МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МИКОЛАЇВСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
Навчально-науковий інститут економіки та управління
Обліково-фінансовий факультет

Кафедра інформаційних систем і технологій

ГЕОІНФОРМАЦІЙНІ СИСТЕМИ І БАЗИ ДАНИХ

методичні рекомендації
до виконання практичних робіт в СУБД Microsoft Access
для здобувачів першого (бакалаврського) рівня вищої освіти
ОПП «Геодезія та землеустрій» спеціальності 193 «Геодезія та землеустрій» денної форми здобуття вищої освіти

МИКОЛАЇВ
2022

УДК 004.65:631.4-027-44
Г35

Друкується за рішенням науково–методичної комісії обліково–фінансового факультету Миколаївського національного аграрного університету від 21 квітня 2022 р., протокол № 9.

Укладач:

Л. О. Борян – ст. викладач кафедри інформаційних систем і технологій Миколаївського національного аграрного університету

Рецензенти:

Л. М. Макарова – канд. техн. наук, доцент кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування ім. адм. Макарова;

Є.Ю. Борчик – канд. ф-м. наук, доцент кафедри вищої та прикладної математики Миколаївського національного аграрного університету.

© Миколаївський національний аграрний університет, 2022

Передмова

Методичні рекомендації розроблені для здобувачів першого (бакалаврського) рівня вищої освіти ОПП «Геодезія та землеустрій» спеціальності 193 «Геодезія та землеустрій» денної форми здобуття вищої освіти з дисципліни «Геоінформаційні системи і бази даних».

Основна мета методичних рекомендацій – підготувати здобувачів вищої освіти до самостійної роботи на комп'ютері, формування у здобувачів вищої освіти знань та умінь з геоінформаційних технологій, збору й обробки інформації, поглиблене вивчення найбільш поширених технологій автоматизації офісу та програмних засобів колективного користування.

Методичні рекомендації містять матеріал, необхідний для засвоєння теоретичних питань та навиків використання персонального комп'ютера для виконання практичних завдань. Програма MS ACCESS – це потужний засіб автоматизації роботи офісу, який підвищує продуктивність праці спеціаліста з використанням комп'ютера.

Цей матеріал рекомендується використовувати для вивчення дисципліни «Геоінформаційні системи і бази даних».

Оцінка виконаного завдання становить – 5 балів.

ТЕОРЕТИЧНІ ОСНОВИ СИСТЕМИ УПРАВЛІННЯ БАЗАМИ ДАНИХ (СУБД) MS ACCESS

Проектування бази даних.

Будь-яку базу даних (БД), навіть невелику за розмірами, слід правильно спроектувати.

Ретельно продумана і вдало розроблена база даних стане гарантом вашого спокою в майбутньому, адже краще одразу створити оптимальний для конкретних цілей проект, ніж згодом займатися реструктуризацією невдало спланованої бази даних, у якій уже зберігається не одна сотня (а можливо, і тисяча) записів.

Розглянемо основні етапи проектування баз даних.

Етап 1. Загальне проектування і мета створення бази даних.

На цьому етапі визначається загальна концепція створюваного проекту. У першу чергу розроблювачеві слід виконати наступне:

- Порадитися з майбутніми користувачами цієї системи і з'ясувати призначення проектованої бази даних.
- У ході опитування безпосередніх користувачів БД варто виявити і по можливості формалізувати звичайну послідовність їхніх дій, працюючи з тією інформацією, що має бути внесена в БД.
- Уважно ознайомитися з уже використовуваними формами реєстрації даних (наприклад, бланками, довідками, журналами і т.п.).
- Обговорити з замовником ескізи майбутніх звітів. Якщо заздалегідь відомо, який вигляд матимуть звіти нової бази даних, можна вибрати для таблиць структуру, що оптимально відповідає вимогам звіту.

У міру того, як визначатиметься призначення БД, будуть конкретизуватися і ті фактичні дані, що повинні зберігатися й оброблятися в цій базі. На даному етапі вже виділяються окремі предмети або об'єкти, яким будуть відповідати таблиці, і в першому наближенні намічаються обов'язкові поля цих таблиць, куди вноситимуться дані про основні характеристики об'єктів.

Про мету створення БД можна сказати наступне: основні функції бази даних полягатимуть у створенні, видаленні й відновленні записів; пошуку записів по ідентифікатору або умові; підрахунку числа записів, що відповідають заданій умові; формуванню поточних звітів.

Етап 2. Визначення потрібних таблиць і полів бази даних.

Ось короткий перелік дій, які необхідно виконати на даному етапі:

- Напишіть список таблиць, з яких складатиметься база даних, визначте імена таблиць.
- Для кожної таблиці визначте необхідний набір полів і їхнє положення в таблиці.

- Для кожного поля виберіть ім'я і тип даних, що будуть у ньому зберігатися. Визначте розмір текстових полів і підтип числових.

Продумайте і сформулюйте для полів правила перевірки допустимості даних, що будете вводити.

Правила і рекомендації, яких необхідно дотримуватися.

- Намагайтеся внести в базу даних всю інформацію, з якою необхідно буде працювати.
- Створюйте прості таблиці, використовувати які завжди зручніше, пам’ятайте, що додаток Access має великі можливості щодо зв'язку таблиць між собою.
- Іноді хорошою основою для створення таблиць є вже знайомі і використовувані у вашій роботі паперові носії інформації — форми, бланки і т.п. Як правило, надалі на основі таблиць створюються електронні еквіваленти цих форм.
- Уникайте давати таблицям і полям бази даних імена, що можуть збігтися з зарезервованими словами самої системи Access (наприклад, SELECT, date), щоб уникнути конфліктів і появи помилок при виконанні запитів.
- Формуючи список полів таблиці, розбивайте інформацію про об'єкт на мінімальні логічні компоненти.
- Якщо дані містять перерахування (наприклад, типи товарів), придивіться до них уважніше. Можливо, такі поля доведеться виділити в окрему таблицю.
- Не рекомендується поміщати в таблицю поля, значення яких є результатом обчислення (тобто можуть бути обчислені на підставі даних з інших полів таблиці, як у разі з відомими значеннями кількості й вартості замовленого товару, сума замовлення є простим добутком цих значень).
- Призначення поля загалом має бути зрозуміле з його назви. Наприклад, Користувачі — таблиця користувачів, Товари — каталог товарів.
- Хорошим стилем вважається використання складних назв таблиць і полів. При цьому перше слово має означати сутність інформаційності об'єкта, а друге слово (або інші слова) уточнення, наприклад, КодТовара, ДатаНарождения. І хоча Access допускає використання пропусків у назвах полів, краще все-таки залишати такі імена без пропусків.

Обмеження Access на імена полів і об'єктів:

- Ім'я не може починатися з пропуску.
- В імені поля або таблиці може бути до 64 символів. Однак на практиці подібна багатослівність є зайвою. Не слід вдаватися і в іншу крайність, придумуючи для полів або таблиць малозрозумілі символи,

що нагадують шифр, назви.

- Ім'я може мати будь-яку комбінацію букв, цифр, пропусків і спеціальних знаків за винятком крапки (.), знака оклику (!), надрядкового знака (‘), квадратних дужок ([ ]).

Етап 3. Визначення полів з унікальними значеннями в кожному записі.

При проектуванні таблиць обов'язковою умовою є використання першого поля таблиці (або набору полів) для однозначної ідентифікації записів таблиці. Таке поле є первинним ключем таблиці й на наступному етапі братиме безпосередню участь в організації зв'язків між таблицями.

Етап 4. Визначення зв'язків між таблицями.

Зв'язок між таблицями здійснюється за допомогою спільного поля.

Етап 5. Удосконалення структури БД.

При розробці великих проектів зі складною організацією бази даних на даному етапі обов'язково проводиться заповнення таблиць спеціально підготовленими тестовими даними. У якості таких даних можуть використовуватися як свідомо правильні дані, за допомогою яких перевіряється сама структура таблиць і коректність генерованих звітів, так і "помилкові" дані, що дозволяють перевірити припустимі межі значень для полів таблиць і виявити всі можливі типи помилок.

На даному етапі доцільно також створити в першому наближенні майбутні форми і звіти, щоб упевнитися в тому, що в них будуть заноситися саме ті дані, що необхідні. На тестових наборах даних за допомогою запитів перевіряються зв'язки, установлені для різних таблиць.

Етап 6. Введення даних і створення інших об'єктів БД.

Після створення й успішного тестування структури таблиць можна заповнювати їх реальними даними. На цьому етапі розробляються і створюються також усі інші необхідні об'єкти бази даних — запити, форми, звіти, сторінки доступу до даних, макроси і модулі.

Методи створення оптимальної моделі баз даних.

Під оптимальною логічною моделлю баз даних розуміють модель, яка не має аномалій, пов'язаних з модифікацією БД, тобто проблем, що можуть виникнути у зв'язку із замінами, вставками і вилученнями даних із БД.

Для створення такої моделі баз даних застосовується теорія нормалізації реляційних баз даних.

В основу реляційних моделей покладено поняття відношення, яке подають у вигляді двовимірної таблиці.

Реляційна БД — це набір взаємопов'язаних відношень. Кожне відношення (таблиця) в ЕОМ подається як файл. Відношення можна поділити на два класи: об'єктні й зв'язкові.

Об'єктні відношення зберігають дані про інформаційні об'єкти предметної області. Наприклад; КЛІЄНТ (код клієнта, назва клієнта, адреса, телефон) є об'єктним відношенням.

В об'єктному відношенні один з атрибутів однозначно ідентифікує окремий об'єкт. Такий атрибут називається *первинним ключем* відношення. У наведеному відношенні роль ключа виконує атрибут «код клієнта».

Ключ може мати кілька атрибутів, тобто бути складеним. В об'єктному відношенні не повинно бути рядків з однаковим ключем, тобто не допускається дублювання об'єктів. Це основне обмеження реляційної моделі для забезпечення цілісності даних.

Зв'язкове відношення зберігає первинні ключі двох або більше об'єктних відношень. Ключі зв'язкового відношення мають на меті встановлення зв'язків між об'єктними відношеннями.

Розглянемо, наприклад, ще одне об'єктне відношення БАНК (Код банку, назва банку, адреса банку).

Тоді зв'язкове відношення БАНК-КЛІЄНТ (код банку, код клієнта) буде сполучним між двома об'єктними відношеннями БАНК і КЛІЄНТ. У зв'язковому відношенні можуть дублюватися ключові атрибути. Крім ключів, за якими встановлюють зв'язок у зв'язковому відношенні, можуть бути ще й інші атрибути, які функціонально залежать від цього складового ключа.

Ключі в зв'язкових відношеннях називаються *вторинними або зовнішніми ключами,* оскільки вони є первинними ключами об'єктів інших відношень. Реляційна модель накладає на зовнішні ключі обмеження, яке називають посилковою цілісністю. Воно необхідне для забезпечення цілісності даних.

Посилкова цілісність — це відповідність між об'єктними та зв'язковими відношеннями, яка полягає в тому, що кожному зовнішньому ключеві зв'язкового відношення має відповідати рядок якогось об'єктного відношення.

Нормалізація відношень.

Нормалізація відношень — це ітераційний зворотний процес декомпозиції початкового відношення на кілька простіших відношень меншої розмірності. Під зворотністю процесу розуміють те, що операція об'єднання відношень, здобутих у результаті декомпозиції, має дати початкове відношення. Внаслідок нормалізації склад атрибутів відношень БД має відповідати таким вимогам:

• між атрибутами не повинно бути небажаних функціональних залежностей;

• групування атрибутів не повинно мати збиткового дублювання даних;

• забезпечувати обробку і поновлення атрибутів без ускладнень.

Відношення перебуває в 1НФ тоді і тільки тоді, коли на перетині кожного стовпця і кожного рядка є тільки елементарне значення атрибутів.

Відношення, які знаходяться в 1НФ називають просто нормалізованими відношеннями. Відповідно, ненормалізовані відношення можуть інтерпретуватися як таблиці з нерівномірним заповненням, наприклад, таблиця „Розклад”:

Викладач	День тижня	Номер пари	Назва дисципліни	Тип занять	Група
Петров	Понеділок	1	Інформатика	Лекція	4906
	Вівторок	1	Інформатика	Лаб. роб.	4907
	Вівторок	2	Інформатика	Лаб. роб.	4906
Кіров	Понеділок	2	СУБД	Лекція	4906
	Вівторок	3	СУБД	Лаб. роб.	4907
	Вівторок	4	СУБД	Лаб. роб.	4906
Серов	Понеділок	3	Комп’ютерні мережі	Лекція	4944 4942
	Середа	3	Комп’ютерні мережі	Лаб. роб.	4922
	Четвер	4	Комп’ютерні мережі	Лаб. роб.	

