МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.
_____Ю. П. Кондратенко
«____»_____2022 р.

БАКАЛАВРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

ВИКОРИСТАННЯ ТЕХНОЛОГІЇ КЛІЄНТ-СЕРВЕР ДЛЯ МОДЕРНІЗАЦІЇ ФАЙЛОВОЇ БАЗИ ДАНИХ PARADOX СТОМАТОЛОГІЧНОЇ КЛІНІКИ

Спеціальність 122 «Комп'ютерні науки»

122 – БКР – 402. 21810220

Виконала студентка 4-го курсу, групи 402
_____ А. О. Паплаускайте
«__» червня 2022 р.

Керівник: канд. фіз.-мат. наук, доцент
_____А. І. Воробйова
«__» червня 2022 р.

Миколаїв – 2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет ім. Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

Рівень вищої освіти **бакалавр**
Спеціальність **122 «Комп'ютерні науки»**
(шифр і назва)
Галузь знань **12 «Інформаційні технології»**
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри інтелектуальних інформаційних систем, д-р техн. наук, проф.
_____ Ю. П. Кондратенко
«____» _____20____р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

Видано студентці групи 402 факультету комп'ютерних наук Паплаускайте Анастасії Олександрівні

1. Тема кваліфікаційної роботи «Використання технології клієнт-сервер для модернізації файлової бази даних Paradox стоматологічної клініки».

Керівник роботи Воробйова Алла Іванівна, канд. фіз.-мат. наук, доцент

Затв. наказом Ректора ЧНУ ім. Петра Могили від «07» 12 2021 р. № 318

2. Строк представлення кваліфікаційної роботи студентом «__» _____ 20__ р.

3. Очікуваний результат роботи: програмний застосунок для міграції СКБД Paradox у SQL Server.

4. Перелік питань, що підлягають розробці (зміст пояснювальної записки):
- огляд основних стратегій планування задач;
- аналіз сучасних систем тайм-менеджменту;
- вибір інструментальних засобів створення застосунку;
- програмна реалізація веб-системи для планування задач із підтримкою групових заходів.

5. Перелік графічного матеріалу: сторінок – 83 , таблиць – 0 , рисунків – 16, посилань – 23 , презентація.

6. Завдання до спеціальної частини: «Охорона праці при користуванні екранними пристроями».

7. Консультанти:

Розділ	Прізвище, ініціали та посада консультанта	Підпис
Спеціальна частина з охорони праці	Алексєєва А.А., канд. техн. наук, доцент	

Керівник роботи ___канд. фіз.-мат. наук, доцент Воробйова А. І._____
(наук. ступінь, вчене звання, прізвище та ініціали)

_____
(підпис)

Завдання прийнято до виконання _Паплаускайте А. О._____
(прізвище та ініціали)

_____
(підпис)

Дата видачі завдання «_23_» ___листопада_____ 2021 р.

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

Тема: «Використання технології клієнт-сервер для модернізації файлової бази даних Paradox стоматологічної клініки»

№	Найменування роботи	Початок	Закінчення	Примітки
1	Подання заяви на затвердження теми та керівників БКР	01.11.2021	01.11.2021	виконано
2	Отримання завдання на виконання БКР	23.11.2021	23.11.2021	виконано
3	Складання календарного плану роботи на весь період виконання БКР	06.12.2021	10.12.2021	виконано
4	Отримання завдання на переддипломну практику	23.05.2022	23.05.2022	виконано
5	Проходження переддипломної практики, збір та аналіз матеріалів до БКР	24.05.2022	02.06.2022	виконано
6	Розробка звіту з переддипломної практики	03.06.2022	04.06.2022	виконано
7	Виконання БКР: аналіз предметної сфери, вибір інструментальних засобів створення застосунку, розробка веб-додатку, тестування	28.02.2022	29.05.2022	виконано
8	Попередній захист БКР на засіданні комісії кафедри	30.05.2022	30.05.2022	виконано
9	Доробка та остаточне оформлення БКР	31.05.2022	11.06.2022	виконано
10	Подання БКР рецензенту	17.06.2022	17.06.2022	виконано
11	Подання БКР, її електронної копії та інших документів (відгуку, рецензії) до захисту	24.06.2022	24.06.2022	
12	Захист БКР перед екзаменаційною комісією (ЕК)	29.06.2022	29.06.2022	

Розробив студент _____Паплаускайте А. О._____ _____
(прізвище та ініціали) (*підпис*)

Керівник роботи канд. фіз.-мат. наук, доцент Воробйова А. І. _____
(наук. ступінь, вчене звання, прізвище та ініціали) *(підпис)*

«____» _____ 202__ р.

АНОТАЦІЯ

бакалаврської кваліфікаційної роботи студентки групи 402 ЧНУ ім. Петра Могили

Паплаускайте Анастасії Олександрівни

Тема: «Використання технології клієнт-сервер для модернізації файлової бази даних Paradox стоматологічної клініки»

Керівник: кандидат фізико-математичних наук, доцент кафедри Інтелектуальних інформаційних систем Воробйова Алла Іванівна.

Об’єкт дослідження – процес створення програмного застосунку для переносу бази даних з одної СКБД на іншу.

Предмет дослідження – засоби для перенесення БД та інструменти для розробки застосунка.

Мета – спрощення процесу зміни СКБД для стоматологічної клініки.

У першому розділі розглядаються та здійснюється аналіз основних понять. У другому розділі – причини для переносу та деталі фігуруючих СКБД. У третьому розділі представлено опис процесу розробки програмного застосунку для виконання переносу даних. В останньому розділі було розглянуто норми та заходи з охорони праці й техніки безпеки при роботі з комп’ютером.

В результаті виконаної роботи було зроблено висновки щодо можливості перенесення великого об’єму даних між різними СКБД.

Сторінок – 83 , таблиць – 0 , рисунків – 16 , посилань – 23 , додатків – 3.

Ключові слова: база даних, система керування базами даних, SQL, Paradox, SQL Server.

ABSTRACT

for bachelor's qualification work of a student of 402 group at Petro Mohyla Black Sea National University

Paplauskaite Anastasiia Oleksandrivna

Topic: " Using client-server technology to modify Paradox file database of dental clinic"

Supervisor: candidate of physical and mathematical sciences, assistant professor of the Department of Intelligent Information Systems Vorobyova Alla Ivanivna

The object of research is the process of creating a software application for transferring a database from one DBMS to another.

The subject of research is tools for database transfer and tools for application development..

The goal is to to simplify the process of transfering the database for a dental clinic.

The first section considers and analyzes the basic concepts. The second section describes the reasons for the transfer and details of the DBMS. The third section describes the process of developing a software application for data transfer. In the last section the norms and measures on labor protection and safety when working with a computer were considered.

As a result of the performed work, conclusions were drawn on the possibility of transferring large amounts of data between different databases.

Pages – 83, tables – 0, figures – 16, links – 23, appendices – 3.

Keywords: database, Database Management System, SQL, Paradox, SQL Server.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ 4

ВСТУП 2

1 ПОНЯТТЯ БАЗИ ДАНИХ, ПРИЧИНИ НЕОБХІДНОСТІ, СИСТЕМИ КЕРУВАННЯ БД, ВИДИ ТА ПРИКЛАДИ ВИКОРИСТАННЯ 4

1.1 Поняття БД та СКБД 4

1.2 Структура найпростішої бази даних 9

1.3 Основні моделі даних 12

1.4 Об’єктно-орієнтовані СКБД 19

1.5 Microsoft Access як приклад СКБД 20

Висновки до розділу 1 24

2 ПРИЧИНИ ЗАСТАРІЛОСТІ БД, Paradox, ПЕРЕНОС ДАНИХ МІЖ РІЗНИМИ СКБД, ПЕРЕВАГИ БД MICROSOFT SQL SERVER 25

2.1 Особливості СКБД Paradox 25

2.2 Причини для зміни СКБД 27

2.3 Особливості СКБД Microsoft SQL Server 27

2.4 Перенесення даних між СКБД 31

Висновки до розділу 2 34

3 РОЗРОБКА ЗАСТОСУНКУ ДЛЯ ПЕРЕНЕСЕННЯ ДАНИХ 35

3.1 Підготовка до роботи 35

3.2 Опис програмної реалізації застосунка 35

3.3 Користувацький інтерфейс 38

Висновки до розділу 3 45

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ 48

4. ОХОРОНА ПРАЦІ 49

ВСТУП 49

4.1 Аналіз виробничого травматизму 50

4.2. Освітлення робочого місця та робочого приміщення 52

4.3 Вимоги до організації приміщення 53

4.4 Протипожежний захист 56

4.5. Техніка безпеки при роботі з ПК 57

ВИСНОВКИ ДО РОЗДІЛУ 4 64

ВИСНОВКИ 65

СПИСОК ВИКОРИСТАННИХ ДЖЕРЕЛ 66

ДОДАТОК А 68

ДОДАТОК Б 71

ДОДАТОК В 73

ПЕРЕЛІК СКОРОЧЕНЬ

DDL – Data definition language
DML – Data manipulation language
IEC – International Electrotechnical Commission
ISO – International Organization for Standardization
SQL – Structured Query Language

БД – база даних
ПО – програмне забезпечення
РБД – реляційні бази даних
СКБД – система керування базами даних

Пояснювальна записка

до кваліфікаційної роботи

на тему:

«ВИКОРИСТАННЯ ТЕХНОЛОГІЇ КЛІЄНТ-СЕРВЕР ДЛЯ МОДЕРНІЗАЦІЇ ФАЙЛОВОЇ БАЗИ ДАНИХ PARADOX СТОМАТОЛОГІЧНОЇ КЛІНІКИ»

Спеціальність 122 «Комп'ютерні науки»

122 – БКР – 402.21810220

Виконала студентка 4-го курсу, групи 402

_____*А. О. Паплаускайте*

(підпис, ініціали та прізвище)

«___» _____ 2022 р.

Керівник: *канд. фіз.-мат. наук, доцент*

(наук. ступінь, вчене звання)

_____*А.І. Воробйова*

(підпис, ініціали та прізвище)

«___» _____ 2022 р.

Миколаїв – 2022

ВСТУП

Застарілість бази даних – сьогодні велика проблема для багатьох підприємців які почали свою діяльність досить давно. Взагалі база даних це дуже важлива і незамінима частина будь якої роботи з великими об’ємами даних, яких у кожному бізнесі завжди багато.

В даний час реляційні системи керування базами даних (СКБД) є важливим інструментом в багатьох областях, починаючи з таких традиційних областей застосування, як бізнес, наукове дослідження, освіта, і закінчуючи розробкою пошукових серверів в Internet. Проте, не зважаючи на важливість наявності хорошої бази даних для ведення інформаційних ресурсів і доступу до них, багато організацій не застосовують їх в своїй роботі. Історично склалося так, що СКБД коштували дуже дорого, а продавці встановлювали дуже високі ціни як на програмне забезпечення, так і на свої послуги з технічної підтримки. Крім того, механізми СКБД вимагали задоволення вимог по продуктивності від апаратних платформ, що ще більше підвищувало вартість таких рішень.

Побудована належним чином база даних забезпечує доступ до оновлених і точних відомостей. Оскільки правильна структура є необхідною умовою для досягнення поставленої мети під час роботи з базою даних, доцільним буде вивчення принципів правильної побудови бази даних. Це дозволить створити таблицю, яка відповідатиме вашим потребам, і яку можна легко змінювати.

Мета роботи – перенесення даних стоматологічної клініки з максимальним збереженням на більш новітню базу даних.

Об’єкт дослідження – засоби для перенесення БД та інструменти для розробки застосунка.

Предмет дослідження – засоби проектування та розробки веб-застосунків для ефективного планування й обліку поставлених задач та цілей.

Досягнення поставленої мети обумовлює необхідність вирішення наступних **задач**.

1) Аналіз предметної області, пошук теоретичних відомостей, особливості БД. Аналіз предметної області теми. Поняття СКБД, БД та загальні положення.

2) Обґрунтувати вибір інструментальних засобів розробки. У другому розділі проаналізуємо та розглянемо причини застарілості використовуваної СКБД, а саме Paradox та переваги новітньої Microsoft SQL Server, причини переходу.

3) Створення програми переносу. У третьому розділі розглянемо та створимо програмну реалізацію яка буде виконувати перенесення великого об’єму даних.

1 ПОНЯТТЯ БАЗИ ДАНИХ, ПРИЧИНИ НЕОБХІДНОСТІ, СИСТЕМИ КЕРУВАННЯ БД, ВИДИ ТА ПРИКЛАДИ ВИКОРИСТАННЯ

1.1 Поняття БД та СКБД

База даних (англ. database) – сукупність даних, організованих відповідно до концепції, яка описує характеристику цих даних і взаємозв'язки між їх елементами; ця сукупність підтримує щонайменше одну з областей застосування (за стандартом ISO/IEC 2382:2015[1]). В загальному випадку база даних містить схеми, таблиці, подання, збережені процедури та інші об'єкти. Дані у базі організовують відповідно до моделі організації даних. Таким чином, сучасна база даних, крім самих даних, містить їх опис та може містити засоби для їх обробки.

1960-ті рр. розроблення перших БД. CODASYL — мережева модель даних та одночасно незалежне розроблення ієрархічної БД фірмою North American Rockwell, яка пізніше взята за основу IMS — власної розробки IBM.

1970-ті рр. наукове обґрунтування Едгаром Ф. Коддом основ реляційної моделі, котра на початку зацікавила лише наукові кола. Уперше цю модель було використано у БД Ingres (Берклі) та System R (IBM), що були лише дослідними прототипами, анонсованими протягом 1976 року.

1980-ті рр. поява перших комерційних версій реляційних БД Oracle та DB2. Реляційні БД починають успішно витісняти мережеві та ієрархічні. Дослідження децентралізованих (розподілених) систем БД, проте вони не відіграють особливої ролі на ринку БД.

1990-ті рр. увага науковців спрямовується на об'єктно-орієнтовані БД, які знайшли застосування в першу чергу в тих галузях, де використовуються комплексні дані: інженерні, мультимедійні БД.

2000-ні рр. головним нововведенням є підтримка та застосування XML у БД. Розробники комерційних БД, які панували на ринку у 1990-их рр.,

отримують все більшу конкуренцію з боку руху відкритого програмного забезпечення. Реакцією на це стає поява безкоштовних версій комерційних БД.

У сучасних інформаційних системах для забезпечення роботи з базами даних використовують системи керування базами даних (СКБД). Система керування базами даних — це система, заснована на програмних та технічних засобах, яка забезпечує визначення, створення, маніпулювання, контроль, керування та використання баз даних (за стандартом ISO/IEC 2382:2015[2]). Застосунки для роботи з базою даних можуть бути частиною СКБД або автономними. Найпопулярнішими СКБД є MySQL, PostgreSQL, Microsoft SQL Server, Oracle, Sybase, Interbase, Firebird та IBM DB2. СКБД дозволяють ефективно працювати з базами даних, обсяг яких робить неможливим їх ручне опрацювання.

Через тісний зв'язок баз даних з СКБД під терміном «база даних» інколи необґрунтовано та неточно мають на увазі систему керування базами даних. Але варто розрізняти базу даних — сховище даних, та СКБД — засоби для роботи з базою даних. СКБД з інформаційної системи може бути видалена, але база даних продовжить існувати. І навпаки: СКБД може функціонувати без жодної бази даних.

В загальному базу даних неможливо просто перемістити з однієї СКБД до іншої. Але СКБД використовують стандарти (SQL, ODBC, JDBC), які уніфікують ряд операцій по роботі з даними і дозволяють різним застосункам працювати з базами даних різних СКБД. СКБД часто класифікують за моделлю організації даних. Найуживаніші СКБД використовують реляційну модель, у якій дані подають у виді таблиць. Для кінцевого користувача (та прикладних програм) робота з базою даних напряму неможлива. Всі маніпуляції над даними здійснюють через спеціальні запити, які надсилають до СКБД. СКБД опрацьовує їх і повертає результат. Безпосередньо з базою даних працює виключно СКБД.

Сучасні СКБД забезпечують функції щодо керування даними, які можна поділити на такі групи:

- Оголошення даних. Створення, зміна та видалення визначень, які описують організацію даних;
- Модифікація даних. Додавання даних, їх редагування та видалення;
- Отримання даних. Надання даних за запитом застосунку у формі, яка дозволяє їх безпосереднє використання. Дані можуть надаватись або у формі, в якій вони зберігаються у базі даних, або в іншій формі (наприклад, через поєднання різних даних);
- Адміністрування даних. Реєстрування та відслідковування дій користувачів, дотримання безпеки роботи з даними, забезпечення надійності та цілісності даних, моніторинг продуктивності, резервне копіювання та відновлення даних тощо.

Архітектура «файл-сервер» передбачає виділення однієї з машин мережі як головної (сервер). На такій машині зберігається спільна централізована БД. Усі інші машини мережі виконують функції робочих станцій, за допомогою яких підтримується доступ користувацької системи до бази даних. Файли бази даних відповідно до призначених для користувача запитів передаються на робочі станції, де в основному і проводиться обробка даних. При великій інтенсивності доступу до одних і тих же даних продуктивність інформаційної системи різко падає. Користувачі також можуть створювати на робочих станціях локальні БД, які використовуються ними монопольно.

