва книги», «тираж», «об’єм сторінок», «відпускна ціна».

Таблиця “Магазини”. Має поля: «код замовника», «назва магазину», «адреса», «район міста», «телефон», «директор магазину».

Завдання для побудови запитів:

знайти на певну дату постачання (ввести як параметр) всі замовлення для кожного магазину;

зменшити відпускну ціну книг певного автора на 5%;

на кожну дату по замовниках знайти вартість замовленої партії продукції, суму до сплати (вартість замовлення * пільги + 20% ПДВ);

знайти суму замовлень за останній квартал поточного року за кожним магазином;

яка продукція не користуються попитом(замовлення на неї відсутні протягом попереднього і поточного місяцю)?

Завдання для створення форм й обробки даних:

розробити систему, що дозволяє реєструвати замовлення, підраховувати вартість замовленої партії, суму до сплати (вартість замовлення +20% ПДВ), обсяг партії книжок, що необхідно допоставити, їх вартість.

Завдання для побудови звітів:

кількість поставлених книг за місяцями для кожного району міста й загальну суму;

обсяг книжкової партії, що була недопоставлена за кожен місяць поточного року.

ВАРІАНТ 11

Розробити довідкову систему щодо асортименту продукції, яка виробляється на промисловому підприємстві різними цехами й перевозиться до складу. При передачі партії продукції до цеху її реєструють. Продукцію можуть виробляти різні цехи цього підприємства. Створіть базу даних, що має три таблиці “Продукція”, “Цех”, “Партія продукції”.

Таблиця “Продукція” містить дані про види продукції, що виробляються підприємством. Передбачити, що продукція може бути різного гатунку. Вона має поля: назва продукції, вартість одиниці продукції вищого гатунку, вартість одиниці продукції першого гатунку, вартість одиниці продукції другого гатунку, мінімальна партія, термін зберігання.

Таблиця “Цех” містить дані про цехи підприємства. Таблиця має поля: № цеха, назва, прізвище начальника цеха, телефон.

Таблиця “Партія продукції” містить дані про партію продукції, що розміщується на складі. Має поля: код партії, назва продукції, № цеха, показник якості, обсяг партії продукції, дата реєстрації.

Завдання для побудови запитів

показати довідку про певний вид продукції (в яких цехах виробляється, якого гатунку, найдавніша і остання партія, що є на складі);

для певної продукції збільшити вартість одиниці продукції вищого гатунку на 5%;

для кожної партії продукції знайти її загальну вартість, кінцеву дату зберігання;

для кожного цеху за кожен місяць знайдіть загальну вартість виробленої продукції, що є на складі;

яка продукція відсутня на складі?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації продукції, що розміщується на складі, підраховувати її загальну вартість, кінцеву дату зберігання. Надати можливість отримувати довідку про певний вид продукції (у яких цехах виробляється, якого гатунку, найдавніша й остання партія, яку зберігають на складі). Передбачте можливість оновлення таблиці “Партія продукції”, коли термін зберігання продукції скінчився (видалити відповідні записи).

Завдання для побудови звітів:

для кожного цеху за кожен місяць знайти загальну вартість виробленої продукції, що є на складі;

зайти кількість продукції кожного виду (за гатунками), що є на складі.

ВАРІАНТ 12

Розробити базу даних “Комерційна лікарня”, що складається з трьох таблиць “Тарифи”, “Хірурги”, “Пацієнти”.

Таблиця “Тарифи”. Має такі поля: категорія операції, назва, вартість, приблизний термін після операційної реабілітації, вартість однієї доби у цей період.

Таблиця “Хірурги” складається з полів: код хірурга, прізвище, ім’я та по батькові, дата народження, категорія, стать, домашній телефон.

Таблиця “Пацієнти” має поля: прізвище, ім’я та по батькові, дата народження, стать, категорія пацієнта, дата операції, код хірурга, категорія операції, фактичний термін післяопераційної реабілітації.

Завдання для побудови запитів

для кожного лікаря знайти список операцій на певну дату (ввести як параметр);

для певної категорії операцій зменшити тариф на 10%;

для кожного пацієнта знайти вартість операції, загальну суму лікування й сума до сплати;

обчислити щомісячну суму премії для кожного хірурга;

який хірург не займався лікувальною практикою протягом місяця?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації пацієнтів в комерційних хірургічних лікарнях, збереження інформації про хірургів цієї лікарні, діючих тарифах. Обчислити попередню вартість лікування, скориговану вартість лікування на дату виписки, суму до сплати й премію хірурга. Розрахунки робити за такими правилами. Пацієнти поділяються на категорії:

перша - пільгова, пацієнти обслуговуються безкоштовно(ветерани війн, діти до 7 років, чорнобильці);

друга – 50% сплати(пенсіонери, діти до 16 років, інваліди);

третя – повна сплата.