Для приведення відношення „Розклад” до 1НФ необхідно доповнити кожний рядок прізвищем викладача:

Викладач	День тижня	Номер пари	Назва дисципліни	Тип занять	Група
Петров	Понеділок	1	Інформатика	Лекція	4906
Петров	Вівторок	1	Інформатика	Лаб. роб.	4907
Петров	Вівторок	2	Інформатика	Лаб. роб.	4906
Кіров	Понеділок	2	СУБД	Лекція	4906
Кіров	Вівторок	3	СУБД	Лаб. роб.	4907
Кіров	Вівторок	4	СУБД	Лаб. роб.	4906
Серов	Понеділок	3	Комп’ютерні мережі	Лекція	4944
Серов	Середа	3	Комп’ютерні мережі	Лаб. роб.	4942
Серов	Четвер	4	Комп’ютерні мережі	Лаб. роб.	4922

Відношення знаходиться в 2НФ тільки тоді, коли воно перебуває в 1НФ і не містить неповних функціональних залежностей не первинних атрибутів від атрибутів первинного ключа.

Структура відношення, що моделює складання студентами сесії, визначається набором атрибутів (ПІБ, Номер зал.кн., Група, Дисципліна, Оцінка).

Тому, що кожен студент складає багато дисциплін під час сесії, первинним ключем відношення може бути (Номер зал. кн., Дисципліна), який однозначно визначає кожний рядок відношення. З іншого боку, атрибути ПІБ і Група залежать тільки від частини первинного ключа – від значення атрибута Номер зал.кн., тому ми маємо неповні функціональні залежності в даному відношенні. Для приведення даного відношення до 2НФ слід розбити його на проекції. Такими проекціями можуть бути відношення:

(ПІБ, Номер зал.кн., Група)

(Номер зал.кн., Дисципліна, Оцінка)

Відношення перебуває в 3НФ тільки тоді, коли воно є в 2НФ і не має транзитивних залежностей.

Структура відношення, що пов’язує студентів з групами, факультетами і спеціальностями, на яких вони вчаться

(ПІБ, Номер зал.кн., Група, Факультет, Спеціальність, Випускова кафедра).

Первинним ключем відношення є Номер зал.кн. Група, в якій навчається студент, однозначно визначає факультет, на якому він навчається, а також спеціальність і випускову кафедру. Крім того, ця кафедра однозначно визначає факультет, на якому навчаються студенти-випускники.

У цьому разі маємо наступні функціональні залежності:

Номер зал.кн. → ПІБ
Номер зал.кн. → Група
Номер зал.кн. → Факультет
Номер зал.кн. → Спеціальність
Номер зал.кн. → Випускова кафедра

Група → Факультет
Група → Спеціальність
Група → Випускова кафедра
Випускова кафедра → Факультет

Ці залежності утворюють транзитивні групи. Для того щоб уникнути цього ми можемо створити такі відношення:

(Номер зал.кн., ПІБ, Спеціальність, Група)
(Група, Випускова кафедра)
(Випускова кафедра, Факультет)

Первинні ключі відношення виділені.

Відношення R перебуває в 4НФ лише тоді, якщо у разі існування багатозначної залежності A→B всі інші атрибути R функціонально залежать від A.

Дано відношення, яке моделює здачу екзаменів на сесії.

(Номер зал.кн., Група, Дисципліни)

Перелік дисциплін, які має здавати студент, однозначно визначається номером групи (тобто спеціальністю, на якій він навчається)

У даному відношенні існують наступні дві багатозначні залежності:

Група → Дисципліна
Група → Номер зал.кн.

Це означає, що кожній групі однозначно відповідає перелік дисциплін за навчальним планом і номер групи визначає список студентів, які навчаються в цій групі.

Можна провести декомпозицію початкового відношення за два відношення:

(Номер зал.кн., Група)
(Група, Дисципліна)

Обидва ці відношення перебувають у 4НФ і вільні від аномалій.

Тож, обидві операції модифікації тепер спрощуються: доповнення нового студента пов’язане з доповненням усього одного кортежа в перше відношення, а додавання нової дисципліни входить у додавання одного кортежа в друге відношення. Крім цього, в другому відношенні ми можемо зберігати будь-яку кількість груп з визначеним переліком дисциплін, в які поки що не зараховані студенти.

Останньою нормальною формою є 5НФ, яка пов’язана з аналізом нового виду залежностей „проекції з’єднання”, і позначається PJ.

Відношення R перебуває в 5НФ (PJ/NF) лише в тому разі, коли будь-яка залежність з’єднання в R витікає із існування деякого можливого ключа в R.

Розглянемо відношення R1:

R1(Викладач, Кафедра, Дисципліна)

Кожен викладач може працювати на декількох кафедрах і на кожній з них може мати декілька дисциплін. У цьому разі ключем відношення є повний набір із трьох атрибутів. У відношенні відсутні багатозначні залежності, і тому відношення перебуває в 4НФ.

Введемо такі позначення наборів атрибутів:

ПК (Викладач, Кафедра)

ПД (Викладач, Дисципліна)

КД (Кафедра, Дисципліна)

Припускаємо, що відношення R1 задовольняє залежність проекції з’єднання (ПК, ПД, КД). Тоді відношення R1 не перебуває в NF/PJ, тому що єдиним ключем його є повний набір атрибутів, а наявність залежності PJ пов’язана з набором атрибутів, що не становлять можливі ключі відношення R1. Для того, щоб привести це відношення до NF/PJ, його слід представити у вигляді трьох відношень:

R2(Викладач, Кафедра)

R3(Викладач, Дисципліна)

R4(Кафедра, Дисципліна)

5НФ рідко використовується на практиці. В основному вона є теоретичним дослідженням.

Побудова логічної моделі даних.

Наступний крок після нормалізації відношень - це побудова логічної моделі БД.

Логічна модель є основою бази даних, вона повинна відображати взаємозв'язки між реляційними таблицями. Між реляційними таблицями можуть бути наступні типи зв'язків - 1:1, 1:Б та Б:Б. Найбільш поширеним є зв'язок 1:Б. Зв'язок 1:1 зустрічається рідше, тому що дані, між якими існує такий тип зв'язку здебільшого входять до складу однієї реляційної таблиці. Зв'язок Б:Б безпосередньо не підтримується в реляційних СУБД. Для реалізації такого зв'язку слід створювати додаткову реляційну таблицю, яка відіграватиме роль зв'язкової. Зв'язкова таблиця має обов'язково містити первинні ключі таблиць, між якими встановлюється зв'язок.

Деякі сучасні СУБД мають інструментальні засоби побудови логічної моделі даних. Розглянемо побудову такої моделі засобами СУБД Microsoft Access. Логічна модель у середовищі СУБД Microsoft

Access називається схемою. Для побудови схеми даних попередньо необхідно створити таблиці, визначивши в них первинні ключі.

Зв'язок установлюється між полем первинного ключа об'єктного відношення та полем вторинного ключа зв'язкового відношення. Зв'язок між полем первинного і полем вторинного ключа встановлюється, якщо поля задані одним типом і подані в одному форматі.

Схема в Access забезпечує за замовчуванням посилкову цілісність та узгодженість даних, що зберігаються в різних таблицях. Так, наприклад, Access не дозволить до записати в таблицю «Реалізація» дані про реалізацію виробу якомусь споживачеві, якщо дані про цей виріб та відповідного споживача відсутні в таблицях «Виріб» та «Довідник споживачів». Крім того, якщо користувач системи зробить спробу вилучити дані про якісь вироби чи про певних споживачів з таблиць довідкової інформації «Виріб» та «Довідник споживачів» тоді, коли дані про ці вироби та про споживачів занесені до таблиць «Реалізація» та «Договір», то Access повідомить його про те, що вилучення відповідних рядків, з головної таблиці неможливе, тому, що про це є відповідні записи у підпорядкованих таблицях.

Зв’язок „один-до-одного”

Таблиця А
Відомості про співробітників

№	Найменування
1	Табельний номер
2	Прізвище, ім’я, по батькові
3	Дата народження
4	Адреса
5	Телефон
6	Фотографія
7	Примітки

Зв’язок 1:1

Таблиця В
Професійні відомості

№	Найменування
1	Табельний номер
2	Посада
3	Зарплата
4	Рейтинг
5	Дата прийняття на
6	роботу
7	Категорія

Зв’язок „один-до-багатьох”

Таблиця А
Клієнти

№	Найменування
1	Табельний номер
2	Прізвище, ім’я, по батькові
3	Дата народження
4	Місто
5	Адреса
6	Підприємство
7	Примітки

Зв’язок 1:Б

Таблиця В
Замовлення

№	Найменування
1	Код клієнта
2	Код товару
3	Дата замовлення
4	Замовлено, шт.
5	Дата продажу
6	Продано
7	Примітки

Зв’язок „багато-до-багатьох”

Таблиця А
Постачання товару

№	Найменування
1	Код підприємства
2	Код товару
3	Ціна
4	Мінімальна плата
5	Термін постачання
6	Примітки

Зв’язок Б:Б

Таблиця В
Замовлення клієнтів

№	Найменування
1	Код клієнта
2	Код товару
3	Ціна
4	Кількість
5	Дата постачання
6	Примітки

ПРАКТИЧНА РОБОТА № 1. Створення бази даних

Завдання. Фірма спеціалізується на продажі сільськогосподарської продукції. Продажу організують менеджери фірми через філіали, що знаходяться у регіонах. Згідно наведеного фрагмента даних продаж запроектувати реляційну модель бази даних.

Таблиця 1 Вхідні дані

Код	Менеджер (П І Б)	Назва сільського-сподарської продукції	Рік вироб-ництва	Кількість, (т)	Дата продажу	Вартість, (грн за 1 т)	Код регіону	Назва регіону (область)
1	Іванченко	Пшениця	2019	680	03/2010	780	01	Київська
1	Іванченко	Пшениця	2019	125	04/2010	890	01	Київська
1	Іванченко	Соняшни	2018	110	05/2010	800	01	Київська
1	Іванченко	Ячмінь	2018	600	06/2010	600	01	Київська
2	Дік П.В.	Жито	2018	297	04/2010	950	02	Львівська
2	Дік П.В.	Жито	2018	165	04/2010	930	02	Львівська
2	Дік П.В.	Пшениця	2019	500	05/2010	880	02	Львівська
2	Дік П.В.	Пшениця	2019	125	04/2010	900	02	Львівська
2	Дік П.В.	Овес	2019	500	05/2010	775	02	Львівська
3	Котов П.Ю.	Пшениця	2018	680	03/2010	820	02	Львівська
3	Котов П.Ю.	Жито	2018	220	04/2010	540	02	Львівська
4	Шмига Р.В.	Жито	2018	156	03/2010	530	03	Тернопільська
4	Шмига Р.В.	Соняшни	2018	200	05/2010	800	03	Тернопільська
4	Шмига Р.В.	Пшениця	2019	350	05/2010	840	03	Тернопільська
5	Шедров В.Ф.	Пшениця	2019	133	04/2010	833	04	Харківська
5	Шедров В.Ф.	Соняшни	2019	500	04/2010	980	04	Харківська
5	Шедров В.Ф.	Ячмінь	2018	700	04/2010	780	04	Харківська
5	Шедров В.Ф.	Овес	2018	950	04/2010	710	04	Харківська
6	Федірко С.Г.	Соняшни	2019	100	04/2010	920	05	Полтавська
6	Федірко С.Г.	Пшениця	2019	500	03/2010	870	05	Полтавська

Створити базу даних продажі сільськогосподарської продукції.

Послідовність виконання роботи.
Створення таблиці.

Для того, щоб завантажити **Microsoft Access**, у головному меню натискуємо кнопку **Пуск**, обираємо команду **Программы,** потім **Microsoft Access**

1. В оболонці **Microsoft Access**, у головному меню **Файл** оберемо команду **Создать**.
2. У діалоговому вікні **Создание** виділимо піктограму **Новая база данных** і натискуємо кнопку **ОК.**
3. Зберігаємо базу даних за допомогою команди **Сохранить базу даных как** з ім'ям **«Продаж»** у своїй папці.
4. Оберемо вкладку **Создать** команда **Конструктор таблиц**.
5. У діалоговому вікні **Таблица** вводимо імена полів у стовпчик.
6. Створюємо структури всіх таблиць бази даних **«Продаж».**
7. У діалоговому вікні бази даних обираємо об'єкт **Таблицы,** заповнюємо таблиці **„Менеджер"**, **„Продукція"**, **"Регіон"** даними. *Обов' язкові вимоги:*
   a. Стовпець **Описание** призначений для короткого викладу змісту.
   b. У групі **Свойства поля** установіть необхідні властивості полів.
   c. Для поля, відсутність даних в якому лишає запис смислу, встановіть опцію **ДА** в полі **Обязательное поле.**
   d. Для текстових полів в полі **Размер поля** задаємо максимальне число символів, які може містити поле.

Структура БД

Таблиця **“*Менеджер*”**

Ім’я поля	*Тип даних*	*Довжина або розмір*	*Індексоване поле*
Код менеджера	**Счетчик**	**Длинное целое**	**Да (Совпадения не допускаются)**
ПІБ	**Текстовый**	50	**Нет**

Таблиця ***“Регіон”***

Ім’я поля	*Тип даних*	*Довжина або розмір*	*Індексоване поле*
Код регіону	**Счетчик**	**Длинное целое**	**Да (Совпадения не допускаются)**
Назва регіону	**Текстовый**	50	**Нет**

Таблиця ***“Продукція”***

Ім’я поля	*Тип даних*	*Довжина або розмір*	*Індексоване поле*
Код продукції	**Счетчик**	**Длинное целое**	**Да (Совпадения не допускаются)**
Назва продукції	**Текстовый**	50	**Нет**

Таблиця ***“Продаж”***

Ім’я поля	*Тип даних*	*Довжина або розмір*	*Індексоване поле*
Код	**Счетчик**	**Длинное целое**	**Да (Совпадения не допускаются)**
Код продукції	**Числовой**	**Длинное целое**	**Да (Совпадения допускаются)**
Рік виробництва	**Текстовый**	10	**Нет**
Кількість	**Числовой**	**Целое**	**Нет**
Дата продажу	**Дата/время**	**Средн.формат даты**	**Нет**
Вартість	**Числовой**	**Одинарное с плавающей точкой**	**Нет**
Код менеджера	**Числовой**	**Длинное целое**	**Нет**
Код регіону	**Числовой**	**Длинное целое**	**Нет**

Створення схеми даних.

