

Міністерство освіти і науки України
Національний університет водного господарства та
природокористування

Кафедра комп'ютерних наук та прикладної математики

04-01-49

МЕТОДИЧНІ ВКАЗІВКИ

до виконання курсової роботи з навчальної дисципліни
«Бази даних»

для здобувачів вищої освіти першого (бакалаврського) рівня
за освітньо-професійною програмою «Комп'ютерні науки»
спеціальності 122 «Комп'ютерні науки»
денної, заочної та дистанційної форм навчання

Рекомендовано науково-
методичною радою
з якості ННІ АКOT

Протокол № 7
від 27.03.2020 р.

Рівне – 2020

Методичні вказівки до виконання курсової роботи з навчальної дисципліни «Бази даних» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» денної, заочної та дистанційної форм навчання [Електронне видання] / Бачишина Л. Д., Харів Н. О. – Рівне : НУВГП, 2020. – 24 с.

Укладачі:

Бачишина Л. Д., к.е.н, доцент кафедри комп'ютерних наук та прикладної математики;

Харів Н. О., старший викладач кафедри комп'ютерних наук та прикладної математики.

Відповідальний за випуск:

Мартинюк П. М., д.т.н., професор, завідувач кафедри комп'ютерних наук та прикладної математики.

Керівник групи забезпечення спеціальності 122 «Комп'ютерні науки»:

Мартинюк П. М., д.т.н., професор, завідувач кафедри комп'ютерних наук та прикладної математики

© Бачишина Л. Д.,
Харів Н. О., 2020
© НУВГП, 2020

ЗМІСТ

1. Мета і завдання курсової роботи.....	4
2. Тематика курсових робіт.....	5
2.1. Вибір теми курсової роботи.....	5
2.2. Приклади тем курсових робіт.....	5
3. Вимоги до курсової роботи.....	6
3.1. Оформлення курсової роботи.....	6
3.2. Вимоги до розробки бази даних.....	7
4. Захист курсової роботи.....	7
4.1. Шкала оцінювання курсової роботи.....	7
5. Приклад типової курсової роботи.....	8
5.1. Основні етапи виконання курсової роботи.....	8
5.2. Приклад курсової роботи.....	9

1. Мета і завдання курсової роботи

Згідно з навчальним планом для здобувачів вищої освіти першого (бакалаврського) рівня за спеціальністю 122 «Комп'ютерні науки» передбачено виконання курсової роботи яка поглиблює, закріплює та систематизує навички самостійного, творчого підходу до розв'язування поставленої задачі та сприяє удосконаленню вмінь, набутих під час практичних занять.

Виконання курсової роботи є завершальним етапом вивчення дисципліни «Бази даних» і має своєю *головною метою*:

1. Засвоєння методики проектування інформаційних систем;
2. Закріплення основних прийомів розробки баз даних;
3. Закріплення вивчених та освоєння нових можливостей візуально-подійного підходу в програмуванні баз даних;
4. Освоєння методики створення прикладних комплексних додатків та самостійне вивчення спеціальних алгоритмів, що стосуються окремої теми.

Кожна курсова робота є проблемно-орієнтованою, а тому містить певну наукову новизну. Виконання роботи сприяє залученню студентів до сучасних проблем програмування і є підґрунтям для майбутніх дипломних робіт.

Курсова робота *повинна містити*

- постановку задачі;
- опис предметної області;
- нормалізацію бази даних;
- схему зв'язків;
- інфологічну модель;
- реалізацію з використанням візуально-подійного або об'єктно-орієнтованого підходів, методів розробки Web-додатків;
- тестовий приклад;
- аналіз отриманих результатів;
- висновки.

Під час виконання курсової роботи студент повинен узгоджувати з керівником роботи *всі відхилення* від поставленого завдання, може консультуватися по незрозумілих та важких питаннях на консультаціях і заняттях. Виконання роботи розбивається на етапи. При цьому своєчасність і якість виконання студентом кожного етапу фіксується в журналі викладача, а надалі враховується при захисті курсової роботи та її оцінюванні.

2. Тематика курсових робіт

2.1. Вибір теми курсової роботи

Тема кожної курсової роботи визначається керівником роботи з урахуванням побажань студента, його інтересів та теми майбутньої дипломної роботи.