Хірурги отримують такі премії за проведені операції: 15% від суми до 300 грн., 10% - до 1000 грн., 5% - понад 1000 грн..

Завдання для побудови звітів:

для кожного хірурга показати загальну суму, що сплачена пацієнтами за рік, підрахувати її відсоток (частку) у загальній вартості операцій;

знайти загальну суму премій для кожного хірурга (за місяцями і за поточний рік).

ВАРІАНТ 13

Розробити базу даних “Комерційна ветеринарна лікарня”, що складається з трьох таблиць “Тарифи”, “Лікарі”, “Журнал відвідувань”.

Таблиця “Тарифи” має поля: код захворювання, назва захворювання, ступень важкості(наприклад, початкова, середньої важкості, тяжка форма, дуже тяжка), вартість лікування.

Таблиця “Лікарі” має поля: код лікаря, прізвище, ім’я та по батькові, категорія, стать, домашній телефон .

Таблиця “Журнал відвідувань ”. Тварина може мати декілька діагнозів, за якими може лікуватися. У цьому випадку для неї створюють необхідну кількість окремих записів за однією датою звернення. Таблиця має поля: прізвище господаря, тип тварини, порода, прізвисько, вік тварини, стать тварини, дата відвідування, код лікаря, код захворювання .

Завдання для побудови запитів

для певного господаря тварини (прізвище вводити як параметр) знайти дату першого відвідування на рік;

для певного захворювання для всіх його ступенів підвищити вартість лікування на 5%;

для кожного відвідування показати загальну суму лікування за однією датою звернення й суму до сплати;

зайти кількість відвідувань одного господаря тварини за місяць;

який лікар не займався лікувальною практикою протягом місяця?

Завдання для створення форм й обробки даних:

розробити систему для використання у комерційних ветеринарних лікарнях для реєстрації тварин, збереження інформації про лікарів цієї лікарні, тарифи. Обчислювати вартість відвідування, суму до сплати й відрахування преміальних лікарю. Розрахунки робити за такими правилами. При першому відвідуванні на рік господаря тварин він сплачує внесок у розмірі 5 гривень. Під час третього відвідування за місяць одного господаря (можливо, з різними тваринами) сума до сплати зменшується на 10%. Лікарі отримують такі премії: 20% від суми до 300 грн., 15% - до 1000 грн., 12% - вище 1000 грн.

Завдання для побудови звітів:

для кожного лікаря зайти суму, що сплачена пацієнтами по місяцях за попередній рік з підрахунком відсотку у загальній сумі;

знайти сума премій для кожного лікаря за поточний рік.

ВАРІАНТ 14

Для ведення підрахунків сплат за використаний споживачами газ розробити базу даних “Сплати за газ ” з трьох таблиць :”Споживачі”, “Тарифи”, “Платежі”. Таблиця “ Споживачі ” містить інформацію про споживачів газу. Таблиця “Тарифи” відображає пільги, які надаються при сплаті споживачами деяких категорій(ветерани ВВВ, інваліди, чорнобильці та т. і.). Таблиця “Платежі” – це журнал сплат споживачів, який повинен оновлюватись щорічно. Вважаємо, що кількість газу вноситься на початку наступного року.

Таблиця “Споживачі” має поля: прізвище, ім’я та по батькові, код споживача, код тарифу, показання лічильника на кінець попереднього року, абортваність за попередній рік.

Таблиця “Тарифи” має поля: код тарифу, назва пільгової категорії, розмір сплати.

Таблиця “Платежі” має поля: код споживача, дата платежу, внесена сума, розрахунковий рахунок, попередній показник лічильника, теперішній показник лічильника.

Завдання для побудови запитів:

для кожного споживача знайти всі сплати за поточний рік, кількість газу, що сплачена;

для певної пільгової категорії операцій зменшити тариф на 10%;

для кожного споживача знайти дати платежів, суми до сплати, фактичні платежі, їх різницю;

знайти суму сплат по місяцях і за рік;

які споживачі зовсім не сплачували за газ?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації платежів, підраховувати суми сплат, їх обсяг за поточний рік і заборгованість на дату перевірки. Підрахувати заборгованість за попередній рік. Передбачити можливість оновлення таблиці “Споживачі” і “Платежі” наприкінці року. Для таблиці “Споживачі” контролювати заборгованість кожного споживача (з урахуванням і тих, що зовсім не сплачували за спожитий газ, вважаємо, що показання лічильника на кінець року відомі). До таблиці “ Платежі ” занести записи по кожному споживачу за датою 1 січня, нульовою сумою внеску і показниками лічильника на кінець попереднього року.