У сучасних мережевих інформаційних системах для роботи із загальною базою даних використовують архітектуру «клієнт-сервер». При цьому в мережі розміщують сервер баз даних. Ним виступає комп'ютер (або комп'ютери), який містить бази даних, СКБД та пов'язане з ними програмне забезпечення, і налаштований для надання користувачам інформаційної системи доступу до бази даних. Клієнти, які працюють із даними (вони можуть бути розташовані на

різних комп'ютерах мережі), надсилають відповідні запити серверу. Сервер їх отримує, опрацьовує, та надсилає відповідь клієнту. Сучасні СКБД (MySQL, PostgreSQL, Microsoft SQL Server та інші) працюють відповідно до цієї архітектури. Сервер баз даних, як правило, є достатньо потужною багатопроцесорною системою, яка використовує масиви дисків RAID для підвищення надійності зберігання даних. Використання дискових масивів RAID дозволяє відновити дані, навіть якщо один з дисків вийшов з ладу. Бази даних класифікують за різними критеріями. За моделлю організації даних розрізняють такі бази даних:

- Ієрархічна. Ієрархічна база даних може бути представлена як дерево, що складається з об'єктів різних рівнів. Між об'єктами існують зв'язки типу «предок-нащадок». При цьому можлива ситуація, коли об'єкт не має нащадків або має їх декілька, тоді як у об'єкта-нащадку обов'язково тільки один предок;
- Мережна. Така база даних подібна до ієрархічної, за винятком того, що кожен об'єкт може мати більше одного предку;
- Реляційна. Реляційна база даних зберігає дані у вигляді таблиць. Найуживаніші СКБД використовують реляційну модель даних;
- Об'єктно-орієнтована. У базі даних цього виду дані оформляють у вигляді моделей об'єктів.

За розміщенням даних виділяють такі види баз:

- Локальна, або централізована. Така база даних підтримується на одному комп'ютері;
- Розподілена. Частини такої бази даних розміщують на різних комп'ютерах мережі.

За технологією фізичного зберігання виділяють:

- БД у вторинній пам'яті (традиційні);
- БД в оперативній пам'яті (in-memory database);

– БД у третинній пам'яті (tertiary database).

При роботі з базами даних використовують мови спеціального призначення:

– Мова визначення даних (DDL) — це мова, яка описує дані та структури даних, а також визначає взаємозв'язки між ними (за стандартом ISO/IEC 2382:2015[3]);
– Мова маніпулювання даними (DML) — це мова, яку підтримує СКБД і яка забезпечує виконання операцій отримання, додавання, зміни та видалення даних (за стандартом ISO/IEC 2382:2015[4]);
– Мова запитів (Query language) — це мова для користувачів, яка забезпечує отримання та оброблення даних у базі даних (за стандартом ISO/IEC 2382:2015[5]).

При роботі з реляційними базами даних використовують мову структурних запитів SQL (Structured Query Language)[1], яка поєднує всі три функції (визначення даних, модифікація даних та формування вибірок). Мова SQL стандартизована ANSI та ISO: починаючи з 1986 року, регулярно виходять поновлені стандарти. Слід зауважити, що кожна сучасна СКБД (MySQL, PostgreSQL, Microsoft SQL Server та інші) підтримує свою власну видозміну SQL, так що SQL-запит для однієї СКБД може не працювати в середовищі іншої. Але головні принципи формування SQL-запитів та їх структура однакові та відповідають стандартам ANSI/ISO. При необхідності виконання якоїсь операції над даними клієнт формує лінгвістичну конструкцію мовою SQL, яку називають SQL-запитом, і надсилає її до СКБД. СКБД опрацьовує запит, і результат його виконання (наприклад, вибірку даних) повертає клієнту. Мова, якою оперує СКБД, також може містити засоби для конфігурування СКБД, модифікації, форматування даних та розрахунків, формування обмежень даних.

Первинним призначенням бази даних є зберігання масивів даних. Але їх широко використовують і для збереження адміністративної інформації та

спеціалізованих даних, наприклад, для інженерних даних чи для економічних моделей. Прикладами використання баз даних можуть бути:

- автоматизовані системи обліку;
- реєстри та каталоги;
- геоінформаційні системи;
- лінгвістичні бази даних, тобто машинні словники різного типу і призначення;
- бази даних транспортних систем;
- системи керування вмістом Інтернет-сайтів, які зберігають у базах даних інформацію про web-сторінки сайту.

1.2 Структура найпростішої бази даних

Якщо в базі немає ніяких даних (порожня база), це все рівно повноцінна база даних. Хоча даних у базі і немає, але інформація в ній усе-таки є — це структура бази. Вона визначає методи занесення даних і збереження їх у базі. Найпростіший “некомп'ютерний” варіант бази даних — діловий щоденник, у якому кожному календарному дню виділено по сторінці. Навіть якщо в ньому не записано ні рядка, він не перестає бути щоденником, оскільки має структуру, що чітко відрізняє його від записних книжок, робочих зошитів і іншої паперової продукції.

Структуру двовимірної таблиці утворюють стовпці і рядки. Їхніми аналогами в структурі найпростішої бази даних є поля і записи. Якщо записів у таблиці немає, то її структура утворена тільки набором полів. Змінивши склад полів базової таблиці (чи їх властивості), ми змінюємо структуру бази даних і, відповідно, одержуємо нову базу даних[2].

Властивості полів бази даних. Поля бази даних не просто визначають структуру бази — вони ще визначають групові властивості даних, записуваних в

осередки, що належать кожному з полів. Нижче перераховані основні властивості полів таблиць баз даних на прикладі СКБД Microsoft Access.

- Ім'я поля — визначає, як варто звертатися до даних цього поля при автоматичних операціях з базою (за замовчуванням імена полів використовуються як заголовки стовпців таблиць);
- Тип поля — визначає тип даних, що можуть міститися в даному полі;
- Розмір поля — визначає граничну довжину (у символах) даних, що можуть розміщатися в даному полі;
- Формат поля — визначає спосіб форматування даних в осередках, що належать полю;
- Маска введення — визначає форму, у якій вводяться дані в поле (засіб автоматизації введення даних);
- Підпис — визначає заголовок стовпця таблиці для даного поля (якщо підпис не зазначений, то як заголовок стовпця використовується властивість Ім'я поля);
- Значення за замовчуванням — те значення, що вводиться в осередки поля автоматично (засіб автоматизації введення даних);
- Умова на значення — обмеження, використовуване для перевірки правильності введення даних (засіб автоматизації введення, що використовується, як правило, для даних, що мають числовий тип, грошовий тип чи тип дати);
- Повідомлення про помилку — текстове повідомлення, що видається автоматично при спробі введення в поле помилкових даних (перевірка помилковості виконується автоматично, якщо задана властивість Умова на значення);
- Обов'язкове поле — властивість, що визначає обов'язковість заповнення даного поля при наповненні бази;

- Порожні рядки — властивість, що дозволяє введення порожніх строкових даних (від властивості Обов'язкове поле відрізняється тим, що відноситься не до всіх типів даних, а лише до деяким, наприклад до текстового);
- Індексоване поле — якщо поле має цю властивість, усі операцій, зв'язані з пошуком чи сортуванням записів за значенням, що зберігається в даному полі, істотно прискорюються. Крім того, для індексованих полів можна зробити так, що значення в записах будуть перевірятися по цьому полю на наявність повторів, що дозволяє автоматично виключити дублювання даних.

В основі організації бази даних є модель даних, яка визначає правила, у відповідності з якими структуруються дані. За допомогою моделі представляється велика кількість даних і описуються взаємозв'язки між ними. Найбільш поширені такі моделі даних: ієрархічна, мережева, реляційна.

В ієрархічній моделі зв'язок даних "один до одного" (1:1) означає, що кожному значенню (екземпляру) елемента даних А відповідає одне і тільки одне значення, пов'язаного з ним елемента В. Наприклад, поміж такими елементами пар даних, як код готової продукції і її найменуванням є вищезазначений зв'язок, так як тільки кожному коду продукції відповідає одне її найменування.

В мережевій моделі зв'язок "один до багатьох" (1:В) означає, що значенню елемента А відповідають багато (більше одного) значень, пов'язанню з ним елементів В. Наприклад, поміж елементами даних "код виробу" (елемент А) і "кодом матеріалів" (елементи В) існує такий взаємозв'язок бо на виготовлення одного виробу використовується багато різних матеріалів.

В реляційній моделі зв'язок "багатьох до багатьох" (В:В) указує на те, що декільком значенням елементів даних А відповідає декілька значені елементів даних В. Наприклад, поміж елементами даних "код операції технологічного процесу" і "табельний номер працівника" існує зазначені взаємозв'язок, так як багато операцій технологічного процесу можуть виконувати різні працівники

(табельні номери) і навпаки. На персональних ЕОМ, як правило, використовуються реляційні РБД. Концепція РБД розроблена Е. Коддом у 1970 р. В її основу покладено математичне поняття “відношення” (від англ. relation). Відношення представляються у вигляді двовимірних таблиць. Наприклад, в таблиці, зображеній на рис.1, представлена реляційна база даних, в якій містяться відомості про абонентів телефонної мережі.

1.3 Основні моделі даних

Якщо в базі немає ніяких даних (порожня база), це все рівно повноцінна база даних. Хоча даних у базі і немає, але інформація в ній усе-таки є — це структура бази. Вона визначає методи занесення даних і збереження їх у базі. Найпростіший “некомп'ютерний” варіант бази даних — діловий щоденник, у якому кожному календарному дню виділено по сторінці. Навіть якщо в ньому не записано ні рядка, він не перестає бути щоденником, оскільки має структуру, що чітко відрізняє його від записних книжок, робочих зошитів і іншої паперової продукції[3].

Структуру двовимірної таблиці утворюють стовпці і рядки. Їхніми аналогами в структурі найпростішої бази даних є поля і записи. Якщо записів у таблиці немає, то її структура утворена тільки набором полів. Змінивши склад полів базової таблиці (чи їх властивості), ми змінюємо структуру бази даних і, відповідно, одержуємо нову базу даних.

Властивості полів бази даних. Поля бази даних не просто визначають структуру бази — вони ще визначають групові властивості даних, записуваних в осередки, що належать кожному з полів. Нижче перераховані основні властивості полів таблиць баз даних на прикладі СКБД Microsoft Access.

- Ім'я поля — визначає, як варто звертатися до даних цього поля при автоматичних операціях з базою (за замовчуванням імена полів використовуються як заголовки стовпців таблиць).

- Тип поля — визначає тип даних, що можуть міститися в даному полі.
- Розмір поля — визначає граничну довжину (у символах) даних, що можуть розміщатися в даному полі.
- Формат поля — визначає спосіб форматування даних в осередках, що належать полю.
- Маска введення — визначає форму, у якій вводяться дані в поле (засіб автоматизації введення даних).
- Підпис-визначає заголовок стовпця таблиці для даного поля (якщо підпис не зазначений, то як заголовок стовпця використовується властивість Ім'я поля).
- Значення за замовчуванням — те значення, що вводиться в осередки поля автоматично (засіб автоматизації введення даних).
- Умова на значення — обмеження, використовуване для перевірки правильності введення даних (засіб автоматизації введення, що використовується, як правило, для даних, що мають числовий тип, грошовий тип чи тип дати).
- Повідомлення про помилку — текстове повідомлення, що видається автоматично при спробі введення в поле помилкових даних (перевірка помилковості виконується автоматично, якщо задана властивість Умова на значення).
- Обов'язкове поле — властивість, що визначає обов'язковість заповнення даного поля при наповненні бази;
- Порожні рядки — властивість, що дозволяє введення порожніх строкових даних (від властивості Обов'язкове поле відрізняється тим, що відноситься не до всіх типів даних, а лише до деяким, наприклад до текстового).
- Індексоване поле — якщо поле має цю властивість, усі операцій, зв'язані з пошуком чи сортуванням записів за значенням, що зберігається в даному полі, істотно прискорюються. Крім того, для індексованих полів можна зробити так,

що значення в записах будуть перевірятися по цьому полю на наявність повторів, що дозволяє автоматично виключити дублювання даних.

В основі організації бази даних є модель даних, яка визначає правила, у відповідності з якими структуруються дані. За допомогою моделі представляється велика кількість даних і описуються взаємозв'язки між ними.

Найбільш поширені такі моделі даних: ієрархічна, мережева, реляційна.

В ієрархічній моделі зв'язок даних "один до одного" (1:1) означає, що кожному значенню (екземпляру) елемента даних А відповідає одне і тільки одне значення, пов'язаного з ним елемента В. Наприклад, поміж такими елементами пар даних, як код готової продукції і її найменуванням є вищезазначений зв'язок, так як тільки кожному коду продукції відповідає одне її найменування.

В мережевій моделі зв'язок "один до багатьох" (1:В) означає, що значенню елемента А відповідають багато (більше одного) значень, пов'язанню з ним елементів В. Наприклад, поміж елементами даних "код виробу" (елемент А) і "кодом матеріалів" (елементи В) існує такий взаємозв'язок бо на виготовлення одного виробу використовується багато різних матеріалів.

В реляційній моделі зв'язок "багатьох до багатьох" (В:В) указує на те, що декільком значенням елементів даних А відповідає декілька значені елементів даних В. Наприклад, поміж елементами даних "код операції технологічного процесу" і "табельний номер працівника" існує зазначені взаємозв'язок, так як багато операцій технологічного процесу можуть виконувати різні працівники (табельні номери) і навпаки.

На персональних ЕОМ, як правило, використовуються реляційні РБД.

Концепція РБД розроблена Е. Коддом у 1970 р. В її основу покладено математичне поняття “відношення” (від англ. relation). Відношення представляються у вигляді двовимірних таблиць. Наприклад, в таблиці, зображеній на рис.1, представлена реляційна база даних, в якій містяться відомості про абонентів телефонної мережі.

Ієрархічна модель даних

Ієрархічна модель дозволяє будувати БД з деревовидної структурою. У них кожен вузол містить свій тип даних (сутність). На верхньому рівні дерева є один вузол – корінь, на наступному рівні розташовуються вузли, пов'язані з цим коренем, потім вузли, пов'язані з вузлами попереднього рівня, і т. Д. Кожен вузол може мати тільки одного предка (Рис. 1.1).

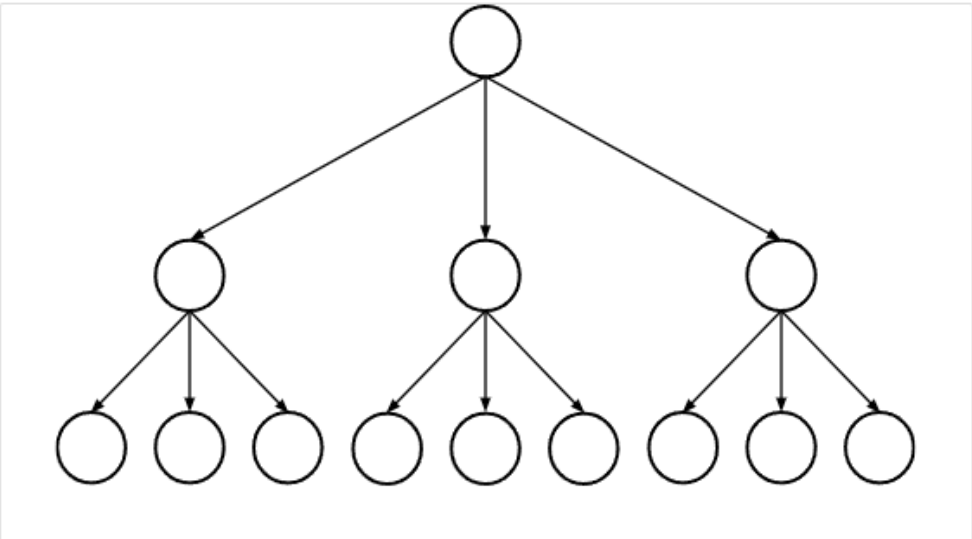

Рисунок 1.1 – Схема ієрархічної моделі даних

Пошук даних завжди починається з кореня і йде у напрямку вниз, поки не знайдений потрібний об'єкт.

Переваги:

- Простота опису ієрархічних структур реального світу;
- Швидке виконання запитів, відповідних структурі даних.

Недоліки:

- Такі БД часто містять надлишкові дані;
- Не завжди зручно кожен раз починати пошук потрібних даних з кореня.

Далі можна побачити приклад ієрархічної моделі (Рис. 1.2).

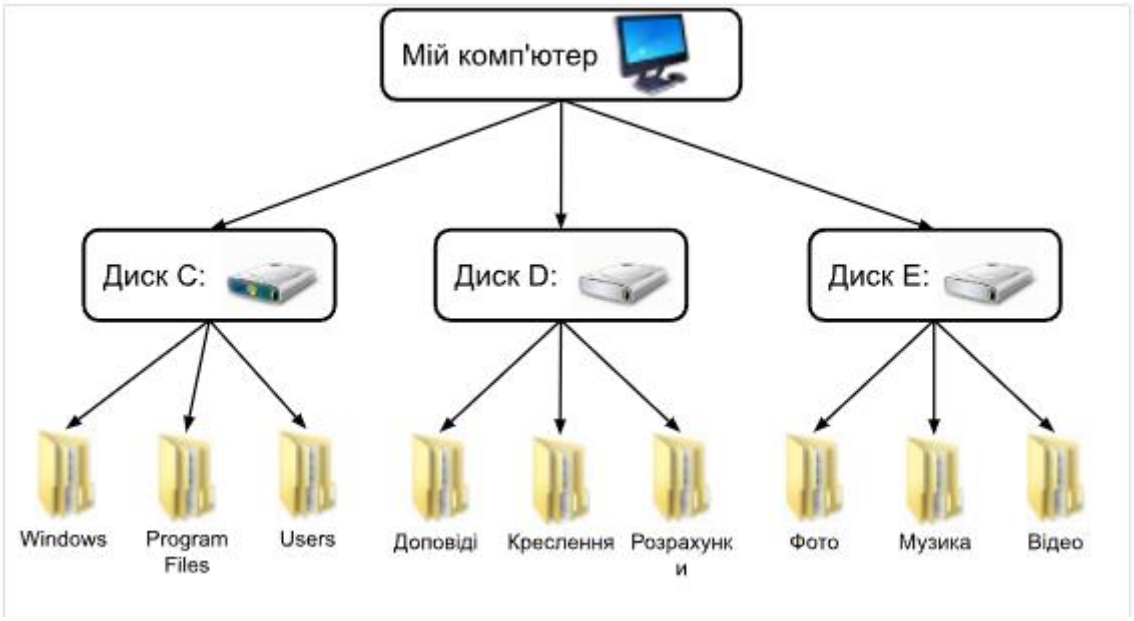

Рисунок 1.2 – Приклад ієрархічної структури БД

Мережева модель

Мережева модель є розширенням ієрархічної. В ієрархічних структурах запис-нащадок повинен мати тільки одного предка; в мережевій структурі даних нащадок може мати будь-яке число предків (Рис. 1.3).