1. Вибрати вкладку **Работа с базами даннях** команда **Схема данных**. Створити схему даних для роботи з таблицями „Менеджер”, „Продукція”, ”Регіон”, ”Продаж”.
2. Для створення схеми даних потрібно вибрати таблиці. В діалоговому вікні **„Добавление таблицы’’** найти таблиці, для яких створюється схема даних.
3. На екрані з’являється вікно **„Добавление таблицы’’**. Взяти закладку **„Таблицы’’** з переліком таблиць, за допомогою кнопки **„Добавить”** вибрати таблиці, вийти з діалогового вікна, клацнувши мишею по кнопці **„Закрыть”.**
4. За допомогою миші створити зв’язки між таблицями:
   таблиця „Продукція” поле ”Код продукції” → таблиця ”Продаж” поле ”Код продукції”.
   таблиця „Менеджер” поле ”Код менеджера” → таблиця”Продаж” поле ”Код менеджера”.
   таблиця ”Регіон” поле ”Код регіону” → таблиця ”Продаж” поле ”Код регіону”.

   Підводимо покажчик миші до потрібного поля, натискуємо ліву клавішу і перетягуємо покажчик до такого ж поля у пов’язаній таблиці, клацаємо і з’являється діалогове вікно **„Изменение связей”**. У цьому вікні встановити прапорець **„Обеспечение целостности данных”**, натиснути кнопку **„Создать”**.
5. Для зміни зв’язку використовується права клавіша миші. В меню, що з’являється після натискання правої клавіші, пункт **„Изменить связь” -** для зміни зв’язку, пункт **„Удалить” -** для знищення зв’язку. Після встановлення зв’язків закрити схему даних і зберегти. На запитання **„Сохранить изменения макета „Схема данных ?”** відповісти **„Да”.**

Створення форм.

1. Вибрати об’єкт **Формы.** Створити форми для роботи з таблицями „Менеджер”, „Продукція”, ”Регіон”, ”Продаж”.
2. Вибрати вкладку **Создание** команда **Мастер форм** і таблицю для якої створюється форма.
3. На екрані з’являється вікно **Создание форм**. Наводиться таблиця з переліком доступних полів. Вибрати поля, що будуть входити у форму за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее.**
4. Вибрати вид оформлення екрана, кнопка **Далее.**
5. Вибрати стиль оформлення екрана, кнопка **Далее.**
6. Задайте назву форми, кнопка **Готово.**
7. Для форми „Продаж” у режимі **Конструктора** створити додаткові елементи. На панелі інструментів вибрати **Элементы управления.**
8. За допомогою елемента **Надпись** створити назву форми.
9. За допомогою елемента **Кнопка** створити кнопку закриття форми.

У діалоговому вікні **Создание кнопок** вибрати категорію **Работа с формой,** дію **Закрытие формы,** зовнішній вигляд кнопки і дати їй назву.

10. Для поля „Код продукції”, „Код менеджера”, „Код регіону” створити елемент **Поле со списком.** За допомогою елемента **Поле со списком** викликати діалогове вікно **Создание полей со списком.**
    a. Майстер створить поле зі списком в якому будуть відображатися значення для вибору. Користувач показує яким чином ці значення буде отримувати **Поле со списком**.
    b. Для поля „Код продукції” вибрати джерелом таблицю „Продукція”, для поля „Код менеджера” вибрати джерелом таблицю „Менеджер”, для поля „Код регіону” вибрати джерелом таблицю „Регіон”.
    c. Вибрати поля таблиці, які стануть стовпцями в об’єкті **Поле со списком.**
    d. Вказати ширину стовпця в об’єкті **Поле со списком**, де зберегти вибране значення, дати назву об’єкта, натиснути кнопку **„Готово’’.**
11. Використовуючи форму ”Продаж”, завантажте БД даними.

Створення запитів.

Теоретичні відомості.

Для створення запитів потрібно використати вкладку **Создание** команда **Конструктор запросов.** Вікно запиту на вибирання має дві частини. Верхня містить схему даних запиту, що включає вибрані таблиці зі списками полів. Нижня частина вікна є бланком запиту за зразком. Кожний стовбець бланка стосується одного поля. Поля можуть використовуватися для введення їх у результат виконання запиту, сортування за цим полем і задання умов відбору.

Для заповнення бланка запиту необхідно виконати такі дії:

- у рядок "**Поле**" ввести імена полів, що застосовуються;
- у рядку "**Сортировка**" зазначити порядок сортування відібраних записів у підсумковій таблиці.
- У рядку "**Вывод на экран**" назвати поля, що введені в підсумкову таблицю, поставивши позначку ∨.
- У рядку "**Условия отбора**" сформувати логічні умови (критерії) відбирання записів, що можуть мати символи (* ; # ; ! ; ?), а також логічні функції И та ИЛИ.

Крім полів, вибраних із таблиць, у нижній частині вікна запиту на вибір можуть формуватися поля, визначені за значенням інших полів таблиць, а також підсумкові поля із застосуванням групових функцій. Для формування таких полів використовуються арифметичні вирази і вбудовані функції.

Для виконання групових операцій служить команда **Итоги** на вкладці **Работа с запросами** для того, щоб у вікні запиту з'явилося поле **Группировка**. У полі **Группировка** можна вибрати операцію **Группировка** для текстових полів, операцію **Count, Sum, Min, Max** для числових полів.

Для створення розрахункового поля слід:

- установити курсор у бланку запиту у не заповнений стовпчик, натиснути команду **Построитель** на вкладці **Работа с запросами**;
- у полі **Построителя выражений** набрати розрахунковий вираз для цього слід:
  - вибрати в переліку полів запиту ті, що потрібні та натиснути кнопку **Вставить**;
  - натиснути кнопку знака арифметичної дії у вікні **Построителя выражений**;
  - вибрати в переліку полів запиту наступне поле та натиснути кнопку **Вставить**;
  - натиснути <**ОК**>. У бланку запиту замість слова **Выражение1**, запропонованого програмою для підпису розрахункового поля, ввести назву поля.

Послідовність створення запиту.

1. Вибрати об’єкт **Запросы.** Створити запит на базі таблиць «Менеджер», «Продукція», «Регіон», «Продаж».
2. Вибрати вкладку **Создание** команду **Конструктор запросов.**
3. На екрані з’являється вікно **Добавление таблицы**. Приводиться перелік усіх таблиць бази даних. Додати в запит всі таблиці, натискаючи кнопку **Добавить.**
4. Ввести в структуру запиту:
   - поле „ПІБ” – таблиці «Менеджер»;
   - поле „Назва продукції” – таблиці «Продукція»;
   - поле „Рік виробництва”, поле „Кількість”, поле „Дата продажу”, поле „Вартість” – таблиці «Продаж»;
   - поле „Назва регіону”– таблиці «Регіон».

   Для цього двічі клацнемо по іменах тих полів, що мають бути в бланку запиту. Поля необхідно обирати в тому порядку, в якому вони мають з’явитися в результатах запиту; при помилковому вводі поле можна знищити, якщо його виділити і натиснути клавішу **Delete**
5. Впорядкувати запит по полю „ПІБ”. Для цього потрібно встановити режим **Сортировка по возрастанию**.
6. Зберегти запит з ім’ям „ПродажПродукції”.

Індивідуальні завдання для створення запитів.

Виконується відповідно до варіанта заданого викладачем. За допомогою запитів вибрати відповідну інформацію.

1. Скільки всього продано пшениці?
2. Хто продає продукцію в Харківську область?
3. Скільки соняшнику продала фірма за всі роки?
4. Визначити вартість продукції, проданої у Львівську область?
5. Яку продукцію продав Щедров В.Ф.?
6. Скільки жита продала фірма?
7. Яка вартість усього проданого соняшнику?
8. З якими регіонами працює Іванченко Н.С.?
9. Які менеджери продавали ячмінь?
10. Скільки пшениці продано в Харківську область?
11. Вартість продукції, проданої в регіони?
12. На яку суму продано жита?
13. Скільки пшениці продано в Київську область?
14. Скільки ячменю продали менеджери в усі регіони?
15. Яка область купує овес?
16. Визначити вартість проданого жита?
17. Хто продає продукцію в в Київську область?
18. Скільки продукції продали менеджери?
19. Яка вартість усього проданого жита?
20. З якими регіонами працює Федірко С.Г.?

Створення звітів.

1. Створити звіт „Щомісячні обсяги продаж по кожному регіону та по фірмі в цілому”.
2. Вибрати вкладку **Создание** команду **Мастер отчетов.** У діалоговому вікні **Создание отчетов** вибрати запит „ПродажПродукції”.
3. На екрані наводиться таблиця з переліком доступних полів. Вибрати поля, що входитимуть у звіт за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее.**
4. Звіт представити у вигляді таблиці «Продаж».
5. Вибрати рівень групування за полем „Дата продажу”, ”Назва регіону”.
6. Вибрати сортування записів по зростанню за полем „Назва продукції”.
7. За допомогою копки **Итоги** визначити суму за полем „Кількість”.
8. Вибрати макет звіту, кнопка **Далее.**
9. Вибрати стиль звіту, кнопка **Далее.**
10. Задайте назву звіту, кнопка **Готово.**
11. Для звіту „ОбсягПродаж” у режимі **„Конструктора”** відредагувати назви полів.

ПРАКТИЧНА РОБОТА № 2. Робота з реляційною базою даних MS ACCESS

Сформувати базу даних, використавши наступну інформацію.

В проектах використовуються деталі, які постачаються декількома постачальниками. Один постачальник постачає деталі лише певного виду. Певна деталь може входити до складу лише одного проекту. Термін реалізації проекту не більше місяця. Спроектувати реляційну базу даних та привести її до нормальної форми.

Коди проектів, дати їх початку та закінчення:

Сирена-1 01.03.2021 - 28.03.2021;
Сирена-2 29.03.2021 - 18.04.2021;
Сирена-3 01.06.2021 - 30.06.2021.

Назва деталі	Назва проекту	Число деталей на проект	Ціна деталі	Постачальник
Конденсатор ПМ	Сирена-1	2	12,57	АТ "Каскад"
Конденсатор КЄ-1	Сирена-2	3	13,00	АТ "Каскад"
Конденсатор МГБ	Сирена-3	4	15,67	АТ "Каскад"
Трансформатор У10х10	Сирена-1	1	258,4	АТ "Квазар"
Трансформатор У20х20	Сирена-2	1	278,9	АТ "Квазар"
Трансформатор У 10х30	Сирена-3	1	300,3	АТ "Квазар"
Дросель СБ-2А	Сирена-1	10	3,56	СП "Сігма"
Дросель СБ-ЗА	Сирена-2	10	2,78	СП "Сігма"
Дросель СБ-4А	Сирена-3	10	5,00	СП "Сігма"
ДіодД2Е	Сирена-1	100	37,60	МП "Оріон"
ДіодД2Ж	Сирена-2	100	38,89	МП "Оріон"
ДіодД2К	Сирєна-3	100	40,98	МП "Оріон"
Шайба №1	Сирена-1	250	1,6	ТОВ ІМПЕК
Шайба №2	Сирена-2	250	1,1	ТОВ ІМПЕК
Шайба №3	Сирена-3	250	1,2	ТОВ ІМПЕК
Тріод ГІ1 ЗА	Сирена-1	20	89,00	МП "КОРСО"
Тріод П4Б	Сирена-2	20	100,00	МП "КОРСО"
Тріод П406	Сирена-3	20	125,30	МП"КОРСО"
Плата 1	Сирена-1	3	1020,00	ТОВ Сіріус
Плата 2	Сирена-2	3	1035,00	ТОВ Сіріус
Плата 3	Сирена-3	3	1200,00	ТОВ Сіріус
Реле-Р1	Сирена-1	2	234,00	ТОВ "Альба"
Реле-Р2	Сирена-2	2	249,00	ТОВ "Альба"
Реле-РЗ	Сирена-3	2	26700	ТОВ "Альба"

Структура БД

Таблиця «**Деталь**»

Ім'я поля	**Тип даних**	**Довжина або розмір**	**Індексне поле**
Код деталі	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Назва деталі	**Текстовый**	20	**Нет**

Таблиця «**Проект**»

Ім'я поля	**Тип даних**	**Довжина або розмір**	**Індексне поле**
Код проекту	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Назва проекту	**Текстовый**	20	**Нет**
Дата початку	**Дата/время**	**Краткий формат даты**	**Нет**
Дата кінця	**Дата/время**	**Краткий формат даты**	**Нет**

Таблиця «**Постачальник**»

Ім'я поля	**Тип даних**	**Довжина або розмір**	**Індексне поле**
Код постачальника	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются**
Назва постачальника	**Текстовый**	30	**Нет**
Адреса	**Текстовый**	50	**Нет**
Код МФО	**Текстовый**	15	**Нет**
Номер рахунку в банку	**Числовой**	**Целое**	**Нет**

Таблиця «**Поставка**»

Ім'я поля	**Тип даних**	**Довжина або розмір**	**Індексне поле**
Код поставки	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Код проекта	**Числовой**	**Длинное целое**	**Нет**
Код деталі	**Числовой**	**Длинное целое**	**Нет**
Число деталей на проект	**Числовой**	**Длинное целое**	**Нет**
Ціна деталі	**Числовой**	**Двойное с плавающей точкой**	**Нет**
Код постачальника	**Числовой**	**Длинное целое**	**Нет**
Деталі фактично поставлені	**Числовой**	**Целое**	**Нет**

Послідовність виконання роботи.