Завдання для побудови звітів:

усі платежі за місяцями;

для кожного споживача знайти заборгованість, відсоток до загальної суми заборгованості усіх споживачів.

ВАРІАНТ 15

Фірма-постачальник медикаментів аптекам міста уклала угоди з певними замовниками. Фірма надає такі пільги: замовлення на суму понад 1000 грн. – 5% знижки (від початкової вартості), на суму понад 10000 грн. - 11%, на суму понад 100000 грн. 25%. Для реєстрації замовлень, збереження інформації про замовників, про асортимент медикаментів і обробки цієї інформації треба розробити базу даних з трьох таблиць.

Таблиця “Замовлення” має поля: код замовлення, дата заповнення, код замовника, назва медикаментів, код медикаментів, об’єм замовленої партії, об’єм поставленої партії, категорія пільг, дата постачання.

Таблиця “Медикаменти» має поля: назва медикаментів, код медикаментів, виробник, вартість одиниці продукції, мінімальна партія, термін зберігання.

Таблиця “Аптеки” має поля: код замовника, номер аптеки, адреса, район міста, телефон.

Завдання для побудови запитів:

знайти на певну дату постачання (ввести як параметр) всі замовлення для кожної аптеки;

для певного виробника збільшити вартість одиниці продукції на 10%;

показати на кожну дату для кожної аптеки замовлення, вартість замовленої партії продукції, суму до сплати (вартість замовлення – пільги +20%ПДВ);
знайти суму замовлень за останній квартал поточного року по кожній аптеці;

яка продукція не користуються попитом (замовлення на неї відсутні протягом попереднього й поточного кварталу)?

Завдання для створення форм й обробки даних:

розробити систему, що реєструє замовлення з підрахуванням вартості замовленої партії продукції, суми до сплати (вартість замовлення - пільги +20%ПДВ), об'єму продукції, що необхідно допоставити, її вартості. Передбачити можливість вибору продукції, що має однакові назви, але відрізняється за іншими атрибутами. Передбачити можливість оновлення таблиці «Медикаменти», коли термін зберігання певного виду продукції закінчився (видалити такі записи).

Завдання для побудови звітів:

знайти суму постачання продукції за місяцями для кожної аптеки міста, загальну суму постачання;

знайти обсяг продукції, що була недопоставлена за кожний місяць поточного року, підрахувати відсоток недопостачання кожного виду продукції в загальній суми.

ВАРІАНТ 16

Для аналізу використання сільськогосподарських угодій області розробити базу даних, що містить три таблиці “Господарства”, ”Культури”, ”Поставлено за держзамовленням”.

Таблиця “Господарства” має такі поля: код господарства, назва господарства, район, телефон, голова господарства, посівна площа, загальна площа.

Таблиця “Культури” має такі поля: код культури, назва, середня врожайність, ціна за продукцію першого ґатунку, ціна за продукцію другого ґатунку, ціна за продукцію вищого ґатунку.

Таблиця “Поставлено за держзамовленням” має такі поля: код господарства, код культури, маса замовлення, маса, що поставлена, врожайність по господарству, ґатунок продукції(обирати зі списку), дата постачання.

Завдання для побудови запитів:

знайти для кожної культури усі господарства, що її вирощують, врожайність по господарству і ґатунок поставленої продукції;

підвищити вартість певної продукції за всіма ґатунками на 10%;

для аналізу роботи господарства показати для кожного з них усі культури, що вони постачають, різницю між врожайністю середньою і по господарству, вартість поставленої продукції;

для кожної культури знайти загальну кількість постачань і за кожним ґатунком на кінець року і їх вартість;

яку культуру зовсім не поставили за держзамовленням?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації постачання, обчислення врожайності, вартості поставленої продукції тощо. Система має підраховувати суму, яку сплачує держава з урахуванням наданих пільг (якщо врожайність по господарству вища на 15% від середньої, продукція вищого гатунку і держзамовлення виконано повністю, то держава доплачує 5% від вартості поставленої продукції). Передбачити можливість отримання довідки про культуру з підрахуванням загальної кількості постачань за кожним гатунком і їх вартість на дату запиту.

Завдання для побудови звітів:

знайти для кожної культури загальну кількість постачань за кожним гатунком на кінець року і їх вартість;

для кожного району області за кожною культурою знайти загальний об'єм постачань, їх вартість та кількість недопоставленої продукції.

ВАРІАНТ 17

Розробити базу даних “ Комерційна стоматологічна лікарня”, що складається з трьох таблиць “Тарифи ”, “Лікарі”, “Пацієнти”.