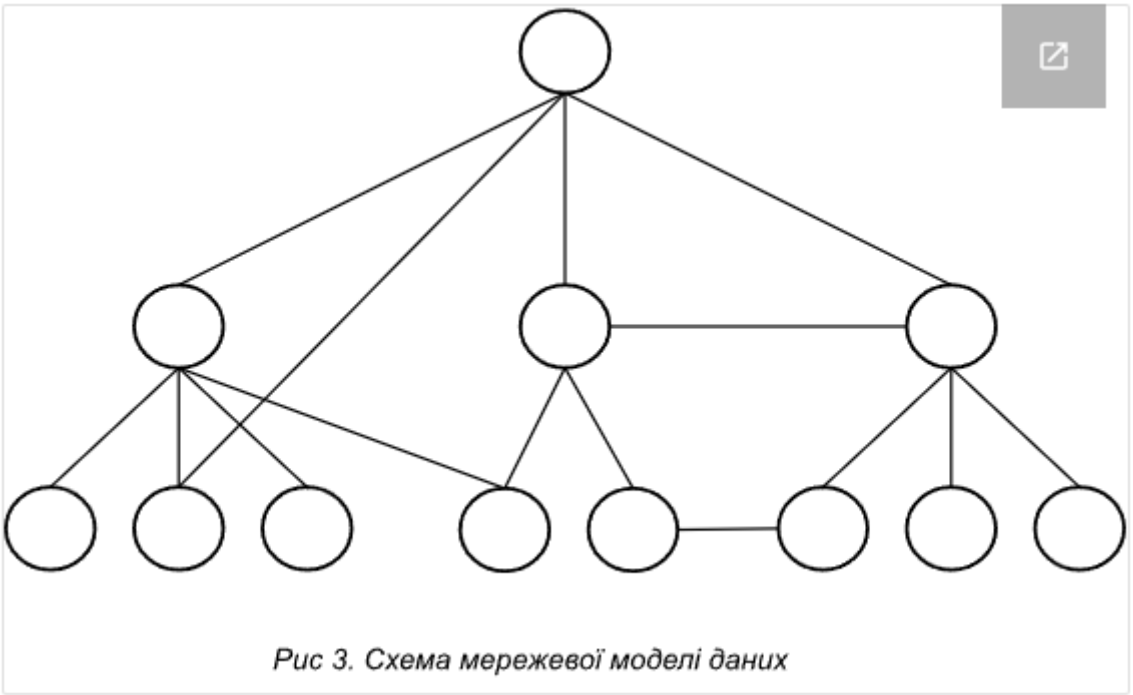

Рисунок 1.3 – Схема мережевої моделі даних

Переваги: ефективна реалізація за показниками витрат пам'яті та оперативності.

Недолік: висока складність і жорсткість схеми БД, побудованої на її основі.

Таким чином, в мережевий моделі можливі зв'язки всіх інформаційних об'єктів з усіма. Наприклад, кожен викладач може навчати багато студентів і кожен студент може навчатися у багатьох викладачів (Рис. 1.4).

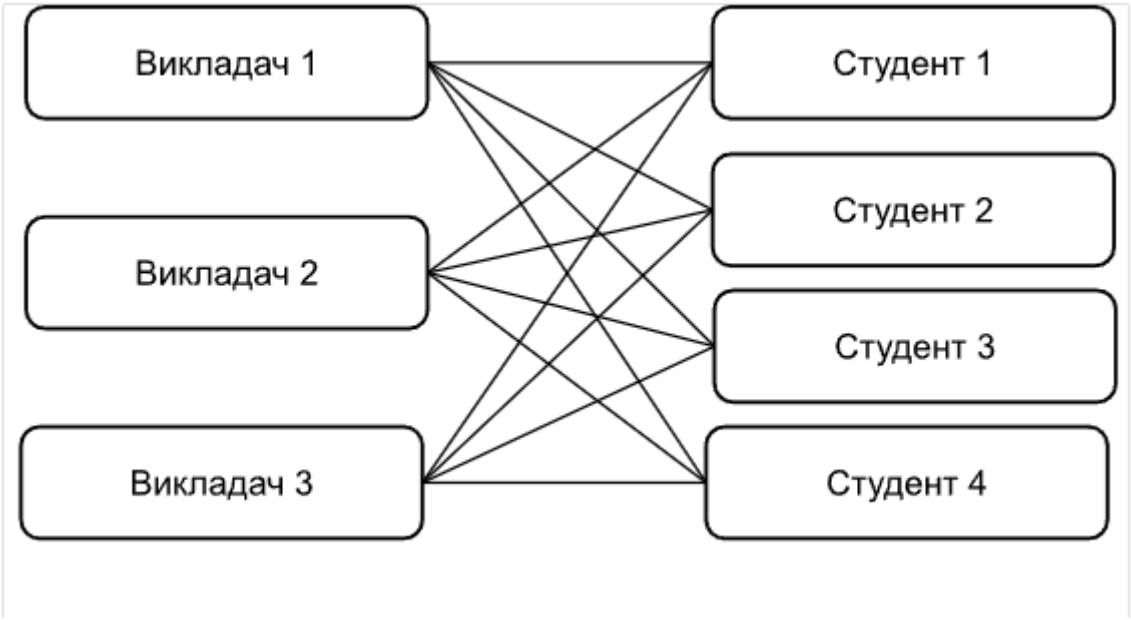

Рисунок 1.4 – Приклад мережевої структури моделі БД

Реляційна модель

Реляційна модель даних являє собою сховище даних, які організовані у вигляді двовимірних таблиць (Рис. 1.5).

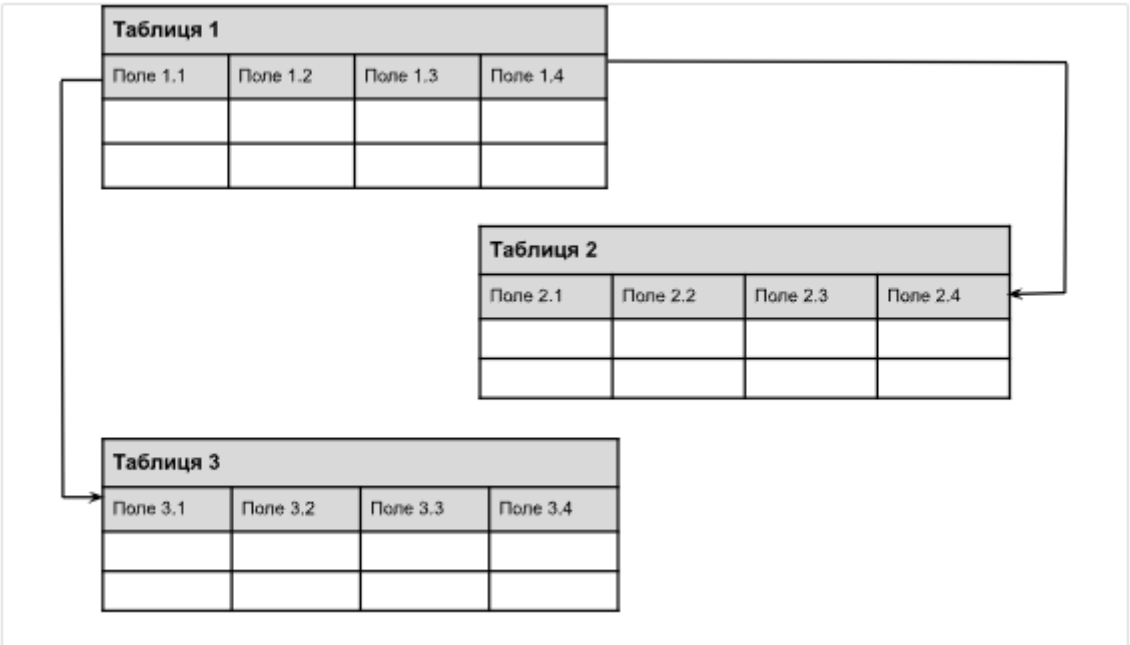

Рисунок 1.5 – Схема реляційної моделі даних

Будь таблиця реляційної бази даних складається з рядків, які називаються записами, і стовпців, які називаються полями. На перетині стовпця і рядка знаходяться конкретні значення даних, що містяться в таблиці (Рис. 1.6).

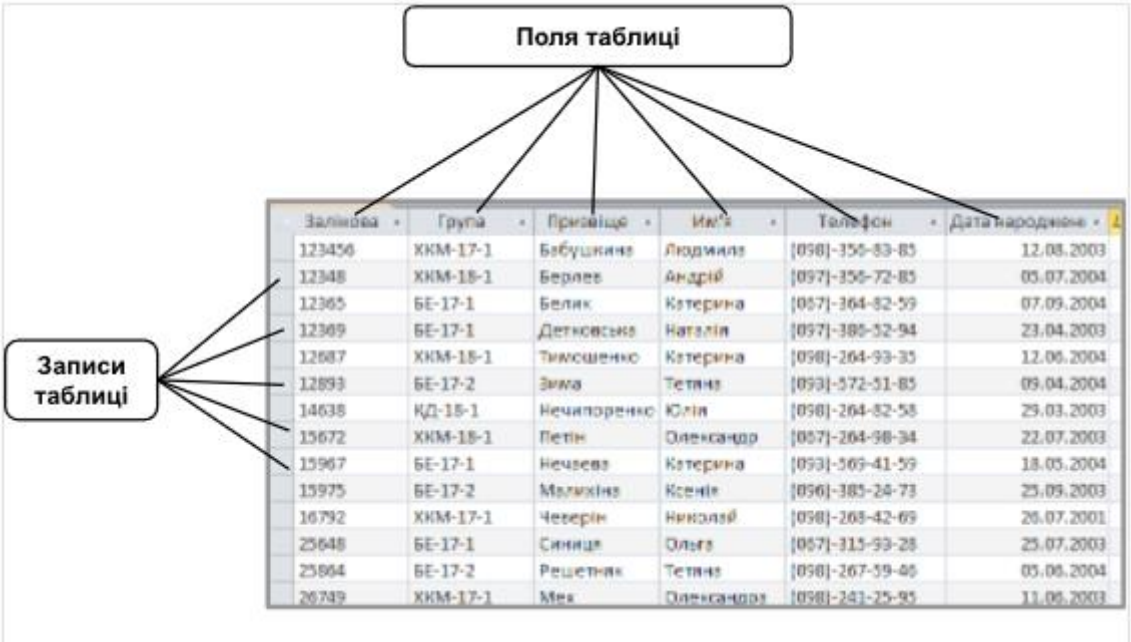

Рисунок 1.6 – Таблиця бази даних

1.4 Об’єктно-орієнтовані СКБД

В принципі, ООСКБД - це об'єктно-орієнтована база даних, що надає можливості СКБД об'єктам, що були створені з використанням об’єктно орієнтованої мови програмування. Прикладні програми, що використовують об'єктно-орієнтовану базу даних, пишуться на об'єктно-орієнтованої мови програмування, а можливості СКБД існують завдяки спеціальному API, в якому є клас бази даних або базовий клас Persistent, в яких реалізовані функції для роботи з даними. ООСКБД дозволяє працювати з об'єктами баз даних так само, як з об'єктами в програмуванні на об'єктно-орієнтованих мовах. ООСКБД розширює мови програмування, прозоро вводячи довгострокові дані, управління паралелізмом, відновлення даних, асоціативні запити й інші можливості. Деякі об'єктно-орієнтовані бази даних розроблені для щільної взаємодії з такими об'єктно-орієнтованими мовами програмування як Python, Java, C #, Visual Basic .NET, C ++, Objective-C і Smalltalk; інші мають свої власні мови програмування. ООСКБД використовують точно таку ж модель, що і об'єктно-орієнтовані мови програмування. Переваги використання ООСКБД:

- Відсутня проблема невідповідності моделі даних в додатку і БД (impedance mismatch). Всі дані зберігаються в БД в тому ж вигляді, що і в моделі програми.
- Не потрібно окремо підтримувати модель даних на стороні СКБД.
- Всі об'єкти на рівні джерела даних строго типізовані.

1.5 Microsoft Access як приклад СКБД

Access – це система управління базами даних (СКБД). Вона призначена для зберігання і пошуку даних, подання інформації в зручному вигляді й автоматизації часто повторюваних операцій (таких, як ведення рахунків, облік, планування тощо). СКБД Access призначена для створення реляційних баз даних. Створення БД починається з проектування бази даних, тобто опису предметної області. Опис предметної області має охоплювати весь клас сутностей, інформація про яких повинна зберігатися в БД, і забезпечувати вимоги до функцій системи.

Система Access – це набір інструментів кінцевого користувача для управління базами даних. До її складу входять конструктори таблиць, форм, запитів і звітів.

База даних – це комп'ютерний термін, використовуваний для позначення інформації з певної теми або відомостей, пов'язаних з деяким додатком. Зберігання інформації у вигляді бази даних полегшує доступ до неї, пошук та вилучення потрібних фрагментів.

В Access база даних – це загальне сховище даних і відповідних їм об'єктів. Об'єкти бази даних – це таблиці, запити, форми, звіти, макроси і модулі.
Ці об'єкти включають дані та інструментальні засоби, необхідні для використання Access:

- Таблиця. Використовується для зберігання даних. Для відображення даних використовується перегляд в режимі таблиці;
- Запит. Дозволяє здійснювати пошук, сортування та витяг певних даних;
- Форма. Забезпечує можливість введення і відображення даних в заданому форматі;
- Звіт. Дозволяє відображати і друкувати відформатовані дані, включаючи результати обчислень і підсумкові значення;

– Макрос. Включає прості команди для автоматизації виконання завдань без програмування;
– Модуль. Програма, написана мовою Visual Basic for Application (VBA).

Якщо уявити собі велику канцелярську шафу, повну всіляких папок, окремих листів і взагалі паперових обривків із записами будь-якого характеру то цю шафу можна цілком назвати повноцінною базою даних. Більш того, в доком'ютерну еру тільки так воно і було. Однак сама по собі ця шафа ніякої практичної цінності не має. У всякому разі, без кваліфікованого служителя, який би знав, що в шафі лежить, де конкретно воно знаходиться і як швидко можна отримати повну відповідь на питання, що цікавить. Так от, цей служитель укупі з обов'язковою системою організації папок в шафі і є системою управління базою даних, тобто СКБД.

У програмі MS Access вона виглядає наступним чином. Вся вихідна інформація зберігається в чітко визначених таблицях. Під чітким визначенням мається на увазі така структура таблиці, в якій кожен рядок має унікальний ідентифікатор (наприклад, номер рядка), а дані представлені стовпцями.

Таким чином, будь-яка таблиця є одновимірним набором записів. Неодмінною правилом створення таблиці в СКБД є строге визначення вмісту самої таблиці. В її осередках може зберігатися тільки фактична і тільки незмінна інформація. Це може здатися дещо дивним і незвичним для користувачів електронних таблиць, проте ні в MS Access, ні в СКБД взагалі, в осередках базових таблиць принципово не може бути обчислюваних значень.

Це не дуже зручно, проте, на щастя, існує можливість обійти цю заборону. Для цього в СКБД використовуються так звані запити. Власне кажучи, запити – це ті ж самі таблиці, тільки вони заповнюються не вручну, а за допомогою заздалегідь заданих формул та інших залежностей. Таким чином, те, що не можна в таблиці, можна в запиті: складати, віднімати, ділити, виконувати інші математичні або логічні операції, вибирати дані з якої-небудь умови. А свою

назву запити отримали від того, що вони схожі з широко поширеним природним дією аналогічного призначення. Прикладом запиту є така конструкція: хто конкретно купував автомобіль ВАЗ 2109 вишневого кольору в період з січня по грудень 1998 року і розплачувався при цьому готівковою іноземною валютою, в якості якої використовувалися швейцарські франки. Зіткнувшись з подібним запитом, СКБД самостійно перегляне відповідні таблиці, в яких зберігаються фактичні дані, і відбере з них всі рядки, які відповідають вимогам запиту. Причому з самими таблицями нічого не відбувається, в таблицю запиту передаються лише копії цих записів.

Запити стали основним робочим інструментом СКБД, завдяки тому, що в них поєднуються як можливості вибірки інформації з деякого її масиву, так і можливості зміни цієї інформації за допомогою формул. Одночасно з відбором, запити можуть виробляти будь-які розрахунки (наприклад, не тільки показати всіх покупців вишневих дев'яток, але і вказати, скільки кожна продаж принесла доходу, яка виявилася її собівартість, скільки довелося віддати в казну, а скільки залишилося в якості чистого прибутку) і навіть підбивати підсумки. В той же час, формально запити самі можуть бути представлені як таблиці і використані в подальшому в якості джерела даних для інших запитів. Це називається запит за запитом.

Само собою зрозуміло, що запити і таблиці лише тоді будуть жити і трудитися в мирі та злагоді, коли між ними з'являться однозначні і строго певні зв'язки. Ці зв'язки служать основою, на яку спирається вся система СКБД. Теоретично, можна сконструювати таку загальну таблицю, в рамках якої можна уявити все зберігаються в базі дані, проте подібна таблиця виявляється майже повсюдно, по-перше, зайво громіздкою, а по-друге, заповненої повторюваними даними. Через громіздкість нею важко оперувати, а зайві дані збільшують її загальний обсяг, що обертається швидким зростанням вимог до системних ресурсів комп'ютера. Припустимо, база даних містить інформацію про продажі.

Отже, в ній неодмінно стануть присутнім стовпці з даними на самого клієнта. Наприклад, його прізвище, посада, телефон та адресу доставки. А тепер уявімо, що у нашої компанії з'явився постійний клієнт, який зробив протягом року п'ятдесят різнойменних покупок. Згідно з правилами організації таблиці, ці дані будуть повторюватися в кожній її рядку, а значить – вони займуть в сорок дев'ять разів більше місця, ніж їм насправді потрібно. У теж час, застосувавши систему зв'язків, можна створити окремо таблицю для оформлення замовлення і окремо таблицю з реєстром клієнтів. У тому випадку, коли покупець звернувся вперше, реєстр поповнюється новим рядком. Якщо ж він прийшов повторно, то, замість нового запису в реєстрі, в таблиці замовлення ставиться покажчик на вже існуючу рядок реєстру.