Тематика курсових робіт розробляється і затверджується на кафедрі комп'ютерних наук та прикладної математики. Всі теми відповідають проблемно-орієнтованим задачам, а тому вимагають від студента творчого підходу до їх розв'язання.

2.2. Приклади тем курсових робіт

1. Облік дорожньо-транспортних пригод.
2. Облік судових справ.
3. Картотека фото-центра.
4. Аналіз роботи фірми (прибутковість, збитковість)
5. Журнал вчителя.
6. Облік студентів вузу.
7. Облік стаціонарних хворих в лікарні.
8. Облік товарів на складі.
9. Нарахування пенсії.
10. Шкідливі викиди в атмосферу.
11. Облік кадрів на підприємстві.
12. Облік ліків в аптеці.
13. Записна книга сімейного лікаря.
14. Діагностичний центр.
15. Облік робіт ЖЕКу.
16. Облік кредитів в банку.
17. Перепис населення.
18. Облік плати за телефон.
19. Облік підписки на пресу.
20. Розрахунок виконання навантаження викладачами.
21. Демографічний стан.
22. Рейтингова таблиця спортивних команд.
23. Інформаційно-довідкова система „Автовокзал”.
24. Інформаційно-довідкова система „Конкурс Міс України”.
25. Приватизація земельних ділянок.
26. Пісенний фестиваль (рейтингова таблиця).
27. Прокат відеокасет.
28. Художній салон.
29. Продаж комп'ютерів.

30. Дизайн офісів.
31. Юридична контора.
32. Туристичне бюро.
33. Страхова компанія.
34. Облік відвідувачів в бібліотеці.
35. Чемпіонат з футболу.
36. Облік відвідувачів нотаріальної контори.
37. Облік актів громадського стану.

3. Вимоги до курсової роботи

3.1. Оформлення курсової роботи

Курсова робота повинна містити наступне:

1. Титульний лист (див. приклад курсової роботи).
2. Зміст (з вказанням сторінок).
3. Вступ (короткий виклад важливості розв'язуваного класу задач та доцільність її реалізації на ПК, опис предметної області, обґрунтування вибору СКБД).
4. Змістовна постановка задачі (аналіз поставленої задачі, наведення загального вигляду вхідних та вихідних даних, форм, запитів, звітів, опис інтерфейсу).
5. Нормалізація бази даних.
6. Схема зв'язків і опис умов цілісності.
7. Інфологічна модель.
8. Зображення та описання всіх таблиць, форм, запитів, звітів, які є в базі даних.
9. Тестовий приклад (з наведеною повною вхідною інформацією, яка дозволяє перевірити всі режими роботи програми, усіма вихідними формами, з аналізом отриманих результатів).
10. Інструкція користувачу, в якій міститься пояснення щодо правил користування програмою.
11. Висновки.
12. Список використаної літератури.

Курсова робота оформляється на листах формату А4, що зверху нумеруються і зшиваються з лівої сторони. Ліве поле повинне бути від 25 до 30 мм, праве – не менше 10 мм; верхнє – 20 мм; нижнє – 20 мм.

Текст повинен відповідати змісту. Всі пункти змісту викладаються чітко і коротко.

3.2. Вимоги до розробки бази даних

СУБД для створення бази даних вибирається за бажанням студента. Це може бути *Oracle, SQL Server, Delphi, InterBase, MySQL, Firebird, PHPMyAdmin* тощо.

При написанні програми потрібно дотримуватися наступних вимог:

1. Початкові дані зберігаються у таблицях (не менше 3-х таблиць, не менше 15 записів в базових таблицях).
2. Форми повинні забезпечувати перегляд, доповнення та коригування даних.
3. Запити повинні виводити дані, які задовольняють певним умовам.
4. Звіти виводять інформацію з таблиць або запитів та підсумкові дані.
5. Довідкова система містить інформацію по користуванню базою даних.
6. База даних повинна містити доступний для користування інтерфейс.
7. На носії інформації обов'язково повинен знаходитися файл бази даних та текст звіту.
8. Тестування бази даних провести на прикладі, що дозволяє перевірити усі режими роботи.
9. Реалізувати використання відображень (VIEW), тригерів (TRIGGER) та збережених процедур (STORED-PROCEDURE).