Створення таблиці.

Для того, щоб завантажити **Microsoft Access**, у головному меню натискуємо кнопку **Пуск**, обираємо команду **Программы,** потім **Microsoft Access**

1. В оболонці **Microsoft Access**, у головному меню **Файл** оберемо команду **Создать**.
2. У діалоговому вікні **Создание** виділимо піктограму **Новая база данных** і натискуємо кнопку **ОК.**
3. Зберігаємо базу даних за допомогою команди **Сохранить базу даных как** з ім'ям **«Деталі»** у своїй папці.
4. Оберемо вкладку **Создать** команда **Конструктор таблиц**.
5. У діалоговому вікні **Таблица** вводимо імена полів у стовпчик.
6. У групі **Свойства поля** встановимо властивості полів.
7. Для текстових полів у рядку **Размер поля** задаємо максимальну кількість символів, яке може містити поле.
8. Для поля типу **Дата/время** задати маску введення **00.00.00** і формат поля **- Краткий формат даты**
9. Для тих полів, відсутність даних в яких лишає запис сенсу, встановимо опцію **Да** в полі **Обязательное поле**.
10. Створюємо структури всіх таблиць бази даних **«Деталі»,** заповнюємо таблиці **«Деталь»**, **«Проект»**, **«Постачальник»**

даними.

11. Працюючи з базою даних, зручно використовувати списки можливих значень поля. Оскільки такі списки мають багато пунктів, то для їх створення і поповнення використовуємо окремі таблиці. Для таблиц «**Поставка»** у режимі **Конструктора** створимо можливість обирати значення полів **Код проекту**, **Код деталі, Код постачальника** зі списку можливих значень, які попередньо введені в таблиці **«Деталь»**, «**Проект**», «**Постачальник**».
12. Для цього:
    - встановимо курсор в одне із названих полів, і в групі **Свойства поля**
    - оберемо вкладку **Подстановка;**
    - у списку **Тип элемента управления** оберемо **Список;**
    - **у** списку **Тип источника строк** оберемо **Таблица/Запрос;**
    - у списку **Источник строк** вибираємо потрібну таблицю;
    - у полі **Присоединённый столбец** задамо 1;
    - у полі **Число столбцов** задамо 2;
    - у полі **Заглавия столбцов** оберемо **нет**.
13. У діалоговому вікні бази даних обираємо вкладку **Таблицы,** натискаємо кнопку **Открыть** і заповнюємо таблицю **«Поставка»** даними.

Створення схеми даних.

1. Вибрати вкладку **Работа с базами даннях** команда **Схема данных**. Створити схему даних для роботи з таблицями. Для створення схеми даних потрібно вибрати таблиці. В діалоговому вікні **„Добавление таблицы’’** найти таблиці, для яких створюється схема даних.
2. На екрані з’являється вікно **„Добавление таблицы’’**. Взяти закладку **„Таблицы’’** з переліком таблиць, за допомогою кнопки **„Добавить”** вибрати таблиці, вийти з діалогового вікна, клацнувши мишею по кнопці **„Закрыть”.**
3. За допомогою миші створити зв’язки між таблицями:
   таблиця **«Деталь»** поле «***Код деталі***» → таблиця **«Поставка»** поле «***Код деталі***»;
   таблиця **«Проект»** поле «***Код проекту***» → таблиця **«Поставка»** поле «***Код проекту***»;
   таблиця **«Постачальник»** поле «***Код постачальника***» → таблиця **«Поставка»** поле «***Код постачальника***».

   Підводимо покажчик миші до потрібного поля, натискуємо ліву клавішу і перетягуємо покажчик до такого ж поля у зв’язаній таблиці, клацаємо і з’являється діалогове вікно **Изменение связей**. У цьому вікні встановити прапорець **Обеспечение целостности данных**, натиснути кнопку **Создать**.
4. Для зміни зв’язку використовується права клавіша миші. В меню, що з’являється після натискання правої клавіші, пункт **Изменить связь -** для зміни зв’язку, пункт **Удалить -** для знищення зв’язку.
5. Після встановлення зв’язків закрити схему даних.

Створення форми.

Вибрати об’єкт **Формы.** Створити форми для роботи з таблицями **«Деталь», «Проект», «Постачальник», «Поставка».**

1. Вибрати вкладку **Создание** команда **Мастер форм** і таблицю для якої створюється форма.
2. На екрані з’являється вікно **Создание форм**. Наводиться таблиця з переліком доступних полів. Вибрати поля, що будуть входити у форму за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее.**
3. Вибрати вид оформлення форми, кнопка **Далее.**
4. Вибрати стиль оформлення форми, кнопка **Далее.**
5. Задайте назву форми, кнопка **Готово.**
6. У таблиці **«Поставка»** виділимо поле зі списком, у контекстному меню оберемо команду **Преобразовать элемент в…,** а потім **Поле.**
7. Для перейменування напису двічі клацнемо по ньому і введемо новий текст.
6. Натиснемо кнопку **Выбор объектов** на **Панели элементов** і протягнемо покажчик по формі **«Поставка»**.
7. Для переміщення групи виділених полів установимо покажчик на виділеному полі (він приймає вигляд розкритої долоні), і перетягнемо поле в необхідне місце.
8. Для змінення розміру поля встановимо покажчик на один із семи маркерів виділеного поля і перетягнемо його.
9. Натиснемо на кнопку **Надпись** на **Панели элементов**, протягнемо покажчик по формі і введемо заголовок форми.
9. Виділимо всі поля, натиснемо **Свойства** контекстного меню в діалоговому вікні **Несколько элементов управления** відкриємо вкладку **Макет,** у полі **Цвет фона** оберемо необхідний колір, у полі **Цвет текста** оберемо необхідний колір.
10. Зберігаємо форму.

Створення запиту

За допомогою **Конструктора** створимо **Запит**1 для підрахунку вартості кожної партії деталей.

Для того, щоб розробити запит за допомогою **Конструктора,** необхідно виконати такі дії:

1. Вибрати об'єкт **Запросы.**
2. Вибрати вкладку **Создание** команду **Конструктор запросов.**
3. На екрані з'являється вікно **Добавление таблицы**. Приводиться перелік усіх таблиць бази даних. Додати в запит всі таблиці, натискаючи кнопку **Добавить.**
4. Натиснемо кнопку **Закрыть**, щоб зняти з екрана діалогове вікно **Добавление таблицы**.
5. Двічі клацнемо по іменах тих полів, що мають бути в бланку запиту **(*Назва проекту, Назва деталі, Число деталей на проект, Ціна деталі, Деталі фактично поставлені, Назва постачальника)*.** Поля необхідно обирати в тому порядку, в якому вони мають з'явитися в результатах запиту, за помилкового введення, поле можно зняти, якщо його виділити і натиснути клавішу **Delete**.
6. Для створення розрахункового поля *Вартість* слід:
   - встановити курсор у бланку запиту у порожній стовпчик та натиснути піктограму **Построитель выражений** на панелі інструментів;
   - у полі **Построителя выражений** набрати розрахунковий вираз: =[*Ціна деталі*]*[*Деталі фактично поставлені*], для цього:
   - вибрати в переліку полів, вибраних до запиту, поле *Ціна деталі* та натиснути кнопку **Вставить**;
   - натиснути кнопку знака множення у вікні **Построителя выражений**;
   - вибрати в переліку полів запиту *Деталі фактично поставлені* та натиснути кнопку **Вставить**;
   - натиснути <**ОК**>. У бланку запиту замість слова **Выражение1:**, запропонованого програмою для підпису розрахункового поля, ввести **Вартість:.**
7. Далі створюємо **Запит2** на основі **Запиту1,** в бланку запиту повині бути поля **(*Назва проекту, Вартість)*** для визначення скільки коштів витрачено на кожний проект.
8. На додатковій вкладці **Работа с запросами** обираємо команду **Итоги**
9. У полі **Групповые операции** оберемо:
   a. операцію **Группировка** для поля ***Назва проекту***;
   b. операцію **Sum** для поля ***Вартість***.
10. У полі **Сортировка** для поля ***Назва проекту*** обираємо **По возрастанию**.
11. Для відбору окремих записів, використовуємо рядок **Условия отбора** для конкретного стовпчика.
12. Зберігаємо запит.

13. Створити запити, які дозволять:

а) надати інформацію про постачальників деталей для певного проекту, вказавши основні атрибути постачальника (адресу, назву постачальника, код МФО, номер рахунку в банку).

б) визначити постачальників, що не поставили деталі, вказавши, які проекти не повністю забезпечені необхідними деталями.

Створення звіту

1. Створюємо звіт **«Інформація по проектам»**.
2. Вибрати вкладку **Создание** команду **Мастер отчетов.** У діалоговому вікні **Создание отчетов** вибрати **Запит**1.
3. На екрані з'являється вікно **Создание отчетов**. Наводиться таблиця з переліком доступних полів.
4. Вибрати поля, що входитимуть у звіт за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее>.**
5. Вибрати вид представлення звіту по таблиці **«Поставка»**.
6. Вибрати рівень групування за полем **«Назва проекту»**.
7. Вибрати сортування записів по зростанню за полем **«Назва деталі»**.
8. За допомогою копки **Итоги** визначити суму за полем **«Вартість»**.
9. Вибрати макет звіту, кнопка **Далее>.**
10. Вибрати стиль звіту, кнопка **Далее>.**
11. Задайте назву звіту, кнопка **Готово.**
12. Для звіту **«Інформація по проектам»** у режимі **Конструктора** відредагувати назви полів.
13. Зберігаємо звіт.

ПРИМІТКА. Якщо на будь-якому кроці майстра зроблено невірну дію, то її можна відмінити за допомогою кнопки **Назад** і повернутися на один крок назад або за допомогою кнопки **Отмена** зовсім відмовитися від послуг майстра.

ПРАКТИЧНА РОБОТА № 3. Основні принципи створення бази даних у MS ACCESS

Завдання

1.Розробіть базу даних за структурою, що відповідає вашому варіанту. Введіть 20 записів.

2.Розробіть форму для введення даних у таблицю.

3.Створіть запит відповідно до вашого варіанту. На базі створеного запиту розробіть форму для відображення результатів.

4.Створіть звіт відповідно до вашого варіанту.

Теоритичні відомості

Створення таблиці

1. В оболонці **Microsoft Access**, у головному меню **Файл** оберемо команду **Создать базу данных** (можно натиснути кнопку **Создать базу данных** на панелі інструментів).
2. У діалоговому вікні **Создание** виділимо піктограму **Новая база данных** і натискуємо кнопку **ОК.**
3. У діалоговому вікні **Файл** нової бази даних у вікні **Имя файла** введемо ім'я бази даних, у вікні **Папка** оберемо свою папку і натиснемо кнопку **Создать.**
4. У діалоговому вікні бази даних оберемо вкладку **Таблицы** і натискуємо кнопку **Создать.**
5. У діалоговому вікні **Новая таблица** оберемо вкладку **Конструктор** і натискуємо кнопку **ОК.**
6. У діалоговому вікні **Таблица1** вводимо імена полів у стовпчик.
7. У групі **Свойства поля** встановимо властивості полів.
8. Для текстових полів у рядку **Размер поля** задаємо максимальну кількість символів, які може містити поле.
9. Зберігаємо таблицю **Таблица1** з унікальним іменем. Потім у діалоговому вікні бази даних обираємо вкладку **Таблицы,** натискуємо кнопку **Открыть** і заповнюємо таблицю даними.

Створення форми

Для того, щоб розробити форму за допомогою **Мастера форм,** необхідно виконати такі дії:

1. У вікні бази даних відкриємо вкладку **Формы** і натиснемо кнопку **Создать.**
2. У діалоговому вікні **Новая форма** оберемо **Мастер форм,** необхідну таблицю**,** натискуємо кнопку **ОК**.
3. Зберігаємо форму.

Створення запиту

Для того, щоб розробити запит за допомогою **Конструктора,** необхідно виконати такі дії:

1.У вікні бази даних відкриємо вкладку **Запросы** і натиснемо кнопку **Создать.**

2.У діалоговому вікні **Новый запрос** оберемо **Конструктор** і натиснемо кнопку **ОК.**

3.У діалоговому вікні **Добавление таблицы** відкриємо вкладку **Таблицы и запросы.**

4.Двічі клацнемо по імені таблиці для того, щоб ввести її у запит.