Таблиця “Тарифи ” має поля: код захворювання або виду роботи, назва, ступінь важкості(наприклад, початкова, середньої важкості, тяжка форма, дуже тяжка), вартість.

Таблиця “Лікарі” має поля: код лікаря, прізвище, ім'я та по батькові, дата народження, категорія, стать, домашній телефон .

Таблиця “Пацієнти” будується з огляду на таке. Певний пацієнт може мати декілька діагнозів, по яким може лікуватися. У цьому випадку для нього створюють необхідну кількість окремих записів з однією датою звернення. Таблиця має поля: прізвище, ім'я та по батькові, дата народження, стать, категорія пацієнта, дата звернення, код лікаря, код захворювання або виду роботи.

Завдання для побудови запитів:

для кожного лікаря показати список пацієнтів, що зверталися до нього на певний період (ввести як параметр, наприклад, місяць і т.ін.);

для певного захворювання для всіх його ступенів підвищити вартість лікування на 5%;

для кожного пацієнта знайти загальну суму лікування за однією датою звернення і суму до сплати;

обчислити суму премії для кожного лікаря (щомісячно);

який лікар не займався лікувальною практикою протягом місяця?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації пацієнтів у комерційних стоматологічних лікарнях, збереження інформації про лікарів цієї лікарні та про тарифи. Надати можливість обчислювати вартість лікування, суму до сплати та премію лікаря. Розрахунки робити за такими правилами. Усіх пацієнтів поділяють на категорії:

перша - пільгова, пацієнти обслуговуються безкоштовно (ветерани війн, діти до 7 років, чорнобильці);

друга – 50% сплати (пенсіонери, діти до 16 років, інваліди);

третя – повна сплата.

Лікарі отримують такі премії: 10% від суми до 300 грн., 5% - до 1000 грн., 2,5% - вище 1000 грн.

Завдання для побудови звітів:

для кожного лікаря знайти суму, що сплачена пацієнтами, помісячно за рік, підрахувати її відсоток відносно загальної суми;

знайти загальну суму премій для кожного лікаря за поточний рік.

ВАРІАНТ 18

Для автоматизації роботи в касах залізниці розробити базу даних “Продаж квитків ”, що містить таблиці: ”Поїзда”, “Рейси”, “Продаж”, “Тарифи”. При сплаті за квитки надають такі пільги: ветеранам війни - 20% від вартості квитка; дітям - 50%; працівникам авіасервісу – безкоштовно (100%).

Таблиця ”Поїзда” має поля: номер поїзда, дні відправлення(кожен день, по парним чи непарним числам), пункт відправлення, пункт призначення, пункт поміж ними 1, пункт поміж ними 2

Таблиця “Рейси” відображає інформацію про рейси вокзалу. Рейс одностайно визначається номером поїзду і датою відправлення. Таблиця має поля: номер поїзду, дата відправлення, кількість плацкартних вагонів, кількість купейних вагонів, кількість СВ.

Таблиця “ Продаж ” – це журнал продажу квитків у касі. Вона має поля: номер поїзду, дата відправлення, дата продажу, тип вагону(вибір зі списку), кількість квитків, категорія пільг.

Таблиця ”Тарифи” містить інформацію про вартість квитків для певного поїзда. Вона має поля: номер поїзду, вартість квитків у плацкартному вагоні, , вартість квитків для купейного вагона, вартість квитків для СВ.

Завдання для побудови запитів:

знайти всі рейси до певного пункту (пункт ввести як параметр);

для певного номера поїзда підвищити вартість квитків по СВ на 20%;

для кожного зареєстрованого продажу квитків знайти номер поїзда, дата відправлення, дата продажу, тип вагона (вибір зі списку), вартість квитків у цьому вагоні, кількість квитків, категорія пільг. Підрахувати суму до сплати (вартість квитка* кількість квитків*пільги +20%ПДВ), загальну вартість квитків;

який рейс не користується попитом(продажу квитків на нього не відбувалось протягом трьох діб)?

Завдання для створення форм й обробки даних:

розробити систему, що дозволяє реєструвати в касах залізниці продаж квитків з підрахуванням суми до сплати (вартість квитка* кількість квитків*пільги +20%ПДВ), загальної вартості квитків. Необхідно також передбачити можливість отримання довідки про кожен рейс з підрахуванням

кількості місць, що залишилися, за кожним видом вагонів. Довідка про рейс: номер поїзда, дата відправлення, кількість плацкартних вагонів, вартість квитків у цьому вагоні, кількість купейних вагонів, вартість квитків у цьому вагоні, кількість СВ, вартість квитків у цьому вагоні.