Важливою складовою СКБД є форми. Як відомо, існує велика різниця між тим, що виводиться на екран, і тим, що насправді записано на вінчестері. Приміром, на екрані ми бачимо зображення людини, а на жорсткому диску "лежать" тільки певним чином чергуються нулі і одиниці. Так от, ця фактично зберігається інформація є прикладом фактичних даних, а відображається на її основі картинка – формою відображення. Причому від того, наскільки вірно організована форма, залежить, наскільки добре буде сприйнята або введена користувачем сама інформація. Всі знають, як важко буває заповнити навіть елементарний друкований бланк, якщо його поля забезпечені малозрозумілими підписами, та ще розкиданими в довільному порядку по листу.

Таким чином, форми в СКБД служать для зручного представлення інформації виключно для людських очей (наприклад, на паперовій роздруківці або частіше на екрані монітора). Звіти використовуються для представлення даних в зручному вигляді. Звіт можна вивести на екран або частіше роздрукувати на принтері. У ньому можна групувати і сортувати дані в будь-якому порядку. Звіти дозволяють відображати і друкувати дані з будь-яким ступенем деталізації.

Висновки до розділу 1

Таким чином база даних — це певний набір даних, які пов'язані між собою спільною ознакою або властивістю, та впорядковані, наприклад, за алфавітом. Об'єднання великої кількості даних в єдину базу дає змогу для формування безлічі варіації групування інформації — особисті дані клієнта, історія замовлень, каталог товарів та будь-що інше.

Головною перевагою БД є швидкість внесення та використання потрібної інформації. Завдяки спеціальним алгоритмам, які використовуються для баз даних, можна легко знаходити необхідні дані всього за декілька секунд. Також в базі даних існує певний взаємозв'язок інформації: зміна в одному рядку може спричинити зміни в інших рядках — це допомагає працювати з інформацією простіше і швидше.

СКБД — це програмні засоби, які виступають посередником між БД та її користувачами. Завдяки сукупності мовних та програмних засобів, СКБД сприяють створенню, ведення та спільного використання БД різними користувачами.

Сучасна програма СКБД складаються з ядра, процесору мови БД, підсистеми підтримки часу виконання та сервісних програм, які надають додаткові можливості для обслуговування інформаційних систем.

2 ПРИЧИНИ ЗАСТАРІЛОСТІ БД, PARADOX, ПЕРЕНОС ДАНИХ МІЖ РІЗНИМИ СКБД, ПЕРЕВАГИ БД MICROSOFT SQL SERVER

2.1 Особливості СКБД Paradox

Paradox - СКБД, що займає стійкі позиції на ринку засобів розробки настільних додатків з базами даних. Принцип зберігання даних у Paradox подібний до принципів зберігання даних у dBase - кожна таблиця зберігається у своєму файлі (розширення *.db), MEMO- та BLOB-поля зберігаються в окремому файлі (розширення *.md). Формат даних Paradox не є відкритим, тому для доступу до даних цього формату потрібні спеціальні бібліотеки.

Ранні версії Paradox зазвичай надавали розробникам баз даних істотно більш розширені можливості, такі як використання ділової графіки в DOS-додатках, оновлення даних у додатках при розрахованій на багато користувачів роботі, візуальні засоби побудови запитів, на основі інтерфейсу QBE - Query by Example (запит за зразком), засоби статистичного аналізу даних, а також засоби візуальної побудови інтерфейсів користувацьких додатків з автоматичною генерацією коду мовою програмування PAL (Paradox Application Language).

Windows-версії СКБД Paradox, крім перерахованих вище сервісів, дозволяють також маніпулювати даними інших форматів, зокрема dBase та даними, що зберігаються у серверних СКБД. Таку можливість користувачі Paradox отримали завдяки використанню бібліотеки Borland Database Engine та драйверів SQL Links. Це дозволило використовувати Paradox як універсальний засіб управління різними базами даних (істотно полегшена версія Paradox 7 під назвою Database Desktop, як і раніше, входить до складу Borland Delphi і Borland C++Builder саме з цією метою). Що ж до базового формату даних, використовуваного у цьому продукті, він має тими самими недоліками, як і всі формати даних настільних СКБД, і тому за можливості його намагаються

замінити на серверну СКБД, навіть зберігши сам Paradox як розробки додатків і маніпуляції даними.

Таким чином, dBase – це досить простий та універсальний формат таблиць. Це про переваги, з недоліків слід зазначити такі, як обмеження на імена полів у таблицях: максимальна довжина - всього 10 символів, допустимі тільки латинські літери і цифри. Подібні обмеження накладаються на імена самих таблиць, причому, у кращих традиціях 16-бітних систем, довжина імені таблиці має перевищувати 8 символів (хоча це обмеження, загалом, можна ігнорувати під час роботи в Delphi під 32-разрядными системами). Але значно суттєвішими недоліками є відсутність автоматичного контролю цілісності даних або зв'язків, прав доступу, а також обмеження на типи даних, що підтримуються, ключі, індекси і т.д. Усе це робить цей формат недостатньо конкурентоспроможним.

Досвід застосування ЕОМ для побудови прикладних систем обробки даних показує, що найефективнішим інструментом є не універсальні алгоритмічні мови високого рівня, а спеціалізовані мови для створення систем управління даними. Такі кошти зазвичай включаються до складу СКБД, але можуть існувати й окремо. СКБД дають можливість користувачам здійснювати безпосереднє управління даними, а програмістам швидко розробляти досконаліші програмні засоби їхньої обробки. Характеристики готових прикладних пакетів визначаються перш за все прийнятою в СКБД організацією даних та типом транслятора, що використовується. Ім'я поля в таблиці формату Paradox є рядком, написання якого підпорядковується наступним правилам:

– Ім'я має бути не довшим за 25 символів;
– Ім'я не повинно починатися з пробілу, однак може містити пробіли. Однак, якщо Ви передбачаєте в майбутньому переносити базу даних в інші формати, розумніше уникати включення пробілів у назву поля. Фактично, з метою перенесення краще обмежитися дев'ятьма символами в назві поля, не включаючи в нього прогалини;

- Ім'я не повинно містити квадратні, круглі або фігурні дужки [], () або {}, тире, а також комбінацію символів "тире" та "більше" (->);
- Ім'я не повинно бути лише символом #, хоча цей символ може бути в імені серед інших символів. Хоча Paradox підтримує точку (.) в назві поля, краще уникати її, оскільки точка зарезервована в Delphi для інших цілей.

2.2 Причини для зміни СКБД

Виділимо основні причини для зміни СКБД на більш новітні:

- Застарілість підтримуваних мов програмування;
- Виділення малої кількості пам'яті;
- Софт надає менше можливостей проводити операцій чим інший софт;
- Є новіші і більш функціональні СКБД;
- Поліпшення захисту від несанкціонованих втручань;
- Зменшення збоїв в базі даних;
- Покращення швидкості вилучення з бази даних для потреб користувачів;
- Простіша і швидша підтримка від супорту;
- Поліпшений інтерфейс для роботи;
- Спрощення можливостей для тестування.

Усі вони, на жаль, стосуються СКБД Paradox, тому і було прийняте рішення її модернізувати.

2.3 Особливості СКБД Microsoft SQL Server

Microsoft SQL Server — система управління базами даних, яка розробляється корпорацією Microsoft. Як сервер даних виконує головну функцію по збереженню та наданню даних у відповідь на запити інших застосунків, які можуть виконуватися як на тому ж самому сервері, так і у мережі.

Мова, що використовується для запитів — Transact-SQL, створена спільно Microsoft та Sybase. Transact-SQL є реалізацією стандарту ANSI / ISO щодо структурованої мови запитів SQL із розширеннями. Використовується як для невеликих і середніх за розміром баз даних, так і для великих баз даних масштабу підприємства. Багато років вдало конкурує з іншими системами керування базами даних.

Базовий код MS SQL Server (до версії 7.0) ґрунтувався на коді Sybase SQL Server[en]. Це дозволило Microsoft вийти на ринок баз даних для підприємств, де конкурували Oracle, IBM, і, пізніше, сама Sybase. Microsoft, Sybase і Ashton-Tate[en] спочатку об'єдналися для створення і випуску на ринок першої версії програми, що отримала назву SQL Server 1.0 для OS/2 (близько 1989 року), яка фактично була еквівалентом Sybase SQL Server 3.0 для Unix, VMS та ін. Microsoft SQL Server 4.2 був випущений у 1992 році та входив до складу операційної системи Microsoft OS/2 версії 1.3. Офіційний реліз Microsoft SQL Server версії 4.21 для ОС Windows NT відбувся одночасно з релізом самої Windows NT (версії 3.1). Microsoft SQL Server 6.0 був першою версією SQL Server, створеною виключно для архітектури NT і без участі в процесі розробки Sybase.

До того часу, як вийшла на ринок ОС Windows NT, Sybase і Microsoft розійшлися та створювали вже власні моделі цього програмного продукту. Microsoft намагалася отримати виняткові права на всі версії SQL Server для Windows. Пізніше Sybase змінила назву свого продукту на Adaptive Server Enterprise[en] щоб уникнути плутанини з Microsoft SQL Server. До 1994 року Microsoft отримала від Sybase три повідомлення про авторські права як натяк на походження Microsoft SQL Server.

Після розділення компанії зробили декілька самостійних релізів програм. SQL Server 7.0 був першим сервером баз даних зі справжнім графічним інтерфейсом адміністрування. Для усунення претензій з боку Sybase у порушенні авторських прав, весь успадкований код в сьомій версії був переписаний. Це

забезпечило також й успіх SQL Server 2000, який був першою редакцією, орієнтованою на архітектуру IA-64.

Протягом подальших шести років корпорація Microsoft працювала над вдосконаленням вже існуючої версії SQL Server 2000 доки не збудувала зручнішу систему Microsoft SQL Server 2005. Були вдосконалені продуктивність, клієнтські інструменти інтегрованого середовища розробки, а також у декількох додаткових системах, що встановлюються разом із SQL Server 2005. Змінено: інструментарій процесів керування сховищами даних (SQL Server Integration Services або SSIS), сервер звітів, сервер OLAP та інтелекутального аналізу даних (Analysis Services[en]), а також декілька технологій повідомлень, особливо Service Broker та Notification Services.

Microsoft SQL Server як мову запитів використовує версію SQL, що отримала назву Transact-SQL (скорочено T-SQL), яка є реалізацією SQL-92 (стандарт ISO для SQL) з багатьма розширеннями. T-SQL дозволяє використовувати додатковий синтаксис процедур, що зберігаються і забезпечує підтримку транзакцій (взаємодія бази даних з керуючим застосунком). Microsoft SQL Server та Sybase ASE для взаємодії з мережею використовують протокол рівня застосунка під назвою Tabular Data Stream[en] (TDS, протокол передачі табличних даних).

Microsoft SQL Server також підтримує Open Database Connectivity (ODBC)— інтерфейс взаємодії застосунків з СКБД. Версія SQL Server 2005 надає можливість підключення користувачів через вебсервер-сервіси, що використовують протокол SOAP. Це дозволяє клієнтським програмам, не призначеним для Windows, кроссплатформенно з'єднуватися з SQL Server. Microsoft також випустила сертифікований драйвер JDBC, що дозволяє застосункам під керування Java (таким як BEA і IBM Websphere) з'єднуватися з Microsoft SQL Server 2000 і 2005.

SQL Server підтримує дзеркалювання та кластеризацію баз даних. Кластер серверу SQL—це сукупність однаково конфігурованих серверів; така схема допомагає розподілити робоче навантаження між декількома серверами. Усі сервери мають одне віртуальне ім'я, а дані розподіляються за IP-адресами машин кластеру протягом робочого циклу. Також у разі відмови або збою на одному з серверів кластеру доступне автоматичне перенесення навантаження на інший сервер. SQL Server підтримує надлишкове дублювання даних за трьома сценаріями:

- Знімок. Виконується «знімок» бази даних, який сервер відправляє одержувачам;
- Історія змін. Всі зміни бази даних безперервно передаються користувачам;
- Синхронізація з іншими серверами. Бази даних декількох серверів синхронізуються між собою. Зміни усіх баз даних відбуваються незалежно на кожному сервері, а під час синхронізації відбувається звірка даних. Дублювання такого типу передбачає можливість вирішення протиріч між базами даних.

SQL Server 2005 має вбудовану підтримку .NET Framework. Завдяки цьому, процедури бази даних, що зберігаються, можуть бути написані на будь-якій мові платформи .NET з використанням повного набору бібліотек, доступних для .NET Framework. На відміну від інших процесів, .NET Framework виділяє додаткову пам'ять і будує засоби керування SQL Server, не використовуючи вбудовані засоби Windows. Це підвищує продуктивність порівняно із загальними алгоритмами Windows, оскільки алгоритми розподілу ресурсів спеціально налагоджені для використання у структурах SQL Server.

2.4 Перенесення даних між СКБД

1. Перший етап міграції даних, проводимо інвентаризацію та робимо бекапи. Спочатку потрібно зрозуміти, які дані є в компанії і де вони зберігаються. У найпростішому випадку є одна база даних і в ній лежать всі дані. У великих проектах ситуація з даними складніша – з'являються кілька різних СКБД, key-value сховища, сервери для пошуку, файлові сховища, документні сховища, сервери кешування тощо. Даних може бути багато, а способів їхнього зберігання дуже багато. Насамперед потрібно розібратися з тим, що і де зберігається. Найпростіше звести всю інформацію до списку:

- Які дані зберігаються у цій системі;
- Типи даних;
- Шляхи до файлів;
- Приблизний розмір даних, що зберігаються;
- Як часто оновлюються дані;
- Наскільки дані є критичними для вашої системи;
- Як часто робляться бекапи;
- Чи можете ви втратити ці дані.

Таке теж буває, наприклад, якщо дані в одній СКБД читаються на основі даних з другої СКБД. Тоді можна видалити перший набір даних і потім відновити з другого набору. Потім слід переконатися, що для всієї важливої інформації є бекапи. Якщо бекапи не створюються - впровадьте інструменти резервного копіювання для цих даних.

2. Другий етап, плануємо міграцію даних. Тепер слід спланувати фронт робіт. Для цього ранжуємо всі джерела інформації за важливістю - від найважливіших до менш важливих. Для кожного джерела даних необхідно підготувати інструменти міграції. Це може бути будь-що, починаючи від текстового файлу з командами для виконання на сервері і закінчуючи складними файлами для систем автоматизації розгортання. Інструментарій може бути

простим, може бути складним, але він повинен бути задокументований. Це потрібно, щоб у разі потреби міграцію та копіювання даних можна було повторити. Наприклад, якщо доведеться мігрувати дані повторно, або щось піде не так.

Якщо ви готуєтеся в ході міграції до зміни архітектури — не забудьте підготувати додаткові тести для систем, які переїжджають на нові технічні рішення. Якщо ви переносите файли з дисків серверів на зберігання в сховищі S3 - після міграції потрібно перевірити доступність і цілісність всіх записів в новій системі.

Якщо ви змінюєте тип СКБД, то протестуйте не лише коректність вставки та видалення записів у БД, але й правильність роботи коннектів у базу під навантаженням, коли у вас запущено кілька копій програм. Звичайно, у вас повинні бути тести на основні компоненти програми, але після зміни архітектури потрібно деякі додаткові перевірки даних.

3. Третій етап, робимо тестову міграцію даних. Тестова міграція потрібна, щоб перевірити, як відбувається перенесення даних від одного провайдера до іншого. Вам потрібно взяти тестові сервери та виконати міграцію з ними. Мета експерименту – переконатися, що інструменти коректно працюють і дані після міграції доступні: бази даних працюють, файли відкриваються, записи вантажаться.

Переміщення файлів здається простою операцією, але іноді в ході копіювання можуть статися неприємні сюрпризи - файли ушкоджуються, версії баз даних не співпадають, десь не завантажується. Завжди потрібно перевіряти якість роботи ваших скриптів та налаштувань.

У процесі тестової міграції часто переносять усі дані, тобто імітують бойову міграцію. Однак якщо даних багато — не обов'язково мігрувати їх усі, це буде надто довго і трудомістко, достатньо перевірити працездатність на обмеженій вибірці. Скажімо, на 1% загальної кількості записів.

4. Четвертий етап, мігруємо дані. Після всіх попередніх перевірок та підготовки можна починати справжнє перенесення:

- Не забудьте повідомити клієнтів та користувачів про технічні роботи;
- Намагайтеся вимкнути всі сервери та сервіси, які можна, тому що в процесі міграції в них не повинні з'являтися нові дані. Ті системи, які не можна вимкнути, мігрують у гарячому режимі. Наприклад, через реплікацію, підтримуючи одночасно по одній копії кожного сервісу у старому та новому дата-центрі. Для гарячої міграції вже на цьому етапі необхідно підняти дві копії сервісів;
- Зробіть актуальні бекапи всієї інформації відразу після вимкнення, перед початком перенесення;
- Рухайте строго за планом, який склали заздалегідь. Використовуйте лише інструменти міграції, які вибрали та випробували, не відступайте від сценарію;
- Після перенесення переналаштуйте всі системи на роботу з новими даними, запустіть та протестуйте всі програми;
- Після цього можна знову запустити сервіси та зробити їх доступними для користувачів, але продовжуйте дивитися свої системи моніторингу – щось все одно могло піти не так, треба знати.

Висновки до розділу 2

Файлова СКБД Paradox сильно застаріла по більшості критеріїв, тому її складно використовувати як актуальну СКБД для збереження даних. Проте є багато альтернатив, які активно оновлюються, мають зручний інтерфейс, актуальні мови програмування та використовуються з більшістю програм. Для даної роботи було використано новітню СКБД Microsoft SQL Server, яка має всі необхідні характеристики для поставленої задачі.