4. Захист курсової роботи

Оформлена курсова робота подається студентом разом з програмним додатком до захисту. Захист роботи проходить у формі співбесіди. Під час захисту потрібно коротко розповісти про зміст задачі, основні етапи проектування бази даних, її реалізацію. На поставлені запитання про виконання роботи студент повинен дати вичерпні відповіді. Мета опитування полягає у встановленні глибини засвоєння студентом відповідного матеріалу з курсової роботи, ступеня самостійності її виконання. При оцінюванні роботи враховується якість її виконання і оформлення, своєчасність виконання етапів роботи, результати співбесіди.

4.1. Шкала оцінювання курсової роботи

Програмна реалізація	Оформлення звіту	Захист
60	20	20

5. Приклад типової курсової роботи

5.1. Основні етапи виконання курсової роботи

Виконання курсової роботи складається з наступних етапів:

- опис предметної області за обраною темою курсової роботи;
- розробка моделі бази даних на задану тему (нормалізація бази даних);
- розробка схеми зав'язків між таблицями (EER-diagram);
- описати умови збереження цілісності бази даних;
- розробити інфологічну модель;
- розробити форми для введення, виведення та коригування даних;
- розробити систему запитів до бази даних;
- розробити форми звітів;
- розробити технічну документацію (у вигляді довідки для користувача).

Реляційна модель бази даних є основою більшості сучасних СКБД, тому пропонується в курсовій роботі розробити реляційну модель, використовуючи один з наведених нижче підходів:

а) підхід запропонований Е. Ф. Коддом полягає у використанні принципу нормалізації на основі нормальних форм.

Нормалізація, це процес перетворення відношення, що має деякі недоліки, у відношення, що їх не має.

Іноді для комп'ютерної реалізації недоцільно зводити базу даних до нормальних форм вище 3-ї, тому в розглянутому прикладі виконано нормалізацію, яку автори вважають найбільш зручною для роботи.

Перша нормальна форма

1. Таблиця не повинна містити записів, що повторюються.
2. Таблиця не повинна містити групи полів, що повторюються.
3. Рядки та стовпці не впорядковані.
4. Таблиця містить логічно неподільні значення.

Друга нормальна форма

1. Відповідає всім умови 1-ї нормальної форми.
2. Будь-яке неключове поле залежить повністю від ключового поля.

Третя нормальна форма

1. Відповідає всім умови 2-ї нормальної форми.
 2. Неключові поля взаємно незалежні.
- б) Доменно-ключову нормальну форму, запропоновану Фагіном.

Визначення: відношення знаходиться в доменно-ключовій

нормальній формі(ДКНФ), якщо кожне обмеження, що накладається на відношення, є логічним наслідком визначення доменів та ключів.

За визначенням Фагіна, обмеження – будь-яке правило, що регулює можливі статичні значення атрибутів. Правила редагування, обмеження взаємовідношень, структури відношень, функціональні та багатозначні залежності є прикладами таких обмежень, за виключенням обмежень, що відносяться до зміни значень атрибутів та залежать від часу. Наприклад обмеження типу ”Зарплата працівника за поточний період не повинна перевищувати зарплату за попередній період” не підпадає під визначення обмежень ДКНФ.

5.2. Приклад курсової роботи

Потрібно зауважити, що в даному прикладі розглянуті тільки деякі аспекти побудови бази даних.

Приклад оформлення титульного листа

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО
ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ**

Кафедра комп'ютерних наук та прикладної математики

КУРСОВА РОБОТА
з дисципліни «Бази даних»
на тему

(назва теми роботи)

Виконав студент ННІ
АКОТ: група

(прізвище, ім'я, по батькові)

Керівник:

(науковий ступінь, вчене
звання, прізвище, ініціали)

Рівне - 20__

Зміст

Вступ.....	
1. Опис предметної області.....	
2. Нормалізація бази даних.....	
3. Схема зв'язків між таблицями.....	
4. Умови забезпечення цілісності даних.....	
5. Інфологічна модель бази даних.....	
6. Форми для введення та коригування даних.....	
7. Запити до інформаційно-довідкової системи.....	
8. Звіти.....	
9. Меню.....	
10. Довідка.....	
Висновки.....	
Література.....	