Завдання для побудови звітів:

показати загальну вартість проданих квитків за кожен місяць з початку поточного року;

знайти для кожного рейсу кількість проданих квитків з підрахунком відсотку входження відносно до загальної кількості проданих квитків по всіх рейсах.

ВАРІАНТ 19

Фірма-постачальник автомобілів приймає замовлення на поставку автомобілів різних виробників і різних моделей автомобілів однієї марки. Фірма надає такі пільги: замовлення на 5 автомобілів –90% вартості, для постійних клієнтів (звертались більш ніж два рази на рік) надається знижка 15%. Для реєстрації замовлень, збереження інформації про замовників, асортимент автомобілів розробити базу даних з трьох таблиць.

Таблиця “Замовлення” має поля: код замовлення, дата заповнення, код клієнта, код моделі, кількість, дата постачання.

Таблиця “Автомобілі” має поля: код моделі, назва марки, назва моделі, фірма-виробник, вартість моделі, об’єм двигуна, потужність двигуна, кількість пального на 100 км, тип двигуна(дизель чи карбюратор виводити списком).

Таблиця “Клієнти” має поля: код замовника, назва чи ім’я, адреса, телефон, розрахунковий рахунок.

Завдання для побудови запитів:

перевірити, чи певний клієнт може користуватися пільгами, якщо так, то якими(наприклад за категоріями) ;

зменшити вартість автомобілів певного виробника;

для кожного замовлення знайти вартість замовленої партії продукції, суму до сплати (вартість замовлення - пільги+20%ПДВ);

знайти суму замовлень за останній квартал поточного року по кожній моделі;

яка модель не користується попитом (замовлення на неї відсутні на протязі попереднього і поточного кварталу)?

Завдання для створення форм й обробки даних:

розробити систему, що дозволяє реєструвати замовлення з підрахуванням вартості замовлених автомобілів, суми до сплати (вартість замовлення - пільги +20%ПДВ);

Завдання для побудови звітів:

знайти кількість замовлень для кожної моделі на кожний квартал і загальну сплачену суму;

знайти для кожного клієнта замовлення за попередній та поточний роки.

ВАРІАНТ 20

Фірма надає транспортні послуги. Вона має можливість використовувати різні види транспорту, інформацію про які зберігають окремо. При реєстрації замовлення фіксують дані про замовників й вимоги до перевезення. При повторних зверненнях інформація про замовника не повторюється, але може оновлюватися, якщо були зміни. На перевезення існують пільги: при дальності 100-999 км діє знижка 5%, при дальності 1000-4999 – 10%, дальності понад 5000 – 15%. Для фіксації, зберігання і обробки цієї інформації розробити базу даних з трьох таблиць: "Замовлення на перевезення", "Транспорт і його вартість", "Замовники".

Таблиця "Замовлення на перевезення" має поля: № перевезення, дата перевезення, відстань, маса, накладні витрати, код транспортного засобу, код замовника.

Таблиця "Транспорт і його вартість" має поля: назва, код транспортного засобу, вартість т/км, об'єм вантажу, маса вантажу.

Таблиця "Замовники" має поля: код замовника, назва чи ім'я замовника, адреса, телефон, розрахунковий рахунок.

Завдання для побудови запитів:

знайти види транспорту для певної ваги чи певного об'єму (ввести, як параметр);

для певного замовника збільшити відстань на 20 км;

вибрати замовлення з підрахуванням вартості перевезення і суми до сплати (відстань* вартість т/км* маса+ накладні витрати - пільги);

знайти загальну вартість перевезень за останній місяць;

який вид транспорту зовсім не замовлявся?

Завдання для створення форм й обробки даних:

розробити систему для реєстрації замовника (з попередньою перевіркою на наявність такої інформації). Система дозволяє вносити замовлення. Передбачити можливість отримання довідки про різні види транспорту для певної ваги чи певного об'єму. Необхідно підраховувати вартість перевезення та суму до сплати (відстань*вартість т/км*маса + накладні витрати - пільги).

Завдання для побудови звітів:

знайти вартість перевезень для кожного виду транспорту за кожен місяць поточного року з підрахунком відсотку входження відносно загальної вартості;

знайти загальну вартість перевезень на відстань менш ніж 100км, 100-999км, 1000-4999км, більше 5000км за поточний рік.