Перенесення даних між СКБД досить складний та відповідальний процес, як і вельми ризикований. СКБД може перестати відповідати потребам клієнта або клієнт просто захоче оновитись, і потрібно зробити так щоб зберегти якомога більше даних.

3 РОЗРОБКА ЗАСТОСУНКУ ДЛЯ ПЕРЕНЕСЕННЯ ДАНИХ

3.1 Підготовка до роботи

Задача даного застосунку – найбільш зручне, надійне та швидке перенесення даних з існуючої бази даних СКБД Paradox на заготовлену пусту базу СКБД Microsoft SQL Server. Для оцінки роботи застосунку було обрано такі критерії:

- перенесення даних не має займати багато часу;
- якнайбільше даних має бути збережено
- таблиці мають бути перенесені у такому ж вигляді як і оригінальні
- мають бути збережені відношення
- інтерфейс програми має бути простим

За основну мову застосунку було взято англійську, проте планується декілька локалізацій, такі як українська і російська для зручності українського замовника.

3.2 Опис програмної реалізації застосунка

Проект було розроблено за допомогою Visual Studio 2022, нової версії продукту Visual Studio компанії Microsoft. Такий вибір був зроблений через зручність роботи з .NET у цьому середовищі розробки.

Сама програма представляє з себе .exe-файл який витягує таблиці із старої бази, тимчасово зберігає їх якщо користувач цього бажає і завантажує у заготовлену пусту базу на MS SQL Server.

Для її створення використовувалась бібліотека «Paradox database native .NET reader» що дозволяє легко витягувати дані з баз на основі Paradox. Її особливість у тому що вона дозволяє напряму читати дані з Paradox не використовуючи BDE. Також вона дозволяє використовувати основний індекс щоб знайти потрібні записи.

Paradox розбиває дані на файли за допомогою об’єктів бази даних – кожна таблиця має власний файл і кожен індекс також. Файли даних та індекси мають дуже схожу структуру, тому їх можна обробляти в одному класі – ParadoxFile. Ось основні класи (Рис. 3.1):

- ParadoxFile - базовий клас, що працює з поширеними структурами з файлів даних і індексів Paradox;
- ParadoxFile.DataBlock - представляє один блок даних;
- ParadoxFile.FieldInfo - тип даних поля;
- ParadoxFile.V4Hdr - структура, яка присутня лише в певних файлах/версіях Paradox;
- ParadoxTable - представляє файл даних таблиці;
- ParadoxPrimaryKey - представляє індекс таблиці;
- ParadoxFileType - перерахування з усіма типами файлів;
- ParadoxFieldType - перерахування з усіма типами даних;
- ParadoxRecord - являє собою запис даних;
- ParadoxDataReader - стандартна реалізація IDataReader;
- ParadoxCondition - базовий клас для умов, які можна використовувати для пошуку в даних індексу;
- Вкладені класи ParadoxCondition - різні реалізації умов;
- ParadoxCompareOperator - перерахування з підтримуваними операторами порівняння (==, !=, <, <=, >, >=).

По суті, файл індексу є просто таблицею з індексами блоків даних і пов’язаними індексованими значеннями. Це можна прочитати безпосередньо, або використати механізм, що переглядає дерево індексів і вибирає потрібний запис.

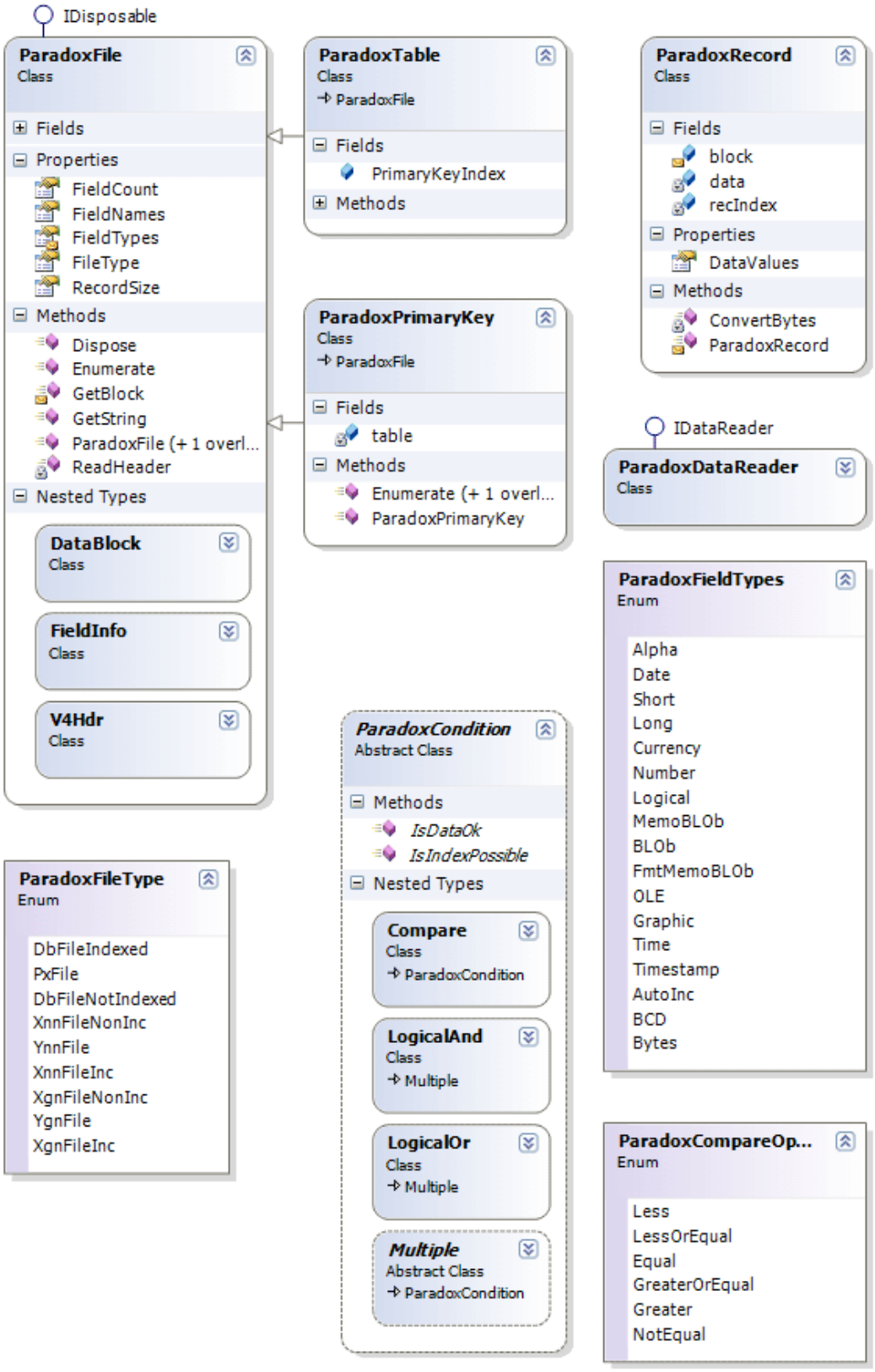

Рисунок 3.1 – Класи бібліотеки «Paradox database native .NET reader»

Для розуміння програми буде представлено декілька основних елементів коду із поясненнями.

По-перше, функція CopyTableJet виконує фактичну роботу з копіювання даних. Вона вставить 1000 записів одночасно, якщо вибрати "SQL Ser 2008+", лістинг коду знаходиться у Додатку А.

По-друге, функція GetCreateTableSqlFromParadox створить таблицю на сервері SQL, якщо вона не існує, лістинг коду знаходиться у Додатку Б.

По-третє, файли Paradox DB можуть бути захищені паролем. Програма використовує для пароля властивість рядка з'єднання JET OLEDB «Jet OLEDB: Database Password», лістинг коду знаходиться у Додатку В.

3.3 Користувацький інтерфейс

Для початку, розглянемо як виглядає наша база даних на Paradox (Рис. 3.2):

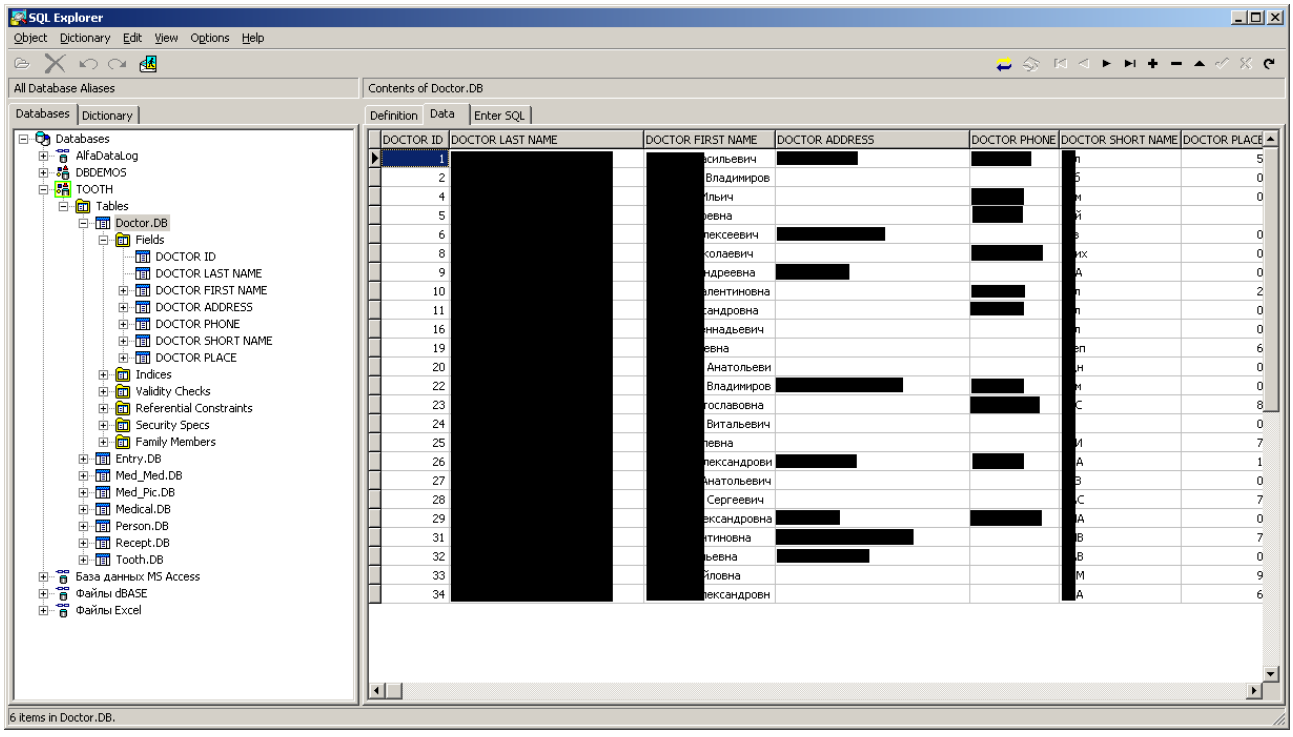

Рисунок 3.2 – Таблиця «Doctor»

Для наглядності та більшої швидкості відпрацювання було обрано невеличку, проте дуже важливу таблицю «Doctor». Для порівняння, у таблиці «Med_Med» більше ніж 490 тисяч записів.

Основним полем для роботи і, по суті, більшою частиною користувацького інтерфейсу постає вікно в якому будуть відображатись наявні таблиці (Рис. 3.3) та деяка інформація про них.

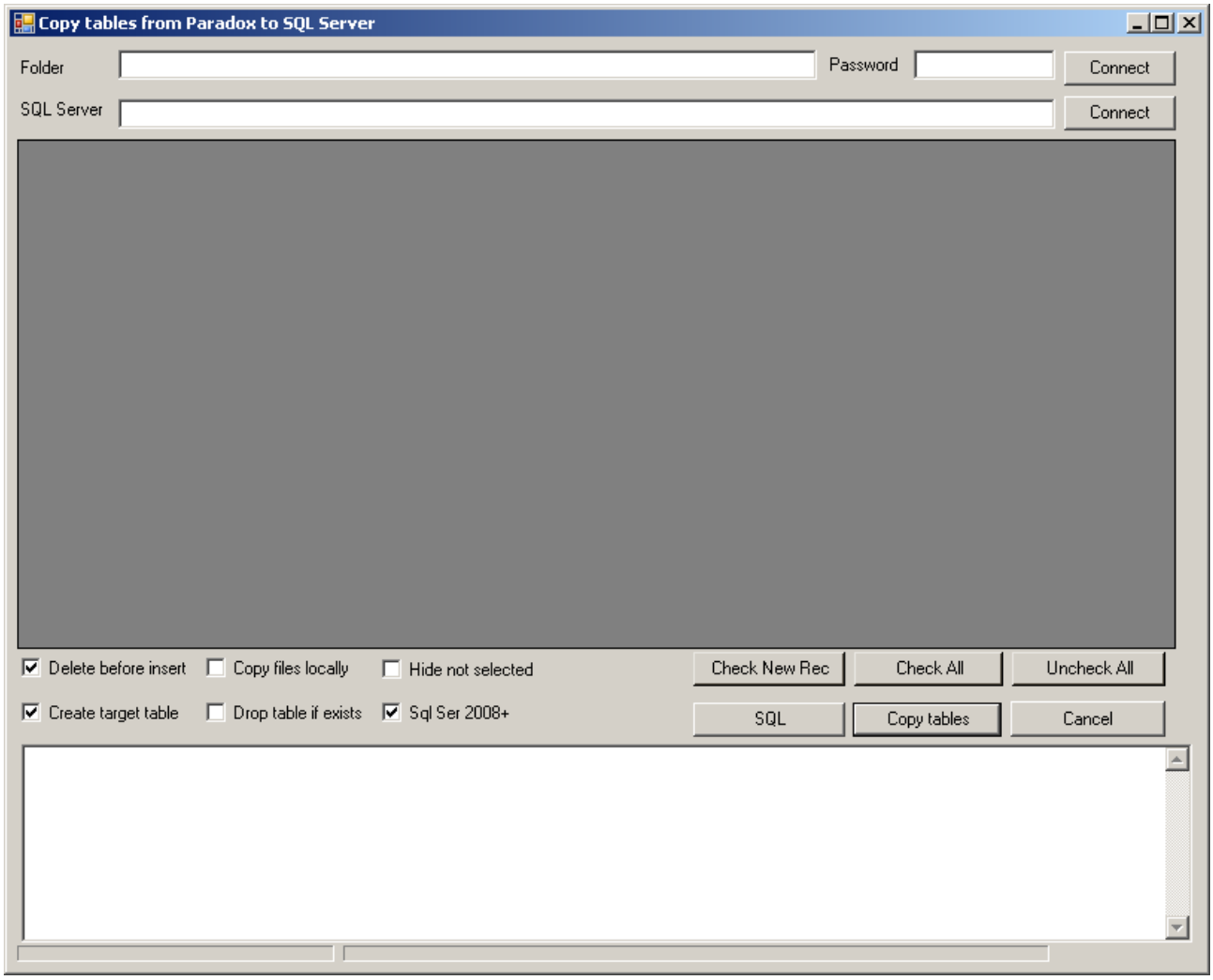

Рисунок 3.3 – Основне вікно роботи з програмою

Три обов’язкові поля згори це поля вибору бази з якої буде братись інформація, пароль до неї, а також поле для connection string кінцевої бази. Також там є кнопки «Connect», про які трохи пізніше. Далі іде поле для відображення таблиць, нижче – чекбокси для додаткових налаштувань. Це, наприклад, вже

згадуваний чекбокс «Sql Ser 2008+», або чекбокс який дозволяє зберегти файли перед завантаженням їх у нову базу. Правіше можна побачити набір кнопок, що відповідають за різні функції, такі як основна «Copy tables» що запускає в роботу алгоритм, «Check All» та «Uncheck All» які дозволяють швидко обирати які таблиці будуть перенесені.

Щоб обрати необхідну папку з базою даних треба натиснути кнопку «Connect» біля верхнього поля «Password» і обрати необхідне у випадаючому вікні. Після цього шлях до папки з'явиться у полі «Folder» (Рис. 3.4). Його також можна вказати вручну. Також одразу введіть пароль, якщо необхідно.