5.Натиснемо кнопку **Закрыть**, щоб зняти з екрана діалогове вікно **Добавление таблицы.**

6.Двічі клацнемо по іменах тих полів, що мають бути в бланку запиту. Поля необхідно обирати в тому порядку, в якому вони мають з'явитися в результатах запиту, за помилкового введення, поле можно зняти, якщо його виділити і натиснути клавішу **Delete**.

7. Для створення розрахункового поля слід:

- установити курсор у бланку запиту в порожній стовпчик та натиснути піктограму **Построитель выражений** на панелі інструментів;
- у полі **Построителя выражений** набрати розрахунковий вираз, для цього:
  вибрати в переліку полів, вибраних до запиту, необхідне поле та натиснути кнопку **Вставить**;
  натиснути кнопку арифметичного знака у вікні **Построителя выражений**;
  натиснути <**ОК**>.
- у бланку запиту замість слова **Выражение1:**, запропонованого програмою для підпису розрахункового поля, ввести необхідну назву поля.

8. Для створення підсумків у меню **Вид** обираємо команду **Групповые операции** (можно натиснути кнопку **Групповые операции** на панелі інструментів для того, щоб у вікні запиту з'явилося поле **Групповая операция**).

9.У полі **Групповые операции** оберемо:

- операцію **Группировка** для текстового поля;
- операцію **Sum** для числового поля;

10. Для відбору окремих записів використовуємо рядок **Условия отбора** для конкретного поля.

11. Зберігаємо запит.

12. У меню **Запрос** оберемо команду **Открыть.**

Створення звіту

Для того, щоб розробити звіт вибираємо об'єкт **Очеты.**

1.Вибрати режим **Создать.** У діалоговому вікні **Новый отчет** вибрати режим **Мастер отчетов,** таблицю або запит.

2.На екрані з'являється вікно **Создание отчетов**. Наводиться таблиця з переліком доступних полів.

3.Вибрати поля, що входитимуть у звіт за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее>.**

4. Вибрати вид подання звіту.

5.Вибрати необхідний рівень групування за полем.

6.Вибрати сортування записів.

7.За допомогою копки **Итоги** визначити необхідну суму.

8.Вибрати макет звіту, кнопка **Далее>.**

9.Вибрати стиль звіту, кнопка **Далее>.**

10.Задайте назву звіту, кнопка **Готово.**

Варіант 1

Розробіть базу даних **Торгівля** з такою структурою:

Рік, Квартал, Країна, Сума експорту, Сума імпорту

за допомогою **Конструктора**.

Для таблиці **Торгівля** за допомогою **Конструктора** створіть запит для підрахунку сумарного обсягу експорту й імпорту по кожній країні.

Для таблиці **Торгівля** створіть звіт, у якому наводяться дані про обсяг експорту й імпорту, згруповані по кожній країні й відсортовані за алфавітом у порядку зростання.

Варіант 2

Розробіть базу даних **МТС** з такою структурою:

Рік, Назва господарства, Площа, Вид машин, Норма

за допомогою **Конструктора.**

Для таблиці **МТС** за допомогою **Конструктора** створіть запит з полями вихідної таблиці і додатковим полем **Загальна потреба**, значення якого обчислюється за формулою: **(Площа * Норма)/1000**

для підрахунку загальної площі угідь по всіх господарствах і суми загальної потреби в машинах на рік.

На базі запиту створіть звіт, у якому наводяться дані про суму загальної потреби в машинах за роками.

Варіант 3

Розробіть базу даних **Цех** з такою структурою:
Місяць, Цех, **Прізвище, Ім'я, По батькові, Стать, Заробітна плата, Премія** за допомогою **Конструктора.**

Для таблиці **Цех** за допомогою **Конструктора** створіть запит для підрахунку кількості працівників у кожному цеху, фонду заробітної плати і преміального фонду цеху.

Для таблиці **Цех** створіть звіт, у якому наводяться дані працівників, згруповані за цехами і відсортовані за алфавітом у порядку зростання.

Варіант 4

Розробіть базу даних **Витрати пального** з такою структурою:
Місяць, Код палива, Найменування палива, Кількість, Сума за допомогою **Конструктора**.

Для таблиці **Витрати пального** за допомогою **Конструктора** створіть запит для підрахунку загальної кількості і суми витрат пального за місяцями.

Для таблиці **Витрати пального** створіть звіт, у якому наводяться дані про сумарні витрати пального, згруповані за найменуваннями і відсортовані за алфавітом у порядку зростання.

Варіант 5

Розробіть базу даних **Будматеріали** з такою структурою:
Місяць, Код матеріалу, **Найменування матеріалу, Кількість, Сума** за допомогою **Конструктора**.

Для таблиці **Будматеріали** за допомогою **Конструктора** створіть запит для підрахунку загальної кількості і суми витрат будматеріалів по місяцях.

Для таблиці **Будматеріали** створіть звіт, у якому наводяться дані про сумарні витрати будматеріалів, згруповані по найменуванню і відсортовані за алфавітом у порядку зростання.

Варіант 6

Розробіть базу даних **Студенти** з такою структурою:
Факультет, Курс, Група, Прізвище, Ім'я, По батькові, Стать, Успішність, Стипендія за допомогою **Конструктора.**

Для таблиці **Студенти з**а допомогою **Конструктора** створіть запит для підрахунку загальної кількості студентів і суми виплати стипендії за факультетами.

Для таблиці **Студенти** створіть звіт, у якому наводяться дані про студентів, згруповані за факультетами і відсортовані за алфавітом у порядку зростання.

Варіант 7

Розробіть базу даних **Агрофірма** з такою структурою:
Місяць, Прізвище, Ім'я, По батькові, Посада, Категорія, Оклад, Премія за допомогою **Конструктора.**

Для таблиці **Агрофірма** за допомогою **Конструктора** створіть запит для підрахунку кількості співробітників, суми заробітку і премії за посадами.

Для таблиці **Агрофірма** створіть звіт, у якому наводяться дані співробітників, згруповані за посадами і відсортовані за алфавітом у порядку зростання.

Варіант 8

Розробіть базу даних **Товари** з такою структурою:
Код товару, Найменування товару, Дата продажу, Кількість, Ціна, Процентна ставка за допомогою **Конструктора.**

Для таблиці **Товари** за допомогою **Конструктора** створіть запит з полями вихідної таблиці і додатковим полем **Сума до виплати**, значення якого обчислюється за формулою:
Кількість * Ціна * (1-процентна ставка/100)
для підрахунку загальної кількості й суми до виплати за датою продажу.

На базі запиту створіть звіт, у якому наводяться дані про сумарні виплати за товарами, згруповані за найменуванням і відсортовані за алфавітом у порядку зростання.

Варіант 9

Розробіть базу даних **Облік праці** з такою структурою:
Цех, Місяць, Прізвище, Ім'я, По батькові, Код деталі, Кількість деталей, Розцінка за допомогою **Конструктора**

Для таблиці **Облік праці** за допомогою **Конструктора** створіть запит з полями вихідної таблиці і додатковим полем **Заробіток**, значення якого обчислюється за формулою: **Кількість деталей * Розцінка**
для підрахунку загальної кількості деталей і заробітку за місяцями.

На базі запиту створіть звіт, у якому наводяться дані про сумарну кількість зроблених деталей по цехах за місяць.

Варіант 10

Розробіть базу даних **План випуску** з такою структурою:
Цех, Місяць, Код виробу, Кількість виробів, Назва матеріалу, Кількість матеріалу за допомогою **Конструктора.**

Для таблиці **План випуску** за допомогою **Конструктора** створіть запит з полями вихідної таблиці й додатковим полем **Потреба в матеріалах**, значення якого обчислюється за формулою:
Кількість виробів * Кількість матеріалу
для підрахунку загальної потреби в матеріалах за місяцями.

На базі запиту за допомогою **Конструктора** створіть звіт, у якому наводяться дані про сумарну потребу в матеріалах цехів за місяць.

Варіант 11

Розробіть базу даних **Реалізація продукції** з такою структурою: **Місяць, Найменування продукції, Обсяг реалізації, Ставка податку** за допомогою **Конструктора.**

Для таблиці **Реалізація продукції** за допомогою **Конструктора** створіть запит з полями вихідної таблиці й додатковим полем **Сума податку**, значення якого обчислюється за формулою:

Обсяг реалізації * Ставка податку

для підрахунку загальної суми податку з обігу за місяць.

На базі запиту створіть звіт, у якому наводяться дані про сумарний обсяг реалізації продукції й загальну суму податку з обороту за місяць.

Варіант 12

Розробіть базу даних **Прибуток підприємств** з такою структурою: **Місяць, Назва підприємства, Вид діяльності, Прибуток, Відсоток податку** за допомогою **Конструктора.**

Для таблиці **Прибуток підприємств** за допомогою **Конструктора** створіть запит з полями вихідної таблиці й додатковим полем **Податок на прибуток** (розраховується як відсоток від прибутку, 20%) для підрахунку загальної суми прибутку і податку на прибуток за місяць.

На базі запиту створіть звіт, у якому наводяться дані про сумарний обсяг прибутку і загальну суму податку на прибуток за місяць за підприємствами.

Варіант 13

Розробіть базу даних **Табель** з такою структурою:

Місяць, Табельний номер, Прізвище, Ім'я, По батькові, Кількість робочих днів у місяці, Кількість відпрацьованих днів, Оклад за допомогою **Конструктора.**

Для таблиці **Табель** за допомогою **Конструктора** створіть запит з полями вихідної таблиці і додатковим полем **Заробіток**, значення якого обчислюється за формулою:

(Оклад/Кількість робочих днів у місяці) * Кількість відпрацьованих днів для підрахунку загальної суми нарахованого заробітку по місяцях.

На базі запиту створіть звіт, у якому наводяться дані про сумарне нарахування заробітку за місяцями.

Варіант 14

Розробіть базу даних **Облік відпрацьованого часу** з такою структурою: **Місяць, Прізвище, Ім'я, По батькові, Професія, Розряд, Відпрацьовано годин, Тарифна ставка** за допомогою **Конструктора.**

Для таблиці **Облік відпрацьованого часу** за допомогою **Конструктора** створіть запит з полями вихідної таблиці і додатковим полем **Заробіток працівника**, значення якого обчислюється за формулою:

Тарифна ставка * Відпрацьовано годин

для підрахунку загальної суми заробітку працівника за місяцями.

На базі запиту створіть звіт, у якому наводяться дані про сумарний заробіток працівника за місяцями.

Варіант 15

Розробіть базу даних **Ремонт автомобілів** з такою структурою: **Прізвище, Ім'я, По батькові, Марка автомобіля, Код деталі, Вид ремонту, Розцінка** за допомогою **Конструктора.**

Для таблиці **Ремонт автомобілів** за допомогою **Конструктора** створіть запит для підрахунку кількості зроблених ремонтів і загальної вартості ремонту автомобіля (вартість ремонту розраховується як сума розцінок усіх видів ремонту з врахуванням 10% ПДВ).

Для таблиці **Ремонт автомобілів** створіть звіт, у якому наводяться дані про власників автомобілів, згруповані за марками автомобіля і відсортовані за алфавітом у порядку зростання.

Варіант 16

Розробіть базу даних **Картотека** квартир з такою структурою: **Номер картки, Район, Телефон, Кількість кімнат, Загальна площа. Поверх, Вулиця, Вартість** за допомогою **Конструктора.**

Для таблиці **Картотека квартир** за допомогою **Конструктора** створіть запит для підрахунку кількості квартир у кожному районі і сумарній вартості житла по району.

Для таблиці **Картотека квартир** створіть звіт, у якому наводяться дані квартир, згруповані по районах і відсортовані за алфавітом у порядку зростання.

Варіант 17

Розробіть базу даних **Завод** з такою структурою:

Цех, Місяць, Прізвище, Посада, Категорія, Оклад, Премія за допомогою **Конструктора.**

Для таблиці **Завод** за допомогою **Конструктора** створіть запит для підрахунку кількості співробітників, суми заробітку і премії за цехами.

Для таблиці **Завод** створіть звіт, у якому наводяться дані співробітників, згруповані за цехами і відсортовані за алфавітом у порядку зростання.

Варіант 18

Розробіть базу даних **Університет** з такою структурою:

Факультет, Група, Прізвище, Успішність, Стипендія за допомогою **Конструктора.**

Для таблиці **Університет з**а допомогою **Конструктора** створіть запит для підрахунку загальної кількості студентів і суми виплати стипендії за факультетами.

Для таблиці **Університет** створіть звіт, у якому наводяться дані про студентів, згруповані за факультетами і відсортовані за алфавітом у порядку зростання.

Варіант 19

Розробіть базу даних **Ремонт** з такою структурою:

Марка автомобіля, Назва підприємсива, Код деталі, Вид ремонту, Кількість, Розцінка за допомогою **Конструктора.**

Для таблиці **Ремонт** за допомогою **Конструктора** створіть запит за формулою: **Кількість* Розцінка**

для підрахунку кількості зроблених ремонтів і загальної вартості ремонту автомобіля.

Для таблиці **Ремонт** за допомогою **Конструктора** створіть звіт, у якому наводяться дані про підприємства власники автомобілів, згруповані за марками автомобіля і відсортовані за алфавітом у порядку зростання.