ВАРІАНТ 21

Створіть в Access[©] базу даних про замовлення товарів, що складається з таблиці "Клієнти" з полями "Код клієнта", "Прізвище", "Адреса", "Фірма" і таблиці "Замовлення" з полями "Номер", "Код клієнта", "Назва товару", "Дата", "Кількість", "Сума". Типи полів виберіть за їх змістом. Створіть форму "Клієнтура" на основі таблиці "Клієнти" і введіть за її допомогою 10 записів в цю таблицю. Введіть 20 записів безпосередньо в таблицю "Замовлення", ураховуючи, що

один клієнт може замовити кілька товарів. Зробіть поле "Код клієнта" ключовим у таблиці "Клієнти" бази даних Access і зв'яжіть через нього таблиці "Клієнти" і "Замовлення" зв'язком типу "один-до-багатьох". Створіть запит "Підсумки" на основі таблиць "Клієнти" і "Замовлення", що показує всіх клієнтів разом з товаром, який вони замовили. На основі запиту "Підсумки" створіть звіт "Замовлені товари", розрахувавши загальну суму замовлених товарів.

ВАРІАНТ 22

Створіть в Access[®] базу даних постачальників обладнання. Для цього створіть таблицю "Постачальники" з полями "Код постач", "Фірма", "Адреса", "Особа", а також таблицю "Поставки" з полями "Номер", "Код постач", "Виріб", "Дата", "Кількість", "Сума". Типи полів виберіть за їх змістом. Створіть форму "Агенти" на основі таблиці "Постачальники" і введіть за її допомогою 7 записів у цю таблицю. Введіть 15 записів безпосередньо в таблицю "Поставки", урахувавши, що один постачальник може постачати кілька видів обладнання. Зробіть поле "Код постач" ключовим у таблиці "Постачальники" і зв'яжіть через нього таблиці "Постачальники" і "Поставки" зв'язком типу "один-до-багатьох". Створіть запит "Підсумки" на основі таблиць "Постачальники" і "Поставки", що показує всіх постачальників разом з обладнанням, яке вони постачали, якщо сума поставки перевищує 1000 грн. На основі запиту "Підсумки" створіть звіт "Поставки", розрахувавши загальну суму поставок, які задовольняють зазначену умову.

ВАРІАНТ 23

Створіть в Access[®] базу даних машин, які перебувають на ремонті. Для цього створіть таблицю "СТО" з полями "Корпус", "Майстер", "Номер дільниці", а також таблицю "Машини" з полями "Номер_дільниці", "ПІБ_власника", "Адреса", "Дата". Типи полів виберіть за їх змістом. Створіть форму "Станція" на основі таблиці "СТО" і введіть за її допомогою 10 записів у цю таблицю. Введіть 15 записів безпосередньо в таблицю "Машини", враховуючи, що на одній ділянці можуть ремонтуватись кілька машин. Зробіть поле "Номер дільниці" ключовим у таблиці "СТО" і зв'яжіть через нього таблиці "СТО" і "Машини" зв'язком типу "один-до-багатьох". На основі таблиць "СТО" і "Машини" створіть запит "Дільниця", який за номером дільниці, що повинен бути параметром запиту, видає всі дані про машини, які ремонтуються на цій дільниці. Створіть запит "Підсумки" і на його основі побудуйте звіт з такою ж назвою, який видає дані про всі машини, що ремонтуються на станції, а також розрахуйте у звіті загальну кількість машин.

ВАРІАНТ 24

Створіть в Access[®] базу даних працівників магазину. Для цього створіть таблицю "Штат" з полями "Код_посади", "Посада", "Оклад", а також таблицю "Співробітники" з полями "Таб_номер", "ПІБ", "Код_посади", "Дата_народження", "Дата_прийняття". Типи полів виберіть за їх змістом. Створіть форму "Штатна"

на основі таблиці "Штат" і введіть за допомогою цієї форми п'ять записів у цю таблицю. Введіть 15 записів безпосередньо в таблицю "Співробітники", урахувавши, що на одній посаді можуть працювати кілька співробітників. Зробіть поле "Код_посади" ключовим у таблиці "Штат" і зв'яжіть через нього таблиці "Штат" і "Співробітники" зв'язком типу "один-до-багатьох". Створіть запит "Співробітник" на основі таблиць "Штат" і "Співробітники", який за прізвищем працівника, що повинно бути параметром запиту, видає всі дані про цього працівника, тобто посаду, оклад та ін. Створіть запит "Весь штат" і на його основі побудуйте звіт "Штатні працівники", який видає дані про всіх співробітників, які працюють у магазині.