Вступ

Бази даних, поза сумнівами, займають провідне положення в області інформаційних технологій. Вони стають невід'ємною частиною життя сучасної людини. Але саме в останні роки, завдяки розвитку Internet, знання технологій баз даних стало одним з найбільш популярних шляхів в кар'єрі. В наш час життя людини настільки насичено інформацією, що для її збереження і обробки створено величезну кількість баз та банків даних різноманітного призначення.

Поняття *база даних* (БД) *та система керування базами даних* (СКБД) найчастіше використовується в аспекті пов'язаному з комп'ютерної обробкою даних.

Проектування баз даних вимагає одночасно і мистецтво і інженерних навиків. Розуміння вимог користувача та втілення їх в ефективній логічній структурі є мистецтвом. Перетворення логічної структури в фізичну базу даних з функціонально завершеними високопродуктивними додатками є інженерною задачею.

База даних – це електронна картотека, електронне сховище даних. При роботі з нею виникає необхідність виконання ряду операцій, в першу чергу це:

- додавання нової інформації до файлів бази даних;
- додавання нових пустих файлів до бази даних;
- зміна інформації в існуючих файлах бази даних;
- пошук інформації в базі даних;
- знищення інформації в базі даних;
- знищення файлів з бази даних.

Мета даної курсової роботи: отримати навички проектування логічної структури та створення фізичної бази даних.

Опис предметної області

Нехай потрібно розробити модель інформаційної системи для автоматизації обліку замовлень покупців.

Система повинна передбачати ведення каталогу товарів, містити дані про замовника, здійснювати облік замовлень.

Всю інформацію з бази даних можна помістити в одну таблицю. Зразу ж відзначимо, що в процесі проектування поля можуть змінюватись.

Назва поля	Тип поля	Розмірність
Код замовника	Числовий(Ціле)	4
Прізвище	Текстовий	15
Ім'я	Текстовий	20
По-батькові	Текстовий	25
Поштовий індекс	Текстовий	5
Країна	Текстовий	20
Область	Текстовий	20
Місто	Текстовий	15
Адреса	Текстовий	40
Телефон	Текстовий	8
Примітки	Мето	
Код замовлення	Числовий (Ціле)	4
Код товару	Числовий (Ціле)	4
Назва товару	Текстовий	20
Марка товару	Текстовий	20
Дата замовлення	Дата	
Кількість замовленого	Числовий	8
Дата продажу	Дата	
Код продавця	Числовий(Ціле)	5
Прізвище продавця	Текстовий	15
Ціна товару	Грошовий	8.2

При проектуванні бази даних потрібно вирішити питання про найбільш ефективну структуру даних.

Основні цілі, які при цьому переслідуються:

- забезпечити швидкий доступ до даних в таблицях;
- виключити непотрібне повторення даних;
- забезпечити цілісність даних таким чином, щоб при видаленні чи зміні одних об'єктів автоматично відбувались зміни в пов'язаних з ними об'єктах.

Замовник повинен мати можливість переглянути прайс-лист та вибрати необхідне для замовлення.

При надходженні нового товару база даних повинна поповнюватись відповідними даними. За це відповідає менеджер або адміністратор.

Продавець-менеджер повинен володіти повною інформацією про замовлення, покупців, товари; слідкувати за веденням каталогу товарів, робити щоденний звіт про реалізацію та надходження товарів.

Замовлення товару можливе при реєстрації покупця.

Адміністрація повинна слідкувати за поповненням кредитів замовниками та мати можливість формування щомісячних та кварталних звітів про реалізацію та надходження товарів, підрахунок прибутків від реалізації.

В кінці кожного дня підраховується залишок товарів на складі.

Нормалізація бази даних

Вихідна таблиця не відповідає всім умовам 1-ї нормальної форми, тому що в ній є дані, які повторюються.

Нормалізуємо нашу базу у відповідності з теорією нормалізації Кодда.

Зведемо її до 1-ї нормальної форми, для цього розіб'ємо таблицю на кілька. Відразу прослідкуємо, щоб кожне поле вмішувало логічну неподільну одиницю. Поле **Адреса** розіб'ємо на поля **Вулиця**, **Будинок**, **Квартира**.

Таблиця 1

Замовник

Поле **Код замовника** – ключове, оскільки дане поле однозначно визначає кожен запис таблиці.