Варіант 20

Розробіть базу даних **Торгівля** з такою структурою:

Магазин, Найменування продукції, Обсяг реалізації, Ставка податку за допомогою **Конструктора.**

Для таблиці **Торгівля** за допомогою **Конструктора** створіть запит з полями вихідної таблиці й додатковим полем **Сума податку**, значення якого обчислюється за формулою:

Обсяг реалізації * Ставка податку

для підрахунку загальної суми податку з обігу за магазинами.

На базі запиту створіть звіт, у якому наводяться дані про сумарний обсяг реалізації продукції й загальну суму податку з обороту за магазинами.

ПРАКТИЧНА РОБОТА № 4. Створення бази даних «Земля»

Виконати наступні завдання:

1. Сформувати базу даних, розробити такі об'єкти: таблиці, форми, запити, звіти.

2. Розробляючи форми, дані що вводяться в форму вибірковим чином, оформити у вигляді списків і полів зі списками. Для цього при проектуванні таблиці:

- встановлюємо курсор в одне із названих полів, і в групі **Свойства поля** оберемо вкладку **Подстановка;**
- у списку **Тип элемента управления** обираємо **Список;**
- **у** списку **Тип источника строк** обираємо **Список значений;**
- у рядку **Источник строк** набираємо можливі значення поля, розділені ";"

Для виконання завдання використовуємо інформацію наступної структури:

Господарство_____

Село (с.м.т., місто)_____Область_____

Картка земельної ділянки (сінокосу) _____

Код господарства_____Район_____

Ідентифікаційний код_____ Площа зем. ділянки, га _____

Сінокіс: заливний, суходільний, заболочений;

не поліпшений, поверхнево поліпшений, докорінно поліпшений (потрібне підкреслити).

Середня врожайність за останні 3 роки, ц/га _____

Вміст гумусу в грунті, %:_____Назва грунту орн. шарі, _____

Ділянка зрошувана, осушена, з подвійним регулюв. вологи (потрібне підкреслити). Рівень забезпечення поливних норм, % _____
Частка перезвол. ріллі, % _____

Солонцюватість: несол.(зал.сол.); сл. сол.; сер. сол.: сильн. сол. (потрібне

підкреслити).

Площа еродов. ріллі, га_____, в т.ч. сл. ер.___; сер. ер.____; дужеер._____.

Хім.забруднення, мг/кг грунту: мідь___; ртуть_____; кадмій_____; свинець_____; цинк_____.

Рад. забруднення, кю/км^2: цезій-137_____; стронцій-90_____;

Група поля _____

Розташування земельної ділянки на території району (міста):

центр; пн.; пн.-сх.; сх.; пд.-сх.; пд.; пд.- зах.; зах.; пн. - зах. (потрібне підкреслити).

Назва нас. пункту, до якого відноситься зем. ділянка _____

Чисельність жителів цього населеного пункту, чол. _____

Загальна площа с.-г. угідь на прилеглій до цього населен. пункту території, га _____.

Відстань земельної ділянки від нас. пункту, км_____. Стан дороги, щозв'язує зем. ділянку з нас. пунктом: незадов.; задов.; хороший; відмінний (потрібне підкреслити).

Відстань нас. пункту до районного центру, км_____

Чисельність жителів районного центру, тис. чол. _____

Відстань нас. пункту до найближчого міста (с.м.т.), км_____

Назва найближчого міста (с.м.т.)_____

Чисельність жителів найближчого міста (с.м.т.), тис. чол. _____

Відстань нас. пун. до зал. станції, км_____, порту (пристані),км _____

Група доріг до: райц._____станц._____найбл. м. (с.м.т.)_____

зал._____порту (прист.)_____

Примітка:_____

Землевпорядник району (міста)_____(прізв. таініц.)

Гол. агроном, райуправління с.г_____(прізв. та ініц.)

Керівник господарства _____ (прізв. та ініц.)

Гол. агроном господарства _____ (прізв. та ініц.)

Структура БД

Таблиця «**Область**»

Ім’я поля	Тип даних	Довжинна або розмір	Індексоване поле
Код області	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Назва області	**Текстовый**	**20**	**Нет**

Таблиця **«Район»**

Ім’я поля	Тип даних	Довжинна або розмір	Індексоване поле
Код району	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Назва району	**Текстовый**	**20**	**Нет**
Чисельність жителів рай.центру, тис. чол.	**Числовой**	**Длинное целое**	**Нет**
Землевпорядник району,міста (ПІБ)	**Текстовый**	**50**	**Нет**
Гол.агроном району (ПІБ)	**Текстовый**	**50**	**Нет**

Таблиця **«Господарство»**

Ім’я поля	Тип даних	Довжинна або розмір	Індексоване поле
Код господарства	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Назва господарства	**Текстовый**	**20**	**Нет**
Керівник господарства (ПІБ)	**Текстовый**	**50**	**Нет**
Гол. агроном господарства(ПІБ)	**Текстовый**	**50**	**Нет**
Код населеного пункту	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Ідентифікаційний код господарства	**Текстовый**	**15**	**Нет**
Код району	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Код області	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Примітка	**Текстовый**	**50**	**Нет**

Таблиця **«Населений пункт»**

Ім’я поля	Тип даних	Довжинна або розмір	Індексоване поле
Код населеного пункту	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Назва населеного пункту	**Текстовый**	**20**	**Нет**
Чисельність жителів, чол.	**Числовой**	**Длинное целое**	**Нет**
Загальна площа с.-г. угідь	**Числовой**	**Длинное целое**	**Нет**
Код району	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Код області	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Відстань нас.пункту до найближчого міста,км	**Числовой**	**Целое**	**Нет**
Назва найближчого міста	**Текстовый**	**20**	**Нет**
Відстань нас.пункту до районого центру,км	**Числовой**	**Целое**	**Нет**
Чисельність жителів найближчого міста	**Числовой**	**Целое**	**Нет**
Відстань до зал. станції	**Числовой**	**Целое**	**Нет**
Група доріг	**Текстовый**	**10**	**Нет**

Таблиця «**Характеристика земельної ділянки**»

Ім’я поля	Тип даних	Довжинна або розмір	Індексоване поле
Код земельної ділянки	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Площа зем. ділянки, га	**Числовой**	**Целое**	**нет**
Сінокіс (тип)	**Текстовый**	10	**нет**
Середня врожайність, ц/га	**Числовой**	**Целое**	**нет**
Вміст гумусу в грунті, %	**Числовой**	**Целое**	**нет**
Назва грунту орн. шарі	**Текстовый**	10	**нет**
Зрошення ділянки	**Текстовый**	20	**нет**
Поливні норми,%	**Числовой**	**Целое**	**нет**
Перезволоження ріллі,%	**Числовой**	**Целое**	**нет**
Солонцюватість	**Текстовый**	10	**нет**
Площа еродов.ріллі	**Числовой**	**Целое**	**нет**
Площа еродов.ріллі ,сл.ер.	**Числовой**	**Целое**	**нет**
Площа еродов.ріллі,сер.ер.	**Числовой**	**Целое**	**нет**
Площа еродов.ріллі,дуже ер.	**Числовой**	**Целое**	**нет**
Хім.забруднення, мг/кг, мідь	**Числовой**	**Целое**	**нет**
Хім.забруднення, мг/кг,ртуть	**Числовой**	**Целое**	**нет**
Хім.забруднення, мг/кг,кадмій	**Числовой**	**Целое**	**нет**
Хім.забруднення, мг/кг,свинець	**Числовой**	**Целое**	**нет**
Хім.забруднення, мг/кг,цинк	**Числовой**	**Целое**	**нет**
Рад.забруднення,кю/км^2,цезій-137	**Числовой**	**Целое**	**нет**
Рад.забруднення,кю/км^2,стронцій-90	**Числовой**	**Целое**	**нет**
Група поля	**Текстовый**	10	**нет**

Таблиця **«Розташування земельної ділянки»**

Ім’я поля	Тип даних	Довжинна або розмір	Індексоване поле
Код	**Счетчик**	**Длинное целое**	**Да(Совпадения не допускаются)**
Код земельної ділянки	**Числовой**	**Длинное целое**	**Да(Совпадения не допускаются)**
Код району	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Код області	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Код населеного пункту	**Числовой**	**Длинное целое**	**Да(Совпадения допускаются)**
Розташування на території району	**Текстовый**	10	**нет**
Відстань земельної ділянки від нас.пункту,км	**Числовой**	**Целое**	**нет**
Стан дороги	**Текстовый**	10	**нет**
Відстань земельної ділянки до рай.центру, км	**Числовой**	**Целое**	**нет**
Назва найближчого міста	**Текстовый**	10	**нет**
Чисельність жителів найближчого міста,тис.чол.	**Числовой**	**Целое**	**нет**

ПРАКТИЧНА РОБОТА № 5. Створення бази даних «Список співробітників»

Створіть базу даних **Список співробітників,** призначену для автоматизації обліку кадрів навчального закладу і розрахунків з оплати праці з такою структурою.

Таблиця **«Список»**

Ім'я поля	**Тип даних**	**Довжина або розмір**	**Індексне поле**
Код	Счетчик	Длинное целое	Да (Совпадения не допускаются)
Прізвище	Текстовый	20	Нет
Ім'я	Текстовый	15	Нет
По батькові	Текстовый	15	Нет
Стать	Текстовый	5	Нет
Прийом на роботу	Дата/время	Краткий формат даты	Нет
Освіта	Текстовый	15	Нет
Організація	Текстовый	20	Нет
Ступінь	Текстовый	10	Нет
Область	Текстовый	15	Нет
Звання	Текстовый	15	Нет
Посада	Текстовый	15	Нет
Відділ	Текстовый	20	Нет
Оклад	Числовой	Длинное целое	Нет
Публікації	Числовой	3	Нет
Адреса	Текстовый	50	Нет
Домашній телефон	Текстовый	10	Нет
Робочий телефон	Текстовый	10	Нет
Сім'я	Логический	Да/Нет	Нет
Діти	Числовой	2	Нет

Створення таблиці

Для того, щоб завантажити **Microsoft Access**, у головному меню натискаємо кнопку **Пуск**, обираємо команду **Программы,** потім **Microsoft Access**

1. В оболонці **Microsoft Access**, у головному меню **Файл** оберемо команду **Создать**.
2. У діалоговому вікні **Создание** виділимо піктограму **Новая база данных** і натискуємо кнопку **ОК.**
3. Зберігаємо базу даних за допомогою команди **Сохранить базу даных как** з ім'ям **«Список співробітників»** у своїй папці.
4. Оберемо вкладку **Создать** команда **Конструктор таблиц**.
5. У діалоговому вікні **Таблица** вводимо імена полів у стовпчик.
6. У групі **Свойства поля** встановимо властивості полів.
7. Для текстових полів у рядку **Размер поля** задаємо максимальну кількість символів, яке може містити поле.
8. Для поля **Прізвище, Ім'я, По батькові** задаємо маску введення, яка перетворює першу букву введеного текста на прописну, інші - на строчні **>?<?????????????**
9. Для поля типу **Дата/время** задати маску введення **00.00.00** і формат поля **- Краткий формат даты**
10. Для поля **Домашній телефон** задаємо маску введення шестизначного номера телефона, розділеного двома дефісами **00-00-00**
11. Для поля **Робочий телефон** задаємо маску введення трьохзначного номера телефона, розділеного дефісом **0-00**
12. Задамо значення за замовчуванням для поля **Стать-** чол., **Освіта** – середня, **Оклад -** 6500, **Публикації -** 0**, Діти -** 0**.**
13. Для тих полів, відсутність даних в яких лишає запис сенсу, встановимо опцію **Да** в полі **Обязательное поле**.
14. Створимо можливість обирати значення полів **Стать**, **Освіта, Ступінь, Область, Звання, Відділ, Діти** зі списку можливих значень, які потрібно попередньо створити. Для цього:

- встановимо курсор в одне із названих полів, і в групі **Свойства поля** оберемо вкладку **Подстановка;**
- у списку **Тип элемента управления** оберемо **Список;**
- **у** списку **Тип источника строк** оберемо **Список значений;**
- у списку **Источник строк** набираємо можливі значення поля, розділені ";"

Поле	Можливі значення
Для поля **Стать** введемо	**чол.; жін.;**
Для поля **Освіта** введемо	**початкова; середня; серед.спец.; вища;**
Для поля **Ступінь** введемо	**канд.; докт.;**
Для поля **Область** введемо	**фіз.мат.наук;хім.наук;екон.наук; істор.наук;біолог.наук;філол.наук; мед.наук**; **тех.наук;**
Для поля **Звання** введемо	**доцент; професор;**
Для поля **Відділ** введемо	**АХО;№1;№5;№6;№15;№20;№25;№26**;
Для поля **Діти** введемо	**0;1;2;3;**

Установити поле **Код** - ключове поле. Збережемо таблицю під ім'ям **Список.**

Працюючи з базою даних, зручно використовувати списки можливих значень поля. Оскільки такі списки мають багато пунктів, то для їх створення і поповнення використовуємо окремі таблиці. Створимо такі таблиці в базі даних **Список співробітників** для полів **Ім'я, По батькові, Організація, Посада** таким чином:

а) створимо таблицю **Імена,** що має одне поле **Ім'я,** заповнюємо її і збережемо на диску. Далі підставляємо таблицю **Імена** в поле **Ім'я.** Для цього:

- відкриємо таблицю **Список** у режимі **Конструктора;**
- установимо курсор в полі **Ім'я**, і в групі **Свойства поля** оберемо вкладку **Подстановка;**
- у списку **Тип элемента управления** оберемо **Список**
- у списку **Тип источника строк** оберемо**Таблица/Запрос**
- у списку **Источник строк** оберемо назву створеної таблиці **Імена**
- у полі **Присоединённый столбец** задамо 1;
- у полі **Число столбцов** задамо 1;
- у полі **Заглавия столбцов** оберемо **нет**.