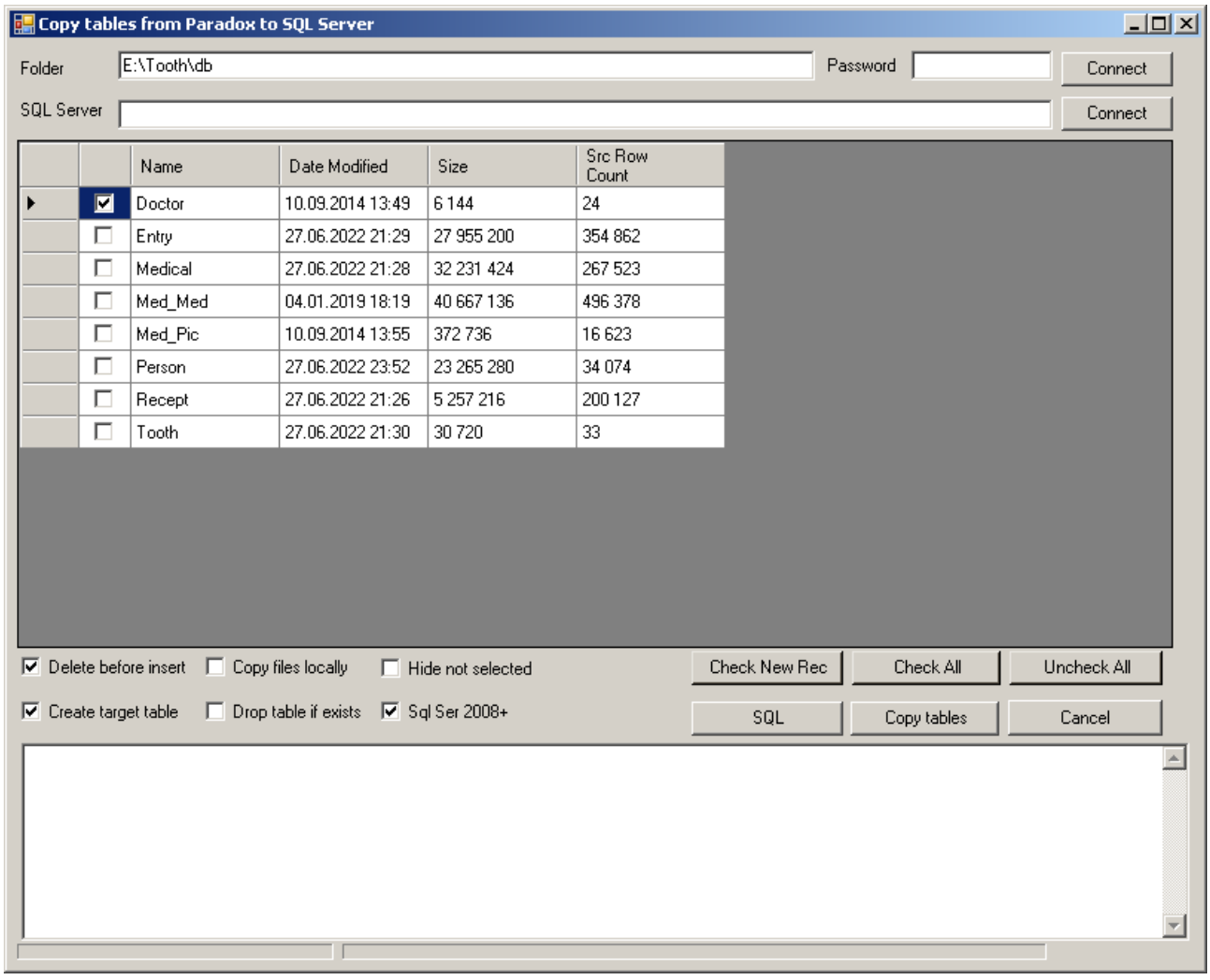

Рисунок 3.4 – Доступні таблиці після отримання доступу до бази Paradox

У полі для таблиць одразу з’явиться список доступних для перенесення таблиць, які можна обирати у будь якому порядку, а також додаткова інформація про кожну у вигляді зручної таблиці.

Наступний крок – отримання доступу до бази даних на SQL Server. Для цього нажимаємо відповідну кнопку «Connect». Після цього відкриється вікно підключення до БД, проте спочатку, на вкладці «Поставщик данных» (Рис. 3.5) потрібно впевнитись що обрано необхідне.

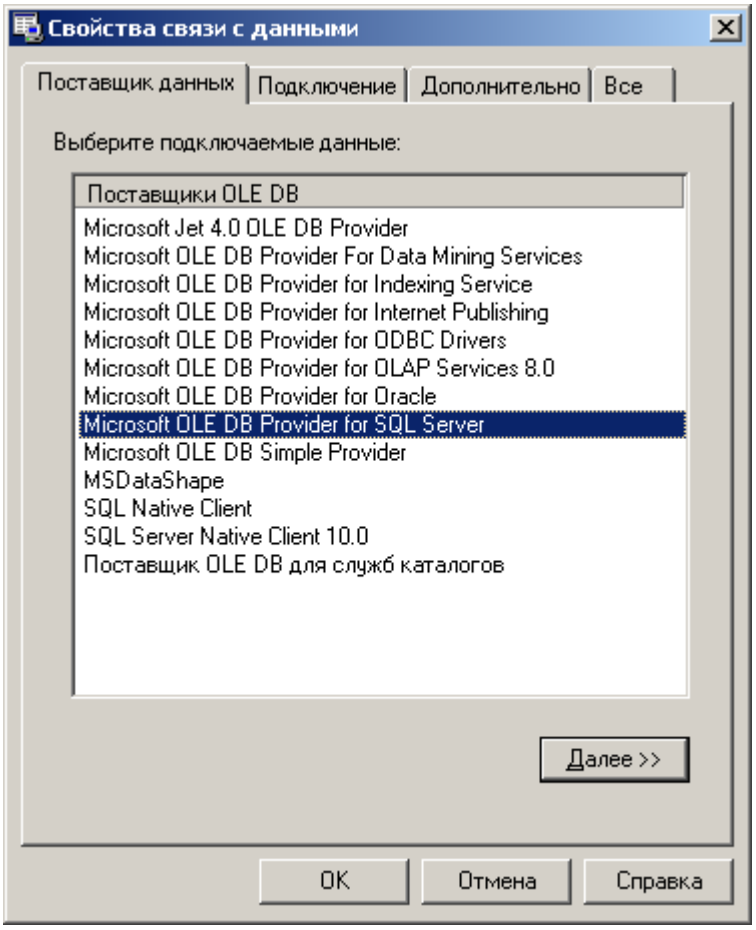

Рисунок 3.5 – Вкладка «Поставщик данных»

Далі потрібно повернутись на вкладку «Подключение» (Рис. 3.6) де необхідно вибрати ім’я сервера, спосіб логіну та, власне, цільову базу даних.

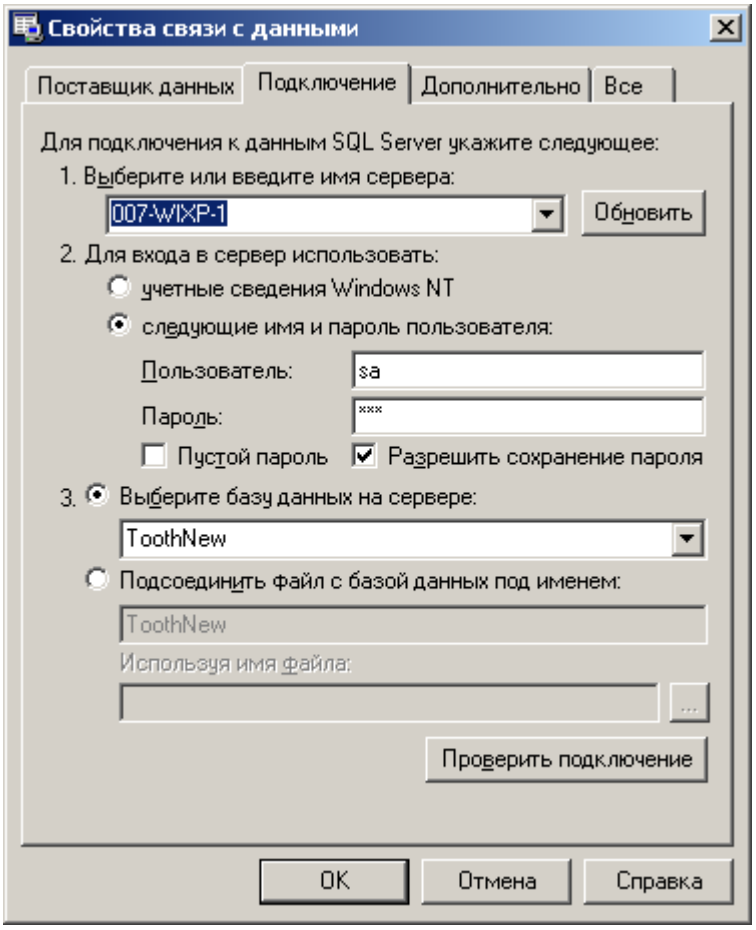

Рисунок 3.6 – Вкладка «Подключение»

Важливо скористатись кнопкою «Проверить подключение» аби впевнитись що конект пройшов успішно. Далі нажимаємо «ОК» і, якщо все правильно, у формі «SQL Server» з’явиться сформований connection string для доступу до нової бази. Також, з’явиться додатковий стовпець (Рис. 3.7) який підраховує кількість необхідних записів. На жаль, на даному етапі розробки, через особливості бази Paradox деякі записи перенести все ще неможливо.

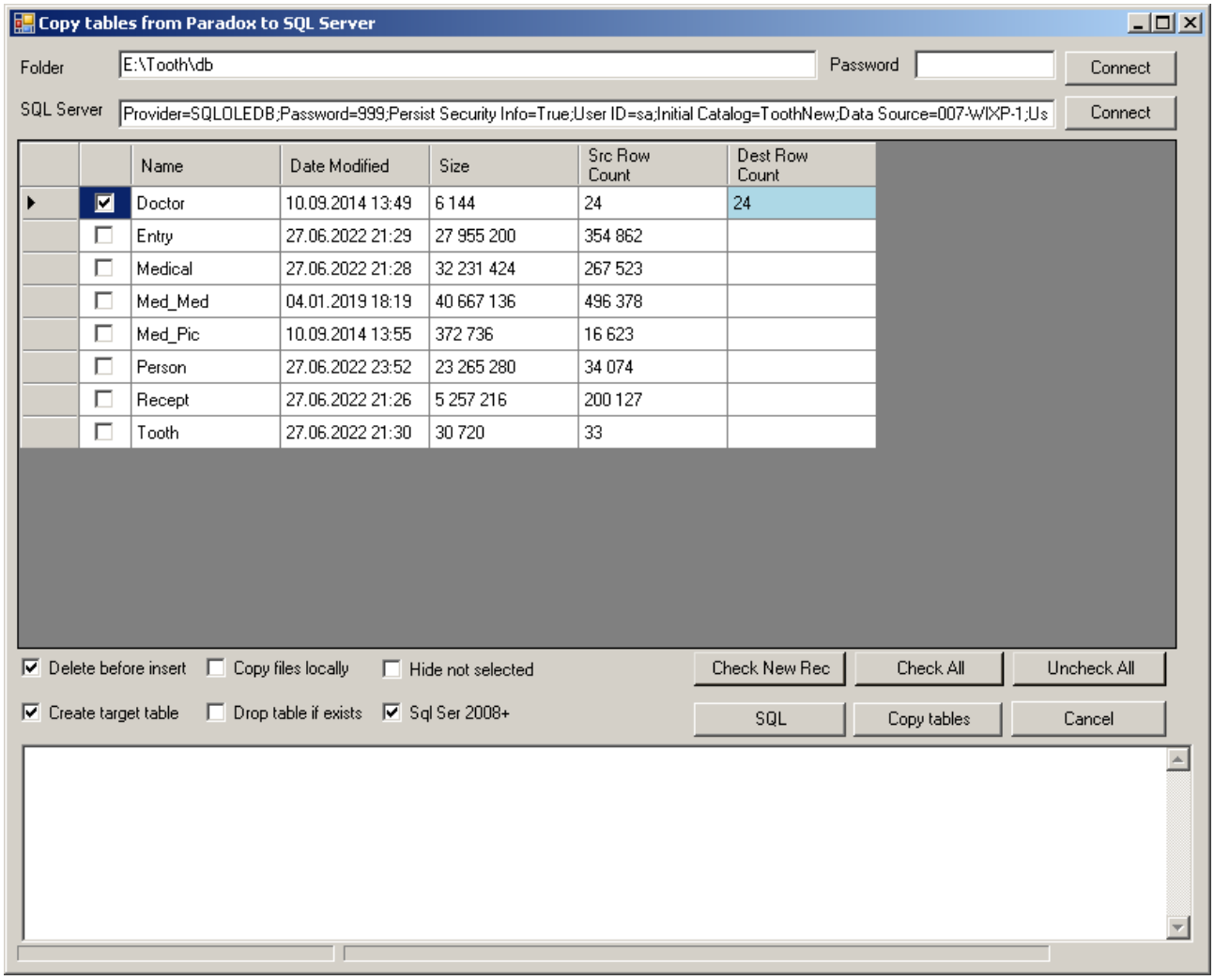

Рисунок 3.7 – Додатковий стовпець підрахунку записів

Обираємо необхідні чекбокси і нажимаємо кнопку «Copy tables». Отримуємо швидкий результат (Рис. 3.8). Подробиці виконаної операції та можливі помилки буде відображено у спеціальному булому полі внизу.

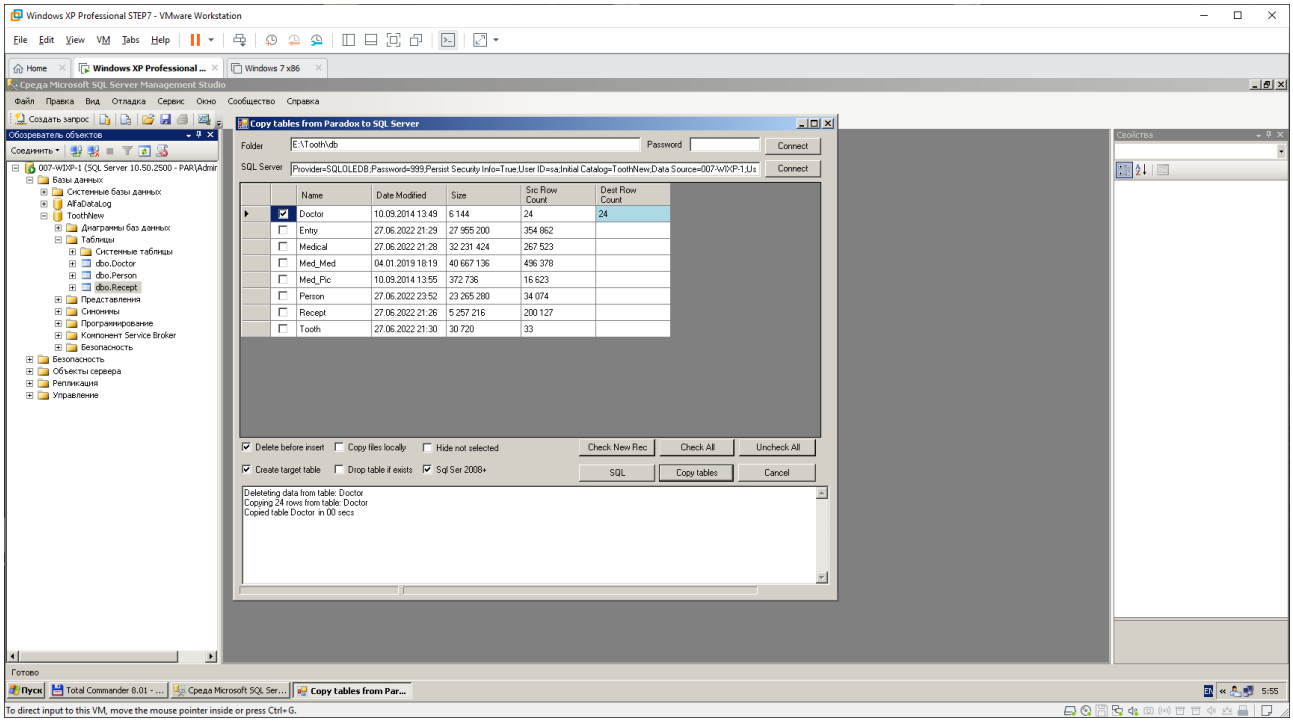

Рисунок 3.8 – Результат роботи програми

В кінці роботи можна впевнитись що задачу виконано успішно переглянувши цільову таблицю за допомогою MS SQL Server Management Studio (Рис. 3.9).

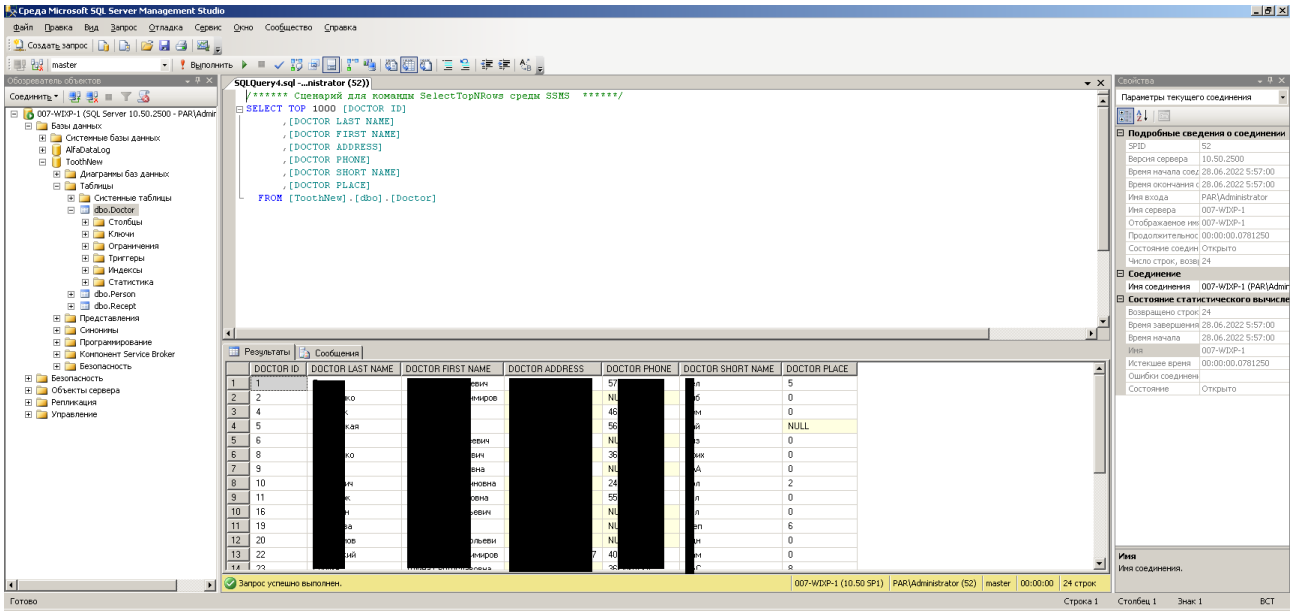

Рисунок 3.9 – Створена таблиця-результат «Doctor»

Як можна помітити, усі записи які вдалось перевести знаходяться у відповідній базі та таблиці. Поля збережені правильно, що можна побачити при порівнянні з початковими полями.

Також, якщо ви обирали чекбокс «», то можна впевнитись що проміжні файли було створено просто переходячи до вказаної у результаті директорії (Рис. 3.10).