Назва поля	Тип поля	Розмірність
Код замовника	int	4
Прізвище	varchar (45)	15
Ім'я	varchar (45)	20
По-батькові	varchar (45)	25
Поштовий індекс	varchar (8)	5
Країна	varchar (15)	20
Область	varchar (15)	20
Місто	varchar (15)	15
Вулиця	varchar (25)	20
Будинок	varchar (3)	5
Квартира	varchar (3)	5
Телефон	varchar (10)	8
Примітки	Мемо	

Таблиця 2

Замовлення

Поле **Код замовлення** – ключове.

Назва поля	Тип поля	Розмірність
Код замовлення	int	4
Код товару	int	4
Назва товару	varchar (45)	20
Марка товару	varchar (25)	20
Ціна товару	decimal(10,2)	8.2
Дата замовлення	date	
Код замовника	int	4
Кількість замовленого	int	8
Дата продажу	date	
Відмітка про виконання	blob	
Код продавця	int	5
Прізвище продавця	varchar (45)	15

Зведення до другої нормальної форми передбачає розділення таблиці з ціллю розташування в окрему таблицю (чи кілька таблиць) стовпців, які не залежать від первинного ключа. В результаті такого розбиття кожне з неключових полів повинно виявитись залежним від первинного ключа. В нашому прикладі це таблиця **Замовлення**. Дана таблиця не є таблицею в 2-ї нормальній формі, тому що поля **Назва товару**, **Марка товару** та **Ціна товару** однозначно ідентифікуються лише полем **Код товару**. Виділимо дані про товари в окрему таблицю і назовемо її **Товари**.

Таблиця 3

Товари

Поля **Код товару** – ключ

Назва поля	Тип поля	Розмірність
Код товару	int	4
Назва товару	varchar (45)	20
Марка товару	varchar (45)	20
Ціна товару	decimal (10,2)	8.2

Звернемось до таблиці **Замовлення**. Поле **Прізвище продавця** цієї таблиці містить дані, які однозначно визначаються полем **Код продавця** і не залежить від поля **Код замовлення**. Оскільки одне неключове поле залежить від іншого неключовим полем, то ця таблиця

не знаходиться в 3-й нормальній формі. Створимо ще одну таблицю **Службовці** і доповнимо її даними про продавців.

Таблиця 4

Службовці

Поле **Таб_номер** – ключове

Назва поля	Тип поля	Розмірність
Таб_номер	int	5
Прізвище	varchar (45)	15
Ім'я	varchar (45)	20
По-батькові	varchar (45)	25
Посада	varchar (45)	15
Дата прийому	date	
Оклад	decimal (10,2)	8.2
Поштовий індекс	varchar (6)	5
Країна	varchar (15)	20
Область	varchar (15)	20
Місто	varchar (10)	15
Вулиця	varchar (15)	20
Будинок	varchar (3)	5
Квартира	varchar (3)	5
Телефон	varchar (10)	8
Примітки	Memo	

В результаті наведених перетворень таблиця **Замовлення** буде мати вигляд:

Таблиця 5

Замовлення

Поле **Код замовлення** – ключове.

Назва поля	Тип поля	Розмірність
Код замовлення	int	4
Дата замовлення	date	
Код замовника	int	4
Кількість замовленого	int	8
Дата продажу	date	
Код продавця	int	5

Отже, база даних буде включати таблиці 1, 3, 4, 5.

Нормалізуємо нашу базу у відповідності з теорією нормалізації Фагіна (доменно-ключова нормальна форма).

Ми маємо початкове відношення **Замовлення** (Код замовника, Прізвище, Ім'я, По-батькові, Поштовий індекс, Країна, Область, Місто, Адреса, Телефон, Примітки, Код замовлення, Код товару, Назва товару, Марка товару, Дата замовлення, Кількість замовленого, Дата продажу, Код продавця, Прізвище продавця, Ціна товару)

Ключ: Код замовника, Код замовлення, Код товару, Код продавця.