б) аналогічно створимо і заповнимо списки для полів **По батькові, Організація, Посада.**

Заповнюємо таблицю даними (30 записів).

Створення форми

1. Вибрати вкладку **Создание** команда **Мастер форм** і таблицю для якої створюється форма.
2. На екрані з'являється вікно **Создание форм**. Наводиться таблиця з переліком доступних полів. Вибрати поля, що будуть входити у форму за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее.**
3. Вибрати вид оформлення форми, кнопка **Далее.**
4. Вибрати стиль оформлення форми, кнопка **Далее.**
5. Задайте назву форми, кнопка **Готово.**
6. Відкрити форму за допомогою **Конструктора.**
7. Виділимо поле зі списком, у контекстному меню оберемо команду **Преобразовать элемент в…,** а потім **Поле.**
8. Для перейменування напису двічі клацнемо по ньому і введемо новий текст.
9. Натиснемо кнопку **Выбор объектов** на **Панели элементов** і протягнемо покажчик по формі.
10. Для переміщення групи виділених полів установимо покажчик на виділеному полі (він приймає вигляд розкритої долоні), і перетягнемо поле в необхідне місце.
11. Для змінення розміру поля встановимо покажчик на один із семи маркерів виділеного поля і перетягнемо його.
12. Натиснемо на кнопку **Надпись** на **Панели элементов**, протягнемо покажчик по формі і введемо заголовок форми **АНКЕТА**.
13. Виділимо всі поля, окрім заголовка, натиснемо **Свойства** контекстного меню, в діалоговому вікні **Несколько элементов управления** відкриємо вкладку **Макет** і в полі **Высота** введемо значення 0,5 см.
14. Виділимо всі поля, що мають розташовуватися в одному рядку і введемо в поле **От верхнего края** необхідне значення.
15. Виділимо всі поля, що мають розташовуватися в одному стовпчику і введемо в поле **От левого края** необхідне значення.
16. Виділимо всі поля зі списком, натиснемо **Свойства** контекстного меню в діалоговому вікні **Несколько элементов управления** відкриємо вкладку **Макет,** у полі **Цвет фона** оберемо необхідний колір, у полі **Цвет текста** оберемо необхідний колір.
17. Зберігаємо форму під іменем **АНКЕТА**.

Створення запиту

Для таблиці **Список** за допомогою **Конструктора** створимо запит для підрахунку кількості співробітників у кожному відділі і фонду заробітної плати відділу.

Для того, щоб розробити запит за допомогою **Конструктора,** необхідно виконати такі дії:

1. Вибрати вкладку **Создание** команду **Конструктор запросов.**
2. На екрані з'являється вікно **Добавление таблицы.** Додати в запит таблицю, натискаючи кнопку **Добавить.**
3. Двічі клацнемо по іменах тих полів, що мають бути в бланку запиту **(Відділ**, **Прізвище,Оклад**). Поля необхідно обирати в тому порядку, в якому вони мають з'явитися в результатах запиту, за помилкового введення поле можно зняти, якщо його виділити і натиснути клавішу **Delete**.
4. Обираємо команду **Групповые операции** (можно натиснути кнопку **Групповые операции** на панелі інструментів для того, щоб у вікні запиту з'явилося поле **Групповая операция**).
5. У полі **Групповые операции** оберемо:
   - операцію **Группировка** для поля **Відділ**;
   - операцію **Count** для поля **Прізвище;**
   - операцію **Sum** для поля **Оклад**;
6. У полі **Сортировка** для поля **Відділ** обираємо **По возрастанию**.
7. Зберігаємо запит під ім'ям **Фонд**.
8. На базі створеного запиту розробляємо форму для відображення результатів

- перетягнемо всі поля в форму і розташуємо в потрібному нам вигляді.
- Натиснемо кнопку **Надпись** на **Панели элементов**, протягнемо покажчик по формі і введемо заголовок форми **ФОНД ЗАРАБІТНОЇ ПЛАТИ.**
- Зберігаємо форму під ім'ям **Фонд.**

Створення звіту

Для таблиці **Список** створемо звіт **Анкетні дані,** в якому наводяться анкетні дані співробітників, згруповані за відділами і відсортованні за алфавітом.

1. Вибрати вкладку **Создание** команду **Мастер отчетов.**
2. На екрані наводиться таблиця з переліком доступних полів. Вибрати поля, що входитимуть у звіт за допомогою кнопки **>>** або кнопки > і натиснути кнопку **Далее**.
3. Вибрати рівень групування за полем «**Відділ».**
4. Вибрати сортування записів по зростанню за полем **Відділ** і поле **Прізвище.**
5. Звіт відкрити в режимі **Конструктора.**
6. Розташуємо поля в області даних у необхідному нам порядку.
7. Для перейменування напису двічі клацнемо по ньому і введемо новий текст.
8. Виділимо поле зі списком, у контекстному меню оберемо команду **Преобразовать элемент в…,** а потім **Поле.**
9. У звіті перетягнемо поле і напис **Відділ** у розділ **Заголовок группы.**
10. Зберігаємо звіт під ім’ям **Анкетні дані.**

ПРАКТИЧНА РОБОТА № 6. Створення бази даних «Готель»

1. Створити базу даних, в якій передбачити збереження в кількох таблицях наступної інформації: код типу, тип номера, вартість проживання за добу, код номера, номер, код адміністратора, прізвище адміністратора, домашня адреса, зарплата адміністратора, код клієнта, прізвище клієнта, паспортні дані клієнта, домашня адреса клієнта, код поселення, дата поселення, тривалість проживання.

2. За допомогою конструктора таблиць створити структури таблиць. Передбачити поля, що дозволяють встановити зв'язки між таблицями.

3. Використовуючи схему даних зв'язати таблиці між собою з параметром підтримки цілісності даних, каскадним оновленням та каскадним вилученням зв'язаних полів.

4. Створити форми для введення даних. Ввести дані (4-5 записів для довідникових таблиць та 8-10 записів для облікової).

5. Створити запити, які дозволяють:

а) виводити на екран дані про номери вказаного типу;

б) виводити на екран дані про всіх клієнтів, які поселилися у готель сьогодні;

в) виводити на екран дані про обслуговування клієнтів адміністратором з кодом 3;

г) вивести на екран рахунок за проживання.

6. Створити звіт «Рахунок за проживання».

Структура БД

Таблиця **«Довідник типу номера»**

Ім'я поля	**Тип даних**	**Примітка**
Код типу	лічильник	ключове поле
Тип номера	текстовий	
Вартість проживання за добу	грошовий	

Таблиця **«Довідник номеру»**

Ім'я поля	**Тип даних**	**Примітка**
Код номера	лічильник	ключове поле
Номер	числовий	кількість знаків після коми - 0
Тип номеру	поле підстановки з таблиці «Довідник типу номера»	

Таблиця **«Довідник адміністраторів»**

Ім'я поля	**Тип даних**	**Примітка**
Код адміністратора	лічильник	ключове поле
Прізвище адміністратора	текстовий	
Домашня адреса	текстовий	
Зарплата адміністратора	грошовий	

Таблиця «**Довідник клієнтів**»

Ім'я поля	**Тип даних**	**Примітка**
Код клієнта	лічильник	ключове поле
Прізвище клієнта	текстовий	
Паспортні дані клієнта	текстовий	
Домашня адреса клієнта	текстовий	

Таблиця «**Облік проживання у готелі**»

Ім'я поля	**Тип даних**	**Примітка**
Код поселення	лічильник	ключове поле
Прізвище клієнта	поле підстановки з таблиці «Довідник клієнтів»	
Дата поселення	дата/час	
Тривалість проживання	числовий	
Номер	поле підстановки з таблиці «Довідник номеру»	
Адміністратор	поле підстановки з таблиці «Довідник адміністраторів»	

2. За допомогою конструктора таблиць створити структури таблиць:

а) завантажити *Access* — **Пуск/Программы/Microsoft Access/ Новая база данных** - ввести ім'я бази даних «ГОТЕЛЬ»;

б) вибрати вкладку **Создать** команда **Конструктор таблиц**

- ввести структуру таблиці «Довідник типу номера» - зберегти таблицю під ім'ям «Довідник типу номера» — закрити конструктор таблиць;

н) структури інших таблиць створити аналогічно.

3. Використовуючи схему даних зв'язати таблиці між собою з параметром підтримки цілісності даних, каскадним оновленням та каскадним вилученням зв'язаних полів.

4. Створити форми для введення даних. Ввести дані (4-5 записів для довідникових таблиць та 8-10 записів для облікової):

а) вибрати вкладку **Создание** команда **Мастер форм**;

б) у полі **«Таблнцы и запросы»** вибрати потрібну таблицю;

в) з вікна **«Доступные поля»** перемістити потрібні поля у вікно **«Выбранные поля»** за допомогою кнопок > та >> . Натиснути кнопку **«Далее»**;

г) вибрати зовнішній вигляд форми та інші потрібні параметри. Натиснути кнопку «**Готово**»;

д) ввести дані;

е) аналогічно створити інші форми та ввести дані.

Зверніть увагу! Дані вводяться спочатку у довідникові таблиці, а потім в облікову.

5. Створити запити, які дозволяють:

а) виводити на екран дані про номери вказаного типу.

Необхідно створити запит з параметром у зв'язку з тим, що не відомо який саме тип номерів потрібно вибрати з бази даних:

- перейти на вкладку **Создание** команда **Конструктор запросов;**
- у вікні **«Добавление таблицы»** виділити таблиці **«Довідник номеру»** та «**Довідник типу номера»** та натиснути кнопку **«Добавить».** Потім натиснути кнопку **«Закрыть»;**
- у вікні конструктора запитів вибрати потрібні поля та перетягти їх у бланк запиту;
- у стовпчику **«Тип номера»** визначити параметр запиту, тобто ввести в рядочку **«Условие отбора»** замість конкретного значення умову відбору в квадратних дужках. А саме: [Введіть тип]. Зберегти та закрити запит;
- для перегляду запиту завантажити його. У вікні введення параметру ввести потрібний тип номеру, наприклад, «двомісний». Натиснути кнопку **«ОК»;**

а) на екрані буде виведено всі двомісні номери;

б) виводити на екран дані про всіх клієнтів, які поселилися у готель сьогодні.

Для вибору всіх клієнтів, які поселилися у готель сьогодні необхідно у стовпчику «Дата поселення» в умові відбору ввести функцію вибору поточної дати — Date ()

в) виводити на екран дані про обслуговування клієнтів адміністратором з кодом 3.

Стовпчик «код адміністратора» не виводимо на друк, умова відбору - код номер 3.

г) вивести на екран рахунок за проживання.

Для виконання запиту вибираємо поля «Прізвище клієнта», «Дата поселення», «Тривалість проживання», «Вартість проживання за добу». Розрахункове поле «Сплачено» обчислюється за формулою:

Сплачено:[Тривалість проживання]*[Вартість проживання за добу]

Для обчислення рахунку клієнта вводимо параметри запиту в стовпчиках «Прізвище клієнта» та «Дата поселення»

6. Створити звіт «Рахунок за проживання».

а) вибрати вкладку **Создание** команда **Мастер отчетов;**

б) у полі «**Таблицы и запросы**» вибрати запит «Рахунок»;

в) з вікна «**Доступные поля»** перемістити всі поля у вікно **«Выбранные поля»** за допомогою кнопки **>>** Натиснути кнопку **«Далее»;**

г) при потребі вибрати рівні групування. Натиснути кнопку «**Далее**»;

д) при потребі вказати поля та порядок сортування по цим полям. Натиснути кнопку «**Далее**»;

є) вибрати вигляд макету. Натиснути кнопку «**Далее**»;

є) вибрати стиль звіту. Натиснути кнопку «**Готово**».

ПРАКТИЧНА РОБОТА № 7. Проектування бази даних «Інтернет-магазин»

Етап 1. Загальне проектування і мета створення бази даних.

База даних Інтернет-магазин може бути створена для Internet-магазину, заявки в якій приймаються безпосередньо через цю глобальну мережу. У даному прикладі ми будемо мати справу з сільськогосподарськими товарами, однак у цілому такий підхід можна застосовувати, розробляючи бази даних практично для будь-якого віртуального магазину. Деякі особливості, що стосуються характеристик тих або інших груп товарів, можуть бути показані в структурі таблиць як додаткові поля. В електронній комерції віртуальним є тільки прилавок (це Web-сторінка, що ви переглядаєте в браузері), а за ним, так само, як і в реальному житті, мають бути реальні співробітники, реальні склади, служба доставки і т.п.

