МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕЛЕКТРИЧНОЇ ІНЖЕНЕРІЇ

ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра автоматизації та інформаційних систем

Звіт з лабораторної роботи №1

З навчальної дисципліни «Організація баз даних та знань»

Виконав: студент групи КН-19-1

Данілочкін Данііл

Перевірив: старший викладач

Васильєв Денис Олегович

КРЕМЕНЧУК 2022

**Тема:** **Огляд технологій доступу до даних.**

**Хід роботи**

1. Створення функцій початку та закінчення роботи бази даних.

static SqlConnection OpenOrCreateDataBase()

{

string connStr = @"Data Source=DESKTOP-M6OMTPE;

Initial Catalog=OBDZ;

Integrated Security=True";

SqlConnection conn = new SqlConnection(connStr);

try

{

conn.Open();

Console.WriteLine("БД відкрита");

}

catch (SqlException se)

{

if (se.Number == 4060)

{

Console.WriteLine("Зачекайте, йде створення БД");

conn.Close();

conn = new SqlConnection(@"Data Source=DESKTOP-M6OMTPE;Integrated Security=True");

SqlCommand cmdCreateDataBase = new SqlCommand(string.Format("CREATE DATABASE [{0}]", "OBDZ"), conn);

conn.