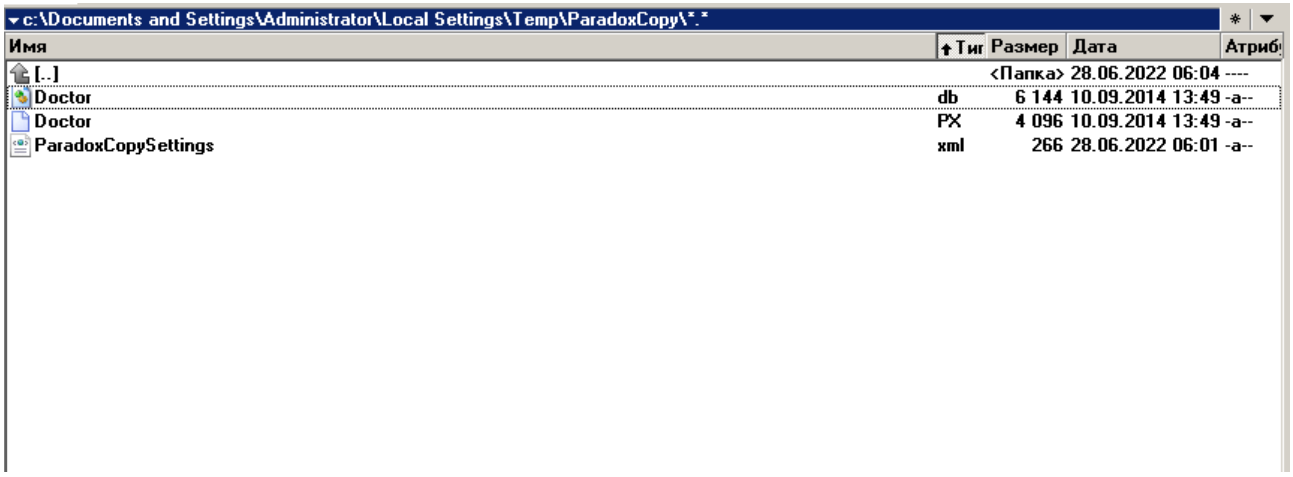

Рисунок 3.10 – Створені проміжні файли у спеціальній директорії

Висновки до розділу 3

Отже, у даному розділі було розглянуто основні вимоги до створення застосунку перенесення даних з бази Paradox на базу MS SQL Server. Було описано основні моменти програмної реалізації, а також продемонстровано основні функції готового застосунку на прикладі таблиці «Doctor».

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

Спеціальний розділ

ОХОРОНА ПРАЦІ

до кваліфікаційної роботи

на тему:

«ВИКОРИСТАННЯ ТЕХНОЛОГІЇ КЛІЄНТ-СЕРВЕР ДЛЯ МОДЕРНІЗАЦІЇ ФАЙЛОВОЇ БАЗИ ДАНИХ PARADOX СТОМАТОЛОГІЧНОЇ КЛІНІКИ»

Спеціальність 122 «Комп'ютерні науки»

122 – БКР – 402.21810220

Виконала студентка 4-го курсу, групи 402
_____*А. О. Паплаускайте*

«___» _____ 2022 р.

Консультант к.т.н., доцент
_____*А. О. Алексєєва*

«___» _____ 2022 р.

Миколаїв – 2022

ЗМІСТ

4. ОХОРОНА ПРАЦІ 49

ВСТУП 49

4.1 Аналіз виробничого травматизму 50

4.2. Освітлення робочого місця та робочого приміщення 52

4.3 Вимоги до організації приміщення 53

4.4 Протипожежний захист 56

4.5. Техніка безпеки при роботі з ПК 57

ВИСНОВКИ ДО РОЗДІЛУ 4 70

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ВДТ	– Візуальний дисплейний термінал
ДСанПіН	– Державні санітарні норми і правила
ДСН	– Державні санітарні норми
ЕОМ	– Електронна обчислювальна машина
КПО	– Коефіцієнт природної освітленості
ПК	– Персональний комп’ютер
ПЕОМ	– Персональні електронні обчислювальні машини
ССБП	– Система стандартів безпеки праці

4 ОХОРОНА ПРАЦІ

Охорона праці при користуванні екранними пристроями

ВСТУП

В сучасній Україні та світові технології постійно розвиваються і тому автоматизація та оптимізація виробничих процесів просто необхідні. Сьогодні, напевно, важко уявити компанію, господарська діяльність в якій здійснювалась би без використання комп’ютерної техніки. Через те що масовість використання робітниками комп’ютерів у виробництві досягла дуже великих масштабів, законодавством України чітко врегульовано норми та вимоги до використання комп’ютерної техніки на підприємстві.

Використання комп’ютера може викликати великі проблеми зі здоров’ям. Захворювання очей та хребта, болі в кистях і мозолі, проблеми з ногами – це все можливі наслідки. Також від комп'ютера надходить шкідливе електромагнітне випромінювання.

Через тривале перебування у вимушеній статичній позі виникають порушення опорно-рухового апарату: зміни хребетного стовпа – порушення постави, сколіоз; ураження променево-зап’ясткового суглоба – тунельний синдром, артрози. Розвиваються порушення зору та стану ока: зоровий комп’ютерний синдром., оскільки око еволюційно не пристосоване для постійного розглядання близько розміщених предметів. Спосіб життя перетворюється на малорухливий, виникає гіподинамія. Природній стан тіла людини – рух, який забезпечує рівномірний розподіл навантаження на суглобові поверхні, нормальний режим роботи – зміна напруги і розслаблення – поперечно смугастих м’язів та активний кровообіг, що забезпечує живлення всім органам.

У зв’язку з цим виникає необхідність розгляду питання охорони праці на підприємствах з електронними пристроями та комп’ютерами.

4.1 Аналіз виробничого травматизму

Науково-технічний процес призводить до корінної зміни характеру і засобів праці. Цей процес створює для людини велику кількість благ:

– зменшує фізичне навантаження;

– робить працю інтелектуальною, цікавою, різноманітною;

– розвиває творчі здібності людини.

Широке впровадження комп'ютерної техніки істотно підвищує продуктивність праці.

Збереження здоров'я користувачів, персональних ЕОМ, всіх працюючих, підтримання ефективності та надійності їх праці на належному рівні є одним із аспектів застосування дисципліни - охорона праці.

Охорона праці - це система законодавчих актів, соціально-економічних, організаційно-технічних, гігієнічних і лікувально-профілактичних заходів та засобів, що забезпечують безпеку, збереження здоров'я та працездатність людини у процесі праці.

Широкомасштабні заходи, спрямовані на поліпшення здоров'я людей, повинні здійснюватися на кожному підприємстві у встановленому законодавчому порядку. Суворе додержання умов гігієни та фізіології праці є не тільки особистою справою людини, але й колективу, оскільки порушення принципів гігієни позначається не тільки на здоров'ї порушника, але й інших членів колективу.

Комплекс організаційних заходів і технічних засобів захисту, накопичений досвід роботи показує, що є реальна можливість добитися успіхів щодо усунення впливу небезпечних і шкідливих факторів на працюючих з комп'ютерами. Проте, особи, допущені до роботи з комп'ютерною технікою, все ще зазнають впливу небезпечних і шкідливих виробничих факторів, таких як монотонність, підвищена температура зовнішнього середовища, відсутність або недостатня освітленість робочої зони, електричний струм, статична електрика та інших.

Праця багатьох працівників пов'язана з впливом психо-фізичних факторів, таких як розумове перевантаження, перенапруження зорових і слухових аналізаторів, емоційні перевантаження.

Нещасні випадки під час роботи за комп'ютерами можуть статися внаслідок дій фізично-небезпечних виробничих факторів при підключенні периферійних пристроїв зі знятим кожухом корпусу системного блока, при ураженні електричним струмом, виконанні працівниками невластивих для них робіт. Електричні установки становлять потенційну небезпеку для людини як у процесі експлуатації, так і під час проведення профілактичних робіт.

4.2. Освітлення робочого місця та робочого приміщення

Робоче місце - це обладнаний технічними засобами (засобами відображення інформації, органами управління, допоміжним обладнанням) простір, де здійснюється діяльність виконавця (або групи виконавців). Вимоги до освітлення для візуального сприймання користувачами інформації з двох різних носіїв (з екрана ПК та паперового носія) різні. Надто низький рівень освітленості погіршує сприймання інформації при читанні документів, а надто високий призводить до зменшення контрасту зображення знаків на екрані. При 10 % зменшенні освітленості працездатність знижується на 1 %. Освітленість можна варіювати від 300 до 700 лк. Оптимальною освітленістю робочих приміщень для роботи з відеотерміналом є освітленість від 300 до 500 лк.

Освітлення робочого місця повинно бути змішаним (природним та штучним). Доцільно, щоб орієнтація світлових отворів для приміщення з ВДТ була на північ. Природне освітлення повинно здійснюватись у вигляді бічного освітлення.

Слід передбачити наявність сонцезахисних засобів, що знижують перепади яскравостей між природним світлом та свіченням екрана ЕОМ. Необхідно використовувати плівки з металізованим покриттям або жалюзі з вертикальними ламелями, що регулюються. Світлорозсіюючі штори повинні мати коефіцієнт відбивання 0,5-0,7.

Робоче місце оператора ЕОМ повинно бути розташоване так, щоб в поле зору не потрапляли вікна або освітлювальні прилади. Коли штучне освітлення змішується з природним, рекомендується використовувати лампи за спектральним складом, які є найбільш близькі до сонячного світла.

Штучне освітлення у приміщеннях з ЕОМ здійснюється у вигляді комбінованої системи з використанням люмінісцентних джерел світла у світильниках загального освітлення, які розташовують над робочими поверхнями у рівномірно-прямокутному порядку. Пульсація люмінісцентних

ламп не повинна перевищувати 10 %. Вікна бажано розміщувати з одного боку робочих приміщень.

Відношення яскравості екрану комп'ютера до яскравості оточуючих його поверхонь не повинно перевищувати у робочій зоні 3:1.

На робочому місці має бути забезпечена рівномірна освітленість за допомогою переважно відбитого або розсіяного світлорозподілу світлових відблисків з клавіатури, екрана та від інших частин відеотермінала (ВДТ) з обох його сторін.

4.3 Вимоги до організації приміщення

Розширене використання комп'ютерної технології повинно супроводжуватись прагненням поліпшити обстановку на робочому місці, яка сприятиме збереженню високої працездатності та створюватиме благотворні умови для співпраці користувачів ПЕОМ.

Приміщення повинні мати змішане освітлення. Доцільно обирати орієнтацію вікон на північ або північний схід. На вікнах повинні бути жалюзі, що регулюються, або штори, що дають можливість їх повністю закривати.

Робочі місця з ПЕОМ повинні, як правило, розміщуватись в окремих приміщеннях. У випадку розміщення робочих місць у спеціальних залах або приміщеннях з джерелами небезпечних (шкідливих) виробничих факторів, вони повинні розташовуватись у повністю ізольованих кабінетах з природним освітленням та організованим повітрообміном. Площа, на якій розташоване одне робоче місце з ВДТ, повинна становити не менше як 6,0 м2, об'єм приміщення - не менше як 20 м3.

Для оздоблення приміщень з ВДТ повинні використовуватись дифузно-віддзеркалюючі матеріали з коефіцієнтами відбиття:

– стелі - 0,7-0,8;
– стін - 0,4-0,5;

– підлоги - 0,2-0,3.

Поверхня підлоги має бути рівною, неслизькою, зручною для очищення та вологого прибирання, мати антистатичні властивості.

Забороняється застосовувати для оздоблення інтер'єру полімерні матеріали, що виділяють у повітря шкідливі хімічні речовини. Вміст шкідливих хімічних речовин у приміщеннях з ВДТ не повинен перевищувати концентрацій, вказаних у ГОСТ 12.1.005-88 ”Общие санитарно-гигиенические требования к воздуху рабочей зоны”.

Висока температура повітря негативно позначається на функціональному стані людини.

Оптимальні та припустимі мікрокліматичні параметри у приміщеннях повинні враховувати специфіку технологічного процесу при використанні ПК. Зокрема, технічні умови експлуатації багатьох типів комп'ютерів містять допустимі робочі діапазони параметрів мікроклімату:

– температура повітря має знаходитись в межах від 10 до 40°С;

– відносна вологість має знаходитися в межах від 40 до 90 %.

За даними ВООЗ, оптимальні значення температури у приміщенні становлять 19-23 °С, відносна вологість повітря - 55 %, швидкість руху повітря не повинна перевищувати на рівні обличчя 0,1 м/с. При відчутному нагрівання поверхонь (більше 45°С), контрактуючих з людиною, передбачаються засоби охолодження або ізоляції. Особлива увага приділяється шляхом відводу повітря, щоб виключити перегрівання або протяг.

Згідно з діючими в нашій країні нормативними документами (ВСНиПРВЦ, СН 4088-86 ”Микроклимат производственных помещений” ГОСТ 12.1.005-88) у холодні періоди року:

– температура повітря повинна складати 22-24°С;

– швидкість його руху - 0,1 м/с;

– відносна вологість повітря - 40-60 %.

В теплі пори року:

– температура повітря дорівнює 23-25 °С;

– швидкість його руху - 0,1-0,2 м/с;

– вологість - 40-60 %.

Температура повітря може коливатися від 22 до 26 °С при збереженні інших параметрів мікроклімату у вказаних вище межах.

Необхідно надати можливість індивідуального регулювання роздачі повітря в приміщеннях шляхом встановлення кондиціонерів.

В процесі роботи ВДТ змінюється концентація іонів у повітрі робочої зони. Нормалізуючий вплив на аероіонний склад повітря робочої зони справлють примусова вентиляція, захисні екрани та застосування іонізаторів.

Одним з основних забруднювачів повітря у приміщеннях, зокрема оксидами вуглецю, є тютюновий дим.

Слід зазначити, що у випадках, коли робота користувачів пов'язана з прийняттям відвідувачів у приміщеннях з ВДТ, вміст мікроорганізмів у повітрі приміщень значно зростає і досягає, залежно від числа і тривалості перебування відвідувачів, 7000 та більше мікробних тіл на 1 м3. Тому при такому режимі роботи необхідно залучати додаткові заходи оздоровлення повітряного середовища (застосовувати додаткову вентиляцію, обмежувати число та тривалість перебування відвідувачів, конструювати робочі місця).

Згідно з діючими нормативними документами повітря, що надходить у робочі приміщення має бути очищене від забруднень, в тому числі від мікроорганізмів (ВСНиПРВЦ). Запиленість повітря не повинна перевищувати вимог, викладених у СН 512-78.

4.4 Протипожежний захист

Пожежі в приміщеннях з оргтехнікою становлять особливу небезпеку, бо поєднані з великими матеріальними збитками. Пожежа може виникнути при взаємодії горючих речовин і джерел запалювання. Горючими речовинами є будівельні та опоряджувальні матеріали, пластмасові корпуси техніки, шнури тощо. Джерелами запалювання можуть бути електронні схеми комп'ютерів, принтерів, пристроїв елекроживлення, де внаслідок різних порушень виникає перегрівання елементів, утворюються електричні іскри та дуги, здатні спричинити займання горючих матеріалів.

При обслуговуванні, ремонтних та профілактичних роботах використовуються різні лейкозаймисті рідини, прокладаються тимчасові елекропровідники, здійснюється паяння. Виникає додаткова пожежна небезпека, яка потребує відповідних заходів пожежного захисту. До засобів гасіння пожежі, призначених для локалізації невеликих займань, належать вогнегасники, сухий пісок, азбестові ковдри. Приміщення, в який встановлено комп'ютери і де немає необхідності влаштування систем автоматичного пожежогасіння, необхідно оснащувати переносними вуглекислотними з розрахунку 2 шт. на кожні 20 м2 в приміщеннях. Звуковбирне облицювання стін, стель приміщень треба виконувати з незгорючих та важкогорючих матеріалів.

З метою виявлення початкової стадії займання необхідно використовувати пристрої систем атоматичного пожежогасіння там, де цього вимагають Правила пожежної безпеки.

4.5. Техніка безпеки при роботі з ПК

При виконанні робіт на комп'ютерах необхідно дотримуватись вимог загальної та даної інструкції з охорони праці.

До самостійної роботи на комп'ютерах допускаються особи, які пройшли медичний огляд, навчання по професії, вступний інструктаж з охорони праці та первинний інструктаж з охорони праці на робочому місці. В подальшому вони проходять повторні інструктажі з охорони праці на робочому місці один раз на півріччя, періодичні медичні огляди один раз на два роки.

Під час роботи на комп'ютерах можуть діяти такі небезпечні та шкідливі фактори, як:

– фізичні;

– психофізіологічні.

Основним обладнанням робочого місця користувача комп'ютера є монітор, системний блок та клавіатура.

Робочі місця мають бути розташовані на відстані не менше 1,5 м від стіни з вікнами, від інших стін на відстані 1м, між собою на відстані не менше 1,5 м. Відносно вікон робоче місце доцільно розташовувати таким чином, щоб природне світло падало на нього збоку, переважно зліва.

Робочі місця слід рошташовувати так, щоб уникнути попадання в очі прямого світла. Джерела освітлення рекомендується розташовувати з обох боків екрану паралельно напрямку погляду. Для уникнення світлових відблисків екрану, клавіатури в напрямку очей користувача, від світильників загального освітлення або сонячних променів, необхідно використовувати антиполискові сітки, спеціальні фільтри для екранів, захисні козирки, на вікнах - жалюзі.

Фільтри з металевої або нейлонової сітки використовувати не рекомендується, тому що сітка спотворює зображення через інтерференцію світла. Найкращу якість зображення забезпечують скляні поляризаційні фільтри.

Вони усувають практично всі відблиски, роблять зображення чітким і контрастним.

При роботі з текстовою інформацією (в режимі введення даних та редагування тексту, читання з екрану) найбільш фізіологічним правильним є зображення чорних знаків на світлому (чорному) фоні.

Монітор повинен бути розташований на робочому місці так, щоб поверхня екрана знаходилася в центрі поля зору на відстані 400-700 мм від очей користувача. Рекомендується розміщувати елементи робочого місця так, щоб витримувалася однакова відстань очей від екрана, клавіатури, тексту[27].