Структура БД зумовлюється тими функціональними завданнями, які має виконувати додаток по обслуговуванню Internet-магазина.

Спочатку визначимо категорії користувачів, що працюють з віртуальним магазином:

- покупці (користувачі, що вже зареєструвалися в магазині);
- анонімні відвідувачі (користувачі, що проігнорували реєстрацію входячи до магазину);
- менеджери (користувачі, що мають більш широкі права доступу до даних).

Покупці можуть бути постійними (зареєстрованими) або разовими. Разовий покупець оформляє разове замовлення. Зареєструвавшись в системі, покупець стає постійним і може не тільки робити покупки в магазині, але і переглядати список своїх раніше оформлених замовлень і їхній стан. Роблячи покупки в магазині, і разові, і постійні покупці повідомляють деякий мінімум інформації про себе, необхідний менеджерам магазину, щоб забезпечити адресну доставку замовлення.

Анонімним відвідувачам дозволено переглядати товар, представлений у магазині.

Для менеджерів магазину має бути передбачена можливість переглядати списки замовлень і список зареєстрованих покупців, додавати товари в каталог, а також редагувати їх назви і ціни.

Наступна категорія об'єктів, що заслуговує уваги, — це ті форми (а в майбутньому — таблиці), в яких має фіксуватися вся інформація, "циркулююча" в рамках Internet-магазина. Для кожного типу інформації буде створена одна або кілька таблиць. Список полів, їхні назви і типи визначають на наступному етапі.

Як і в будь-якому віртуальному магазині, є каталог товарів, запропонованих для продажу. У каталозі повинні бути перераховані найменування товарів, їхні ціни, а також, у разі потреби, інша інформація. Як правило, у реальних електронних магазинах за допомогою каталогу товарів (одразу або за відповідним посиланням) можна ознайомитися з описом товарів і їхнім зображенням.

Усі покупки у віртуальному магазині оформляються за допомогою замовлень. Порядок оформлення замовлень для разових і постійних покупців буде різним, тому що зведення про зареєстрованих покупців уже зберігаються в системі, а всі параметри, необхідні для формування замовлення разового клієнта, повинні вводитися ним при відвідуванні магазину і виборі необхідних товарів. Хоча в реальному житті Internet-магазини пропонують одразу кілька способів оплати вартості товару (післяплата, банківський переказ і т.д.), у рамках даного прикладу обмежимося найпоширенішою формою оплати - готівкою.

Отже, про мету створення БД можна сказати наступне: основні функції бази даних полягатимуть у створенні, видаленні й відновленні записів; пошуку записів за ідентифікатором або умовою; підрахунку числа записів, що відповідають заданій умові, формуванні поточних звітів.

Етап 2. Визначення потрібних таблиць і полів бази даних.

У базі даних Інтернет-магазині можуть бути наступні таблиці: **Товари**, **Покупці** і **Замовлення**.

З огляду на той тип товарів, що пропонується в Internet-магазині, таблиця **Товари** може мати наступні поля, (див. таблицю).

Таблиця «**Товари**»

Ім'я поля	**Тип даних**	**Розмір поля**	**Опис**
КодТовару	**Счётчик**	**Длинное целое**	Ідентифікатор товару. Унікальне число, що автоматично присвоюється новому запису.
НазваТовару	**Текстовый**	60	Назва конкретного товару.
ОдиницяВиміру	**Текстовый**	30	Форма випуску товару.
Виробник	**Текстовый**	40	Ім’я виробника.
ЦінаТовару	**Денежный**		Ціна товару.
НаявністьТовару	**Логический**		Ознака наявності або відсутності товару на складі. Значення: **Да/Нет**. Якщо значення поля дорівнює **Нет**, це значить, що покупці не зможуть одержати доступ до даного товару. Він буде видний у каталозі, але покупець

			зрозуміє, що в даний момент цього товару просто немає на складі.
Актуальність	**Логический**		Службове поле. Ознака актуальності запису. Значення: **Да/Нет**. Якщо який-небудь товар потрібно вилучити з каталогу, то в цьому полі встановлюється ознака неактуальності запису (**Нет**). Тоді при відображенні каталогу такі записи вилучаються зі списку, і покупці не зможуть ні побачити його, ні замовити. Фізично запис з бази даних при цьому не віддаляється, що корисно, наприклад, якщо товар проданий, але в майбутньому ви хотіли б відновити його постачання.
ДатаРеєстрації	**Дата/время**		Службове поле. Дата створення/зміни запису, тобто додавання товару в каталог. Дозволяє сортувати записи в хронологічному порядку, наприклад, для пошуку нових надходжень.

У таблиці **Покупці** зберігаються дані про всіх зареєстрованих користувачів Internet-магазину.

Таблиця **«Покупці»**

Ім'я поля	**Тип даних**	**Розмір поля**	**Опис**
КодПокупця	**Счётчик**	**Длинное целое**	Ідентифікатор покупця. Унікальне число, що присвоюється автоматично новому запису.
РеєстраційнеІм’я	**Текстовый**	40	Ім'я користувача, під яким він зареєстрований у

			Internet-магазині.
Пароль	**Текстовый**	8	Пароль для входу в систему. Реєстраційне ім'я і пароль потрібні для авторизації відвідувача при роботі з Internet-магазином
Ім'яПокупця	**Текстовый**	50	П.І.Б. користувача або назва організації-замовника.
Адрес	**Текстовый**	60	Адреса покупця.
Телефон	**Текстовый**	12	Телофон покупця.
ЕлектроннийАдрес	**Текстовый**	30	Адреса електронної пошти.
Актуальність	**Логический**		Службове поле. Ознака актуальності запису. Значення: **Да/Нет**.
ДатаРеєстрації	**Дата/время**		Службове поле. Дата реєстрації при зміні даних в системі.

Поле **РегістраційнеІм'я** і **Пароль** необхідні для процедур реєстрації і доступу покупців до системи. У процесі авторизації покупець вводить свій ідентифікатор (**РегістраційнеІм'я**) і пароль. Якщо в таблиці Покупці присутній запис з такими параметрами, система розпізнає цього користувача, надає йому необхідні повноваження для роботи з інформацією, а у разі оформлення замовлення автоматично вносить необхідні дані про нього у форму замовлення.

Проаналізувавши ту інформацію, яка повинна зберігатися в таблиці **Замовлення**, можна зробити висновок, що вносити всі дані про замовлені товари в одну таблицю не раціонально. Дотримуючись правил нормалізації, можна розділити заголовки замовлень і списки замовлених товарів. Тому ми будемо використовувати дві таблиці, **Замовлення** і **ЗмістЗамовлень**, що дозволить скоротити об'єми бази даних і підвищити швидкість опрацювання інформації.

Таблиця **«ЗмістЗамовлень»**

Ім'я поля	**Тип даних**	**Розмір поля**	**Опис**
КодЗамовлення	**Числовой**	**Длинное целое**	Код замовлення. Ідентифікатор запису заголовка замовлення в таблиці Замовлення. Збігається зі значенням поля КодЗамовлення у таблиці Замовлення.
КодТовару	**Числовой**	**Длинное целое**	Код товару. Ідентифікатор запису позиції товару в таблиці Товари. Збігається зі значенням поля КодТовару в таблиці Товари.
ЦінаТовару	**Денежный**		Поле містить вартість одиниці даного товару в момент оформлення замовлення.
Кількість	**Числовой**	**Целое**	Кількість одиниць товару.
Актуальність	**Логический**		Службове поле. Ознака актуальності запису. Значення: **Да/Нет.**
ДатаРеєстрації	**Дата/время**		Службове поле. Дата реєстрації або зміни запису.

Таблиця «**Замовлення»**

Ім'я поля	**Тип даних**	**Розмір поля**	**Опис**
КодЗамовлення	**Счётчик**	**Длинное целое**	Ідентифікатор замовлення. Унікальний номер замовлення.
ДатаЗамовлення	**Дата/время**		Дата розміщення замовлення.
КодПокупця	**Числовой**	**Длинное целое**	Ідентифікатор запису зареєстрованого покупця в таблиці Покупці. Збігається зі значенням поля КодПокупця таблиці

			Покупці.
Ім'яОдержувача	**Текстовый**	50	Контактна особа, з якою має зв'язатися менеджер виконуючи замовлення.
АдресаОдержувача	**Текстовый**	60	Адреса одержувача (може не збігатися з адресою покупця).
ТелефонОдержувача	**Текстовый**	12	Телефон одержувача.
ЕлАдресаОдержувача	**Текстовый**	30	Адреса електронної пошти одержувача.
СтанЗамовлення	**Числовой**	Байт	Код поточного стану замовлення: 0 -створюється, 3-відправлений менеджерові, 5-обробляється, 7- виконаний.
Актуальність	**Логический**		Службове поле. Ознака актуальності запису. Значення: **Да/Нет**.
ДатаРеєстрації	**Дата/время**		Службове поле. Дата створення/зміни запису

Етап 3. Визначення полів з унікальними значеннями в кожному записі.

Як первинні ключі використовуються наступні поля:

- для таблиці **Товари** — поле **КодТовару**;
- для таблиці **Покупці** — поле **КодПокупця**;
- для таблиці **Замовлення** — поле **КодЗамовлення**;
- для таблиці **ЗмістЗамовлень** — комбінація полів **КодЗамовлень** і **КодТовару**

Етап 4. Визначення зв'язків між таблицями.

Зв'язок між таблицями **Замовлення** і **ЗмістЗамовлення** за допомогою поля **КодЗамовлення**, а поле **КодТовару** зв'язує записи таблиці **ЗмістЗамовлення** з таблицею **Товари**. Таблиці **Замовлення** і **Покупці** пов'язані між собою полем **КодПокупця**.

Етап 5. Удосконалення структури БД.

На цьому етапі за розробленими раніше схемами створюються таблиці з необхідними типами полів, вибираються первинні ключі і формуються зв'язки між таблицями. Коли структурна основа бази даних

готова, варто уважно проаналізувати її, щоб ще до заповнення таблиць даними усунути можливі недоробки.

Етап 6. Введення даних і створення інших об'єктів БД.

Після створення й успішного тестування структури таблиць можна заповнювати їх реальними даними. На цьому етапі розробити і створити також усі інші необхідні об'єкти бази даних — запити, форми, звіти.

Рекомендована література

1. Гайджарги В. І., Ізварін І. В. Бази даних в інформаційних системах. Київ : Університет "Україна", 2018. 418 с.
2. Дубинин М. Ю., Костикова А. А. Введение в геоинформационные системы : онлайн курс. URL: http://gis-lab.info/docs/giscourse/.
3. Інформаційні технології : навчальний посібник / Ю. В. Волосюк та ін. Миколаїв : МНАУ, 2017. 200 с.
4. Кейк Д., Лященко А. А., Путренко В. В. Системи керування базами геоданих для інфраструктури просторових даних : навчальний посібник. Київ : Планета-прінт, 2017. 456 с.
5. Митропольський О. Ю., Іванік О. М. Основи морської геології. Київ : ВПЦ “Київський університет, 2017. 219 с.
6. Публічна карта GISFile. URL: http://gisfile.com/publicmap.htm.
7. Публічна кадастрова карта України / Державна служба України з питань геодезії, картографії та кадастру. URL: http://map.land.gov.ua/kadastrova-karta.

ЗМІСТ

ПЕРЕДМОВА 3

ТЕОРЕТИЧНІ ОСНОВИ СИСТЕМИ УПРАВЛІННЯ БАЗАМИ ДАНИХ (СУБД) MS ACCESS.....4

ПРАКТИЧНА РОБОТА № 1. СТВОРЕННЯ БАЗИ ДАНИХ 14

ПРАКТИЧНА РОБОТА № 2. РОБОТА З РЕЛЯЦІЙНОЮ БАЗОЮ ДАНИХ MS ACCESS 23

ПРАКТИЧНА РОБОТА № 3. ОСНОВНІ ПРИНЦИПИ СТВОРЕННЯ БАЗИ ДАНИХ У MS ACCESS.....32

ПРАКТИЧНА РОБОТА № 4. СТВОРЕННЯ БАЗИ ДАНИХ «ЗЕМЛЯ» 40

ПРАКТИЧНА РОБОТА № 5. СТВОРЕННЯ БАЗИ ДАНИХ «СПИСОК СПІВРОБІТНИКІВ».....47

ПРАКТИЧНА РОБОТА № 6. СТВОРЕННЯ БАЗИ ДАНИХ «ГОТЕЛЬ» 53

ПРАКТИЧНА РОБОТА № 7. ПРОЕКТУВАННЯ БАЗИ ДАНИХ «ІНТЕРНЕТ-МАГАЗИН» 57

РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....63

Навчально-методичне видання

ГЕОІНФОРМАЦІЙНІ СИСТЕМИ І БАЗИ ДАНИХ

Методичні рекомендації

Укладач: **Борян** Людмила Олександрівна

Формат 60х84 1/16. Ум. друк. арк. 4,1
Тираж 10 прим. Зам. №___

Надруковано у видавничому відділі
Миколаївського національного аграрного університету
54029, м. Миколаїв, вул. Георгія Гонгадзе, 9

Свідоцтво суб'єкта видавничої справи ДК № 4490 від 20.02.2013 р.