ВАРІАНТ 25

Створіть в Access[®] базу даних магазинів, які виконують замовлення. Для цього створіть таблицю "Магазини" з полями "Код_магазину", "Адреса", "Телефон", а також таблицю "Замовлення" з полями "Номер_замови", "Код_магазину", "Назва_товару", "Фірма", "Дата", "Ціна", "Кількість", "Сума". Типи полів виберіть за їх змістом. Створіть форму "Магазин" на основі першої таблиці і введіть за її допомогою 10 записів у цю таблицю. Введіть 20 записів безпосередньо в таблицю "Замовлення", враховуючи, що один магазин може виконувати кілька замовлень. Зробіть поле "Код_магазину" ключовим у таблиці "Магазини" і зв'яжіть через нього таблиці "Магазини" і "Замовлення" зв'язком типу "один до багатьох". Створіть запит "Підсумки" на основі цих двох таблиць, що показує всі магазини разом із замовленнями, якщо сума замовлення перевищує 500 грн. На основі запиту "Підсумки" створіть звіт "Підсумки", в якому розрахуйте загальну суму замовлених товарів, які задовольняють зазначену умову.

ВАРІАНТ 26

Створіть, в Access[®] базу даних про замовлення товарів, що складається з таблиці "Клієнти" і полями "Код клієнта", "Прізвище", "Адреса", "Фірма" і таблиці "Замовлення" з полями "Номер", "Код клієнта", "Назва товару", "Дата", "Кількість", "Сума". Типи полів виберіть за їх змістом. Створіть форму "Клієнтура" на основі таблиці "Клієнти" і введіть за її допомогою 10 записів в цю таблицю. Введіть 20 записів безпосередньо в таблицю "Замовлення", урахувавши, що один клієнт може замовити кілька товарів. Зробіть поле "Код клієнта" ключовим у таблиці "Клієнти" бази даних Access і зв'яжіть через нього таблиці "Клієнти" і "Замовлення" зв'язком типу "один-до-багатьох". Створіть запит "Підсумки" на основі таблиць "Клієнти" і "Замовлення", що показує всіх клієнтів разом з товаром, який вони замовили. На основі запиту "Підсумки" створіть звіт "Замовлені товари", розрахувавши загальну суму замовлених товарів.

ВАРІАНТ 27

Створіть в Access[®] базу даних постачальників обладнання. Для цього створіть таблицю "Постачальники" з полями "Код постач", "Фірма", "Адреса", "Особа", а також таблицю "Поставки" з полями "Номер", "Код постач", "Виріб", "Дата", "Кількість", "Сума". Типи полів виберіть за їх змістом. Створіть форму "Агенти" на основі таблиці "Постачальники" і введіть за її допомогою 7 записів у цю таблицю. Введіть 15 записів безпосередньо в таблицю "Поставки", ураховуючи, що один постачальник може постачати кілька видів обладнання. Зробіть поле "Код постач" ключовим у таблиці "Постачальники" і зв'яжіть через нього таблиці "Постачальники" і "Поставки" зв'язком типу "один-до-багатьох". Створіть запит "Підсумки" на основі таблиць "Постачальники" і "Поставки", ще показує всіх постачальників разом з обладнанням, яке вони поста чали, якщо сума поставки перевищує 1000 грн. На основі запиту "Підсумки" створіть звіт "Поставки", розрахувавши загальну суму поставок, які задовольняють зазначену умову.

ВАРІАНТ 28

Створіть в Access[®] базу даних працівників магазину. Для цього створіть таблицю "Штат" з полями "Код_посади", "Посада", "Оклад", а також таблицю "Співробітники" з полями "Таб_номер", "ПІБ", "Код_посади", "Дата_народження", "Дата_прийняття". Типи полів виберіть за їх змістом. Створіть форму "Штатна" на основі таблиці "Штат" і введіть за допомогою цієї форми п'ять записів у цю таблицю. Введіть 15 записів безпосередньо в таблицю "Співробітники", ураховуючи, що на одній посаді можуть працювати кілька співробітників. Зробіть поле "Код_посади" ключовим у таблиці "Штат" і зв'яжіть через нього таблиці "Штат" і "Співробітники" зв'язком типу "один-до-багатьох". Створіть запит "Співробітник" на основі таблиць "Штат" і "Співробітники", який за прізвищем працівника, що повинно бути параметром запиту, видає всі дані про цього працівника, тобто посаду, оклад та ін. Створіть запит "Весь штат" і на його основі побудуйте звіт "Штатні працівники", який видає дані про всіх співробітників, які працюють у магазині.