Зручна робоча поза при роботі з комп'ютером забезпечується регулюванням висоти робочого столу, крісла та підставки для ніг. Раціональною робочою позою може вважатися таке положення, при якому ступні працівника розташовані горизонтально на підлозі або підставці для ніг, стегна зорієнтовані у горизонтальній площині, верхні частини рук - вертикальні. Кут ліктєвого суглоба коливається в межах 70-90°, зап'ястя зігнуті під кутом не більше ніж 20°, нахил голови 15-20°.

Для нейтралізації зарядів статичної електрики в приміщенні, де виконується робота на комп'ютерах, в тому числі на лазерних та світлодіодних принтерах, рекомендується збільшувати вологість повітря за допомогою кімнатних зволожувачів. Не рекомендується носити одяг з синтетичних матеріалів.

Згідно статті 18 Закону України ”Про охорону праці” працівник зобов'язаний:

а) знати і виконувати вимоги нормативних актів про охорону праці, правила поводження з устаткуванням та іншими засобами виробництва, користуватися засобами колективного та індивідуального захисту;

б) дотримуватись зобов'язань щодо охорони праці, передбачених колективним договором та правилами внутрішнього трудового розпорядку підприємства;

в) співробітничати з власником у справі організації безпечних і нешкідливих умов праці, особисто вживати посильних заходів щодо усунення будь-якої виробничої ситуації, яка створює загрозу його життю чи здоров'ю, або людей, які його оточують, повідомляти про небезпеку свого безпосереднього керівника або іншу посадову особу.

Вимоги безпеки перед початком роботи:

– увімкнути систему кондиціювання в приміщенні;

– перевірити надійність встановлення апаратури на робочому столі;

– повернути монітор так, щоб було зручно дивитися на екран - під прямим кутом (а не збоку) і трохи зверху вниз, при цьому екран має бути трохи нахиленим, нижній його край ближче до оператора;

– перевірити загальний стан апаратури, перевірити справність електропроводки, з'єднувальних шнурів, штепсельних вилок, розеток, заземлення захисного екрана;

– відрегулювати освітленість робочого місця;

– відрегулювати та зафіксувати висоту крісла, зручний для користувача нахил його спинки;

– приєднати до системного блоку необхідну апаратуру. Усі кабелі, що з'єднують системний блок з іншими пристроями, слід вставляти та виймати при вимкненому комп'ютері;

– ввімкнути апаратуру комп'ютера вимикачами на корпусах в послідовності: монітор, системний блок, принтер (якщо передбачається друкування);

– відрегулювати яскравість свічення монітора, мінімальний розмір світної точки, фокусування, контрастність. Не слід робити зображення надто яскравим, щоб не втомлювати очей.

Рекомендується:

– яскравість свічення екрана - не менше 100Kg/M2;

– відношення яскравості монітора до яскравості оточуючих його поверхонь в робочій зоні - не більше 3:1;

– мінімальний розмір точки свічення не більше 0,4 мм для монохромного монітора і не менше 0,6 мм для кольорового, контрастність зображення знаку - не менше 0,8;

– при вивленні будь-яких несправностей роботу не розпочинати, повідомити про це керівника.

Вимоги безпеки під час виконання роботи:

– необхідно стійко розташовувати клавіатуру на робочому столі, не опускати її хитання. Під час роботи на клавіатурі сидіти прямо, не напружуватися;

– для забезпечення несприятливого впливу на користувача пристроїв типу ”миша” належить забезпечувати вільну велику поверхню столу для переміщення ”миші” і зручного упору ліктьового суглоба;

– не дозволяються сторонні розмови, подразнюючі шуми; періодично при вимкненому комп'ютері прибирати ледь змоченою мильним розчином бавовняною ганчіркою порох з поверхонь апаратури. Екран ВДТ та захисний екран протирають ганчіркою, змоченою у спирті. Не дозволяється використовувати рідинні або аерозольні засоби чищення поверхонь комп'ютера.

Забороняється:

– самостійно ремонтувати апаратуру;

– класти будь-яку предмети на апаратуру комп'ютера;

– закривати будь-чим вентиляційні отвори апаратури, що може призвести до її перегрівання і виходу з ладу.

Для зняття статичної електики рекомендується час від часу доторкатися до металевих поверхонь.

Розташувати принтер необхідно поруч з системним блоком таким чином, щоб з'єднювальний шнур не був натягнутий. Забороняється ставити принтери на системний блок.

Для досягнення найбільш чистих, з високою розподільністю зображень і щоб не зіпсувати апарат, має використовуватися папір, вказаний в інструкції до принтера. При заминанні паперу потрібно відкрити кришку і обережно витягнути лоток з папером.

Згідно з інструкцією фірми-виробника потрібно дотримуватися правил зберігання картриджа.

Забороняється:

– зберігати картриджі без упаковки;
– ставити картриджі вертикально;
– перевертати картридж етикеткою донизу;
– відкривати кришку валика і доторкатися до нього;
– самому заповнювати використаний картридж.

Вимоги безпеки після закінчення роботи:

– закінчити та записати у пам'ять комп'ютера файл, що знаходиться в роботі;
– вимкнути принтер та інші периферійні пристрої. Штепсельні вилки витягнути з розеток. Накрити клавіатуру кришкою запобігання попаданню в неї пилу;
– прибрати робоче місце;
– ретельно вимити руки теплою водою з милом;
– вимкнути кондиціонер, освітлення і загальне електроживлення;
– пройти в спеціально обладнаному приміщенні сеанс психофізіологічного розвантаження і зняття втоми з виконанням спеціальних вправ аутогенного тренування.

Для запобігання створенню значної напруженості та захисту від статистичної електрики у приміщеннях з ПЕОМ використовують нейтралізатори та

зволожувачі повітря. Всі ПЕОМ повинні бути заземлені (занулені).Підлога повинна мати антистатичне покриття. Захист від статичної електрикиповинен проводитись згідно з санітарно-гігієнічними нормами напруженості електричного поля, які є допустимими. Ці рівні не повинні перевищувати 20 Кв протягом години (ГОСТ 12.1045-84)[28].

Металеві неструмоведучі частини електрообладнання і електро-установок при порушенні ізоляції між ними і їхніми струмоведучими частинами можуть опинитися під напругою. У таких аварійних умовах дотик до неструмоведучих частин установок рівнозначний дотику до струмоведучих частин.

Усунення небезпеки ураження електричним струмом при такому переході напруги на неструмоведучі частини електроустановок здійснюється за допомогою захисного заземлення.

Під захисним заземленням розуміють з'єднання металічних неструмоведучих частин електроустановок із землею через заземлюючі провідники та заземлювачі для створення між цими частинами і землею малого опору.

Для захисного заземлення застосовують заземлювачі із труб. Це декілька металічних труб діаметром 25-50 мм, довжиною 2-3 м, які забивають в землю на відстані 4-6 м одна від одної і з'єднують між собою металічною трубою. Остання проходить в приміщення, з'єднується з внутрішнім контуром.

Розрахунок зводиться до визначення необхідної кількості труб заземлювача і довжини металічної труби, яка з'єднує вертикальні заземлювачі.

Державне регулювання має здійснюватися через законотворчість та нормотворчість. Нещодавно між Держпромгірнаглядом та Міжнародною Фінансовою Корпорацією Світового Банку було підписано меморандум про співпрацю. На зустрічі, що відбулася наприкінці березня, сторони домовилися постійно розвивати та удосконалювати співпрацю. Першочерговими завданнями на цей рік визначено коригування та інвентаризацію законодавчої бази з промислової безпеки та охорони праці[29]. Адже в Україні понад 2,5 тисячі

державних нормативних актів з охорони праці, більшість з яких застаріла та потребує перегляду. Українська сторона хотіла б отримати пропозиції та рекомендації незалежних закордонних експертів щодо удосконалення документів з точки зору адаптації їх до норм міжнародного та європейського законодавства. Окрім того, Держпромгірнагляд прийняв Концепцію удосконалення законодавчого та нормативно-правового забезпечення охорони праці в Україні та розробив проект довготривалої програми нормативно-правового забезпечення охорони праці та промислової безпеки на 2006 - 2010 роки.

ВИСНОВКИ ДО РОЗДІЛУ 4

Під час виконання спеціальної частини з охорони праці було досліджено умови праці людини на підприємствах та установах, а також роботу при користуванні електронними пристроями.

На сьогодні існує багато проблем з організацією праці для робітників за комп'ютерами. Багато з них страждають від наслідків та не можуть продовжувати роботу що погано як для них так і для бізнесу.

На підприємствах мають дотримуватись вимоги та нормативи щодо охорони праці під час трудової діяльності. Дотримання поставлених вимог до працівників та власників підприємств дозволить мінімізувати шкідливі наслідки, які мають вплив на здоров'я людини.

Правила, які прописані у нормативних документах є актуальними на допоки люди користуються комп'ютерами для роботи. Під час роботи за ПК людина швидко втомлюється та від цього страждає опорно-руховий, зоровий апарат, нервова система и в окремих випадках психічне здоров'я.

ВИСНОВКИ

У результаті виконання бакалаврської кваліфікаційної роботи було проведено аналіз предметної області модернізації файлової бази Paradox. Також було розглянуто аналоги вже існуючих додатків подібного призначення. Перед безпосереднім створенням застосунку було визначено інструментальні засоби розробки шляхом розглядання переваг та недоліків необхідних технологій.

Після цього на основі визначених вимог було створено застосунок за допомогою бібліотеки Paradox database native .NET reader, яка дозволяє легко витягувати дані з баз на основі Paradox, а також напряму читати дані з Paradox не використовуючи BDE. Функціональність системи дозволяє користувачу обирати необхідні для перенесення таблиці та слідкувати за даними, а також за можливими помилками у роботі.

Особливістю застосунку є його майбутня інтеграція у більший проект, простота у використанні та практичність.

СПИСОК ВИКОРИСТАННИХ ДЖЕРЕЛ

1. Try SQL Server on-premises or in the cloud: веб-сайт. URL: https://www.microsoft.com/en-us/sql-server/sql-server-downloads (дата звернення: 21.05.2022).

2. Beaulieu A. Learning SQL: Generate, Manipulate, and Retrieve Data / Alan Beaulieu; 3rd edit. – O'Reilly Media, 2020a. 482 с.

3. Petkovic D. Microsoft SQL Server 2019: A Beginner's Guide, Seventh Edition / Dusan Petkovic; 7th edit – McGraw, 2020. 400 с.

4. Carter P. Pro SQL Server 2019 Administration: A Guide for the Modern DBA / Peter Carter – Apress, 2019. 420 с.

5. Microsoft SQL Server: веб-сайт. URL: https://uk.wikipedia.org/wiki/Microsoft_SQL_Server (дата звернення: 26.05.2022).

6. Tanimura C. SQL for Data Analysis: Advanced Techniques for Transforming Data into Insights / Cathy Tanimura – O'Reilly Media, 2021. 384 с.

7. Создайте будущее с Visual Studio 2022: веб-сайт. URL: https://visualstudio.microsoft.com/ru/vs/ (дата звернення: 27.05.2022).

8. Trust Radius: веб-сайт. URL: https://www.trustradius.com/ (дата звернення: 27.05.2022).

9. SQL: веб-сайт. URL: https://uk.wikipedia.org/wiki/SQL (дата звернення: 28.05.2022).

10. Unit: Intro to SQL: Querying and managing data: веб-сайт. URL: https://www.khanacademy.org/computing/computer-programming/sql (дата звернення: 28.05.2022).

11. Structured Query Language (SQL): веб-сайт. URL: https://www.techtarget.com/searchdatamanagement/definition/SQL (дата звернення: 21.04.2022).

12. SQL Server Tutorial: веб-сайт. URL: https://www.sqlservertutorial.net/ (дата звернення: 21.04.2022).

13. Comparing database types: how database types evolved to meet different needs: веб-сайт. URL: https://www.prisma.io/dataguide/intro/comparing-database-types (дата звернення: 21.04.2022).

14. Paradox database native .NET reader: веб-сайт. URL: https://www.codeproject.com/Articles/101014/Paradox-database-native-NET-reader (дата звернення: 21.04.2022).

15. Korotkevich D. SQL Server Advanced Troubleshooting and Performance Tuning: Best Practices and Techniques / Dmitry Korotkevich – O'Reilly Media, 2022 – 536 с.

16. Законодавство України про охорону праці // Збірник нормативних документів у 4 т. -К.: Держнаглядохоронпраці; Основа, 2006 р

17. Про затвердження Правил охорони праці під час експлуатації електронно-обчислювальних машин [Електронний ресурс] – Режим доступу http://zakon3.rada.gov.ua/laws/show/z0382-99 - Загол. з екрану.

18. ДСТУ 2293-99 Охорона праці. Терміни та визначення основних понять. Київ -1999 р.

19. Москальова В. М. Основи охорони праці. Підручник. - Київ: ВД Професіонал, 2005.-666 с.

20. Гандзюк М. П., Желібо Е. П., Халімовський М. О. Основи охорони праці / За ред.. Гандзюка М. П. - К.: Каравела 2003 - 405 с.

21. Ткачук К. Н., Халімовський М. О., Зацарний В.В., та інші. Основи охорони праці: Підручник. -К.: Основа, 2006. -444 с.

22. Жидецький В.Ц. Основи охорои праці: Підручник. - К.: Основа, 2002. - 320 с.

23. Державні санітарні правила і норми роботи з ВДТ ЕОМ ДСанПІН 3.3.2.007-98 [Електронний ресурс] – Режим доступу: http://mozdocs.kiev.ua/view.php?id=2445 – Загол. з екрану.

ДОДАТОК А

Код функції CopyTableJet

```
Private Sub CopyTableJet(ByVal sTableName As String, dr As OleDbDataReader, _
ByRef cnDst As OleDbConnection)

    Dim oSchemaRows As Data.DataRowCollection = dr.GetSchemaTable.Rows
    Dim sRow As String
    Dim i As Integer
    Dim iRow As Integer = 0
    Dim iRowCount As Integer = 0

    'Get Header
    Dim sHeader As String = ""
    For i = 0 To oSchemaRows.Count - 1
        Dim sColumn As String = oSchemaRows(i)("ColumnName")
        If i <> 0 Then
            sHeader += ", "
        End If
        sHeader += PadSqlColumnName(sColumn)
    Next

    Dim sValues As String = ""

    While dr.Read()
        iRowCount += 1
        sRow = ""
```

```
        For i = 0 To oSchemaRows.Count - 1
            If sRow <> "" Then
                sRow += ", "
            End If

            sRow += GetValueString(dr.GetValue(i))
        Next

        If chkSQL2008.Checked Then
            If sValues <> "" Then sValues += ", "
            sValues += "(" & sRow & ")"

            If iRowCount >= 1000 Then
                Dim sSql1 As String = "INSERT INTO " &
PadSqlColumnName(sTableName) & _
                                      " (" & sHeader & ") VALUES " & sValues
                OpenConnections(cnDst)
                ExecuteSql(sSql1, cnDst)
                iRowCount = 0
                sValues = ""
            End If
        Else
            Dim sSql1 As String = "INSERT INTO " &
PadSqlColumnName(sTableName) & _
                                  " (" & sHeader & ") VALUES (" & sRow & ")"
            OpenConnections(cnDst)
            ExecuteSql(sSql1, cnDst)
```

```
        End If

        iRow += 1
        ProgressBar1.Value = Math.Min(ProgressBar1.Maximum, iRow)
        lbCount.Text = iRow.ToString()
        lbCount.Refresh()

        'Listen for the user to press Cancel button
        Windows.Forms.Application.DoEvents()
        If bStop Then
            Log("Copied table " & sTableName & " stopped. ")
            Exit While
        End If

    End While


    If chkSQL2008.Checked And sValues <> "" Then
        Dim sSql1 As String = "INSERT INTO " & PadSqlColumnName(sTableName)
& _
                              " (" & sHeader & ") VALUES " & sValues
        ExecuteSql(sSql1, cnDst)
    End If

End Sub
```

ДОДАТОК Б

Код функції GetCreateTableSqlFromParadox

```
Private Function GetCreateTableSqlFromParadox_
    (ByVal sTableName As String, dr As OleDbDataReader) As String

  Dim sb As New System.Text.StringBuilder()
  Dim oSchemaRows As Data.DataRowCollection = dr.GetSchemaTable.Rows
  Dim sKeyColumns As String = ""
  Dim i As Integer = 0

  sb.Append("CREATE TABLE " & PadSqlColumnName(sTableName) & " (" &
vbCrLf)

  For iCol As Integer = 0 To oSchemaRows.Count - 1
    Dim sColumn As String =
oSchemaRows(iCol).Item("ColumnName").ToString() & ""
    Dim sColumnSize As String =
oSchemaRows(iCol).Item("ColumnSize").ToString() & ""
    Dim sDataType As String =
oSchemaRows(iCol).Item("DATATYPE").FullName.ToString()
    Dim bAllowDBNull As Boolean = oSchemaRows(iCol).Item("AllowDBNull")
'Does not always work

    If i > 0 Then
      sb.Append(",")
      sb.Append(vbCrLf)
    End If
```

```
        sb.Append(PadSqlColumnName(sColumn))
        sb.Append(" " & PadAccessDataType(sDataType, sColumnSize))

        If bAllowDBNull Then
            sb.Append(" NULL")
        Else
            sb.Append(" NOT NULL")
        End If

        i += 1
    Next

    sb.Append(")")

    If i = 0 Then
        Return ""
    Else
        Return sb.ToString()
    End If

End Function
```

ДОДАТОК В

Код функції GetParadoxConnectionString

```
Function GetParadoxConnectionString(ByVal sFolderPath As String, ByVal sPassword As String) As String
   If sFolderPath = "" Then
      Return ""
   End If

   If sPassword <> "" Then
      Return "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & sFolderPath & ";_
      Extended Properties=Paradox 5.x;Jet OLEDB:Database Password=" & sPassword & ";"
   Else
      Return "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & sFolderPath & ";_
      Extended Properties=Paradox 5.x;"
   End If
End Function
```