Обмеження:

Код замовника --> Прізвище| Ім'я| По-батькові

Код замовника --> Поштовий індекс| Країна| Область| Місто| Адреса| Телефон

Код товару --> Назва товару| Марка товару| Ціна товару

Код замовлення--> Дата замовлення| Кількість замовленого| Дата продажу

Код продавця--> Прізвище продавця

Згадаємо: суть нормалізації полягає в тому , що кожне відношення повинно мати одну тему. В нашому вихідному відношенні присутні 4-ри теми: **Замовник**, **Товар**, **Замовлення**, **Продавець**.

Виходячи з усього описаного вище будемо мати наступну модель бази даних:

Замовник (Код замовника, Прізвище, Ім'я, По-батькові, Поштовий індекс, Країна, Область, Місто, Адреса, Телефон, Примітки)

Ключ: Код замовника

Замовлення (Код замовлення, Код товару, Дата замовлення, Кількість замовленого, Дата продажу, Код продавця)

Ключ: Код замовлення

Товар (Код товару, Назва товару, Марка товару, Ціна товару)

Ключ: Код товару

Продавець (Код продавця, Прізвище продавця,)

Ключ: Код продавця

В результаті застосування різних підходів до нормалізації ми отримали однакову модель бази даних.

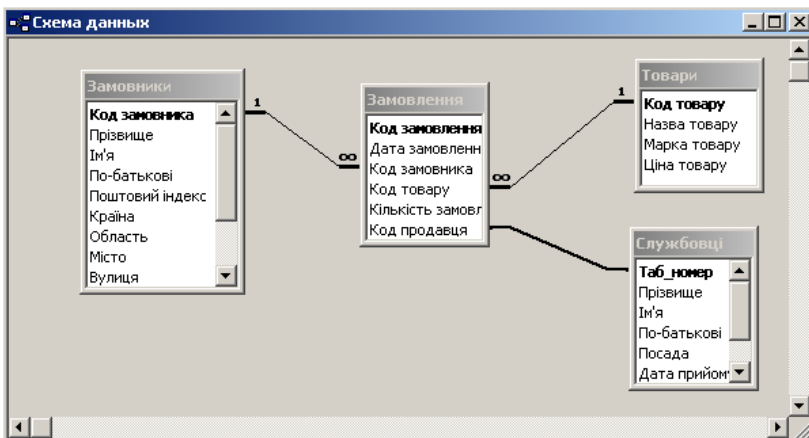


Рис.1. Схема зв'язків між таблицями

Умови збереження цілісності даних

Опишемо умови збереження цілісності даних:

В таблиці **Замовники** зберігаються дані про замовників. Знищення даних з цієї таблиці можливе тільки при попередньому знищенні даних з таблиці **Замовлення**. (Задано умови цілісності даних, каскадне знищення зв'язаних записів не встановлено). Зміна з певних причин коду замовника в таблиці **Замовники** автоматично відображається в таблиці **Замовлення** (Задано умови цілісності даних, встановлено каскадне оновлення зв'язаних полів). Внесення даних про нового замовника у таблицю **Замовлення** не можливе без попереднього внесення даних у таблицю **Замовники**.

Зв'язок між таблицями **Товари** і **Замовлення** встановлено по полю **Код товару**. Задано умови цілісності даних.

Ключове поле **Таб_номер** таблиці **Службовці** зв'язується із відповідним зовнішнім ключем **Код продавця** таблиці **Замовлення**. Цілісність даних не встановлена.

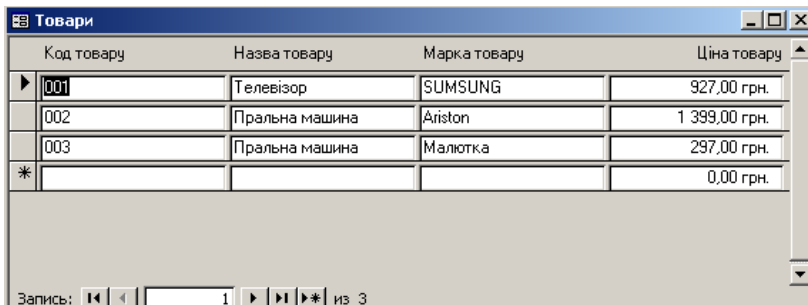
Інфологічна модель даних

(Необхідно розмістити інфологічну модель даних)

Форми для введення та коригування даних

Розробимо форми для введення даних в таблиці. Ці форми будуть використовуватись при перегляді, доповненні та коригуванні бази даних. Їх призначення полегшити роботу кінцевого користувача.