ВАРІАНТ 29

Створіть в Access[®] базу даних машин, які перебувають на ремонті. Для цього створіть таблицю "СТО" з полями "Корпус", "Майстер", "Номер ділянки", а також таблицю "Машини" з полями "Номер_ділянки", "ПІБ_власника", "Адреса", "Дата". Типи полів виберіть за їх змістом. Створіть форму "Станція" на основі таблиці "СТО" і введіть за її допомогою 10 записів у цю таблицю. Введіть 15 записів безпосередньо в таблицю "Машини", ураховуючи, що на одній ділянці можуть ремонтуватись кілька машин. Зробіть поле "Номер ділянки" ключовим у таблиці "СТО" і зв'яжіть через нього таблиці "СТО" і "Машини" зв'язком типу "один до багатьох". На основі таблиць "СТО" і

"Машини" створіть запит "Дільниця", який за номером ділянки, що повинен бути параметром запиту, видає всі дані про машини, які ремонтуються на цій ділянці. Створіть запит "Підсумки" і на його основі побудуйте звіт з такою ж назвою, який видає дані про всі машини, які ремонтуються на стації, а також розрахуйте у звіті загальну кількість машин.

ВАРІАНТ 30

Створіть в Access[®] базу даних магазинів, які виконують замовлення. Для цього створіть таблицю "Магазини" з полями "Код_магазину", "Адреса", "Телефон", а також таблицю "Замовлення" з полями "Номер_замови", "Код_магазину", "Назва_товару", "Фірма", "Дата", "Ціна", "Кількість", "Сума". Типи полів виберіть за їх змістом. Створіть форму "Магазин" на основі першої таблиці і введіть за її допомогою 10 записів у цю таблицю. Введіть 20 записів безпосередньо в таблицю "Замовлення", враховуючи, що один магазин може виконувати кілька замовлень. Зробіть поле "Код_магазину" ключовим у таблиці "Магазини" і зв'яжіть через нього таблиці "Магазини" і "Замовлення" зв'язком типу "один до багатьох". Створіть запит "Підсумки" на основі цих двох таблиць, що показує всі магазини разом із замовленнями, якщо сума замовлення перевищує 500 грн. На основі запиту "Підсумки" створіть звіт "Підсумки", в якому розрахуйте загальну суму замовлених товарів, які задовольняють зазначену умову.

Рекомендована література

1. Бобцов А.А., Шиегин В.В. Банки и базы данных. Основы работы с MS Access. Часть 1 (для пользователей): Уч. пособие. – СПб., 2005.
2. Карпов Б. Microsoft Access 2000: Справочник // СПб: «Питер», 2001.
3. Форт С., Хоуи Т., Релстон Дж. Программирование в среде Access 2000. Энциклопедия пользователя // К.: Изд «ДиаСофт», 2000.
4. Справка по Microsoft Access (входит в состав пакета Access).
5. Кириллов В.В., Громов Г.Ю. Структурированный язык запросов (SQL). Учебное пособие // СПбГИТМО(ТУ). (электронная версия доступна в интернет по адресу http://citforum.ru/database/sql_kg/)
6. Кузнецов С.Д. Введение в стандарты языка баз данных SQL // в электронной форме: <http://citforum.ru/database/sqlbook/>
7. Мартин Грабер. Понимание SQL (Understanding SQL) // в электронной форме: http://www.sql.ru/docs/sql/u_sql/
8. Кириллов В.В. Основы проектирования реляционных баз данных. Учебное пособие // СПбГИТМО(ТУ). (электронная версия доступна в интернет по адресу <http://citforum.ru/database/dbguide/>)
9. Джен Л. Харрингтон. Проектирование реляционных баз данных // М.: «Лори», 2000.
10. Ролланд Ф.Д. Основные концепции баз данных // М.- СПб - Киев: «Вильямс», 2002.
11. Дейт К. Введение в системы баз данных, 6 изд. - К.: «Диалектика», 1998.
12. Крёнке Д. Теория и практика построения баз данных, 9 изд. -СПб: «Питер», 2005.
13. Пушников А.Ю. Введение в системы управления базами данных (в 2 ч.): Уч. пособие // Изд Башкирского ун-та. - Уфа, 1999. (электронная версия доступна в интернет по адресу <http://citforum.ru/database/dblearn/>)

Навчальне видання

Методичні вказівки до самостійної роботи та лабораторних занять з дисципліни «Бази даних » для студентів другого курсу денної і заочної форм навчання напряму підготовки 6.080101 «Геодезія, картографія та землеустрій» спеціальності 6.070900 «Геоінформаційні системи і технології».

Укладач: Андрій Анатолійович Євдокімов,

Редактор: М.З.Аляб`єв

Верстка: Ю.П. Степась

План 2009, 109М

Підп. до друку 07.10.09	Формат 60x84 1/16	Папір офісний.
Друк на ризографі.	Ум.-друк арк. 3,5	Обл.-вид. арк. 4,0
Замовл. №	Тираж 50 прим.	

61002, Харків, ХНАМГ, вул. Революції, 12.

Сектор оперативної поліграфії ЦНІТ ХНАМГ

61002, Харків, вул. Революції, 12.