Форма **Товари** (Рис.2) дозволяє переглянути перелік товарів і ввести дані про новий товар.



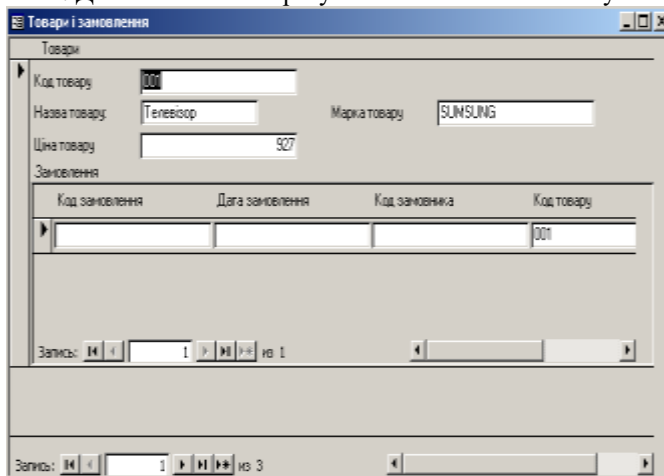
The screenshot shows a window titled "Товари" with a table containing the following data:

Код товару	Назва товару	Марка товару	Ціна товару
001	Телевізор	SUMSUNG	927,00 грн.
002	Пральна машина	Ariston	1 399,00 грн.
003	Пральна машина	Малютка	297,00 грн.
*			0,00 грн.

Below the table, there is a record navigation bar: "Запись: 1 из 3".

Рис. 2. Форма для перегляду та введення даних в базу таблицю.

Форма **Товари і замовлення** (Рис 3.) складається з двох форм: основної **Товари**, яка дозволяє вибрати товар із списку товарів, і підпорядкованої **Замовлення**, яка дозволяє зробити на вибраний товар замовлення. Дозволяє вивести результати виконання запиту.



The screenshot shows a window titled "Товари і замовлення" with the following fields and table:

Товари

Код товару: 001
Назва товару: Телевізор
Марка товару: SAMSUNG
Ціна товару: 927

Замовлення

Код замовлення	Дата замовлення	Код замовника	Код товару
			001

At the bottom, there is a record navigation bar: "Запись: 1 из 3".

Рис 3. Виведення результатів запиту

Форма **Замовники** (Рис 4.) дозволяє ввести дані про замовника:

Рис.4. Форма виведення даних з таблиць

Запити до інформаційно-довідкової системи:

1. Перший запит дозволяє отримати інформацію про товар по назві товару:

```
SELECT Товари.Код товару, Товари.Назва товару,
Товари.Марка товару, Товари.Ціна товару
FROM Товари
WHERE Товари.Назва товару=n_tov
```

2. Другий запит дозволяє отримати інформацію про прізвище клієнта, назву, марку і ціну замовленого товару:

```
SELECT Замовлення.Код замовлення, Замовлення.Дата замовлення,
Замовники.Прізвище, Замовники.Ім'я, Товари.Назва товару,
Товари.Марка товару, Товари.Ціна товару FROM Замовники INNER
JOIN (Товари INNER JOIN Замовлення ON Товари.Код товару =
Замовлення.Код товару) ON Замовники.Код замовника =
Замовлення.Код замовника;
```

3. Третій запит дозволяє підрахувати вартість замовленого товару:

```
SELECT Замовлення.Код замовлення, Замовлення.Код товару,
Замовлення.Кількість замовленого, Товари.Ціна товару, Ціна
```

товару*Кількість замовленого AS До оплати FROM Товари INNER JOIN Замовлення ON Товари.Код товару = Замовлення.Код товару;

4. Четвертий запит дозволяє які замовлення виконані протягом дня.

SELECT Замовлення.Код замовлення, Замовлення.Дата замовлення, Замовлення.Код замовника, Замовлення.Код товару FROM Замовлення WHERE Замовлення.Дата замовлення=Date().

Звіти

1.Звіт за день. Даний звіт підраховує загальну кількість продукції, замовлену за день.

Замовлення					
<i>Дата замовлен</i>	<i>Код замовлен</i>	<i>Код замовник</i>	<i>Код товару</i>	<i>Кількість</i>	<i>Код продавц</i>
24.04.2005					
	111	123	003	2	12
	112	547	003	2	13
	222	123	003	1	12
	234	548	001	1	13
	333	546	002	1	23
	334	548	002	2	12
	335	123	001	2	12
<i>Итого для 'Дата замовлення' = 24.04.2005 (7 записів)</i>					
ИТОГО					11

2. Звіт **Продано**. Дозволяє вивести інформацію про товари, замовлені за день, суму замовлення кожного товару і загальну суму за день

Замовлено				
Код товару	Код замовлення	Кількість замовленого	Ціна товару до оплати	
001				
	234	1	927,00 грн	927
	335	2	927,00 грн	1854
Итого для 'Код товару' = 001 (2 записів)				
Sum				2781
002				
	334	2	1 399,00 грн	2798
	333	1	1 399,00 грн	1399
Итого для 'Код товару' = 002 (2 записів)				
Sum				4197
003				
	112	2	297,00 грн	594
	111	2	297,00 грн	594
	222	1	297,00 грн	297
Итого для 'Код товару' = 003 (3 записів)				
Sum				1485
Итого				8463

Меню

Меню користувача повинно приховувати всі особливості роботи в середовищі СКБД, надавати кінцевому користувачу зручний інтерфейс та можливості по веденню та коригуванню бази даних:

- доповнення таблиць новими даними;
- знищення даних з бази;
- формування умов запитів ;
- формування звітів.

Потрібно передбачити наявність довідкової інформації по роботі з інформаційною системою.

Висновки

Комп'ютерні інформаційні системи представляють собою програмний комплекс, до задач якого входить підтримка надійного збереження, виконання перетворення інформації та необхідних обчислень, надання користувачу зручного та зрозумілого інтерфейсу.

При виконанні даної курсової роботи ми вивчили:

- теорію проектування баз даних;
- різні моделі даних;
- теорію нормалізації Кодда та Фагіна;
- структуровану мову запитів SQL.

Навчилися:

- робити аналіз предметної області;
- проектувати логічну структуру бази даних;
- створювати фізичну базу даних використовуючи

сучасні СКБД.

Набуті нами навички можуть бути використані (при необхідності) в майбутньому в різних сферах діяльності, забезпечать ефективне збереження і обробку бізнесової та особистої інформації.



Від авторів :

Авторські права на Вступ та Висновки захищені, незаконне розповсюдження та копіювання забороняється. Пропонується скористатись власними творчими здібностями і створити особистий шедевр

Методичне забезпечення

1. Інформаційні ресурси у цифровому репозиторії. URL: <http://ep3.nuwm.edu.ua/view/types/methods/> .
2. Харів Н. О. Бази даних та інформаційні системи : навч. посіб. Рівне : НУВГП, 2018. 127 с. URL : <http://ep3.nuwm.edu.ua/9129/> .
3. Бачишина Л. Д., Харів Н. О. Методичні вказівки та завдання до виконання лабораторних робіт з дисципліни «Бази даних» для студентів освітнього рівня бакалавр спеціальностей 113 Прикладна математика і 122 Комп'ютерні науки та інформаційні технології (04-01-13). Рівне : НУВГП, 2016. 39 с. URL: <http://ep3.nuwm.edu.ua/4062/> .

11. Рекомендована література

Базова

1. Мартин Грабер. SQL : Изд-во Лори, 2016. 643 с.
2. Томас Коннолли, Каролин Бегг. Базы данных. Проектирование, реализация и сопровождение. Теория и практика : Москва-Санкт-Петербург : Изд-во Диалектика-Вильямс, 2018. 1440 с.

Допоміжна

1. Бен Форта Освой самостоятельно SQL за 10 минут. Издание 4-е, дополненное. К. : Диалектика, 2015. 288 с.

Інформаційні ресурси

1. Рівненська державна обласна бібліотека.
URL: <http://www.libr.rv.ua/> .
2. Наукова бібліотека. URL: <http://www.library.snu.edu.ua/>.
3. Національна бібліотека ім. В.І. Вернадського. URL:
<http://www.nbuv.gov.ua/>.
4. Бібліотека НУВГП . URL: <http://www.nuwm.rv.ua/book.php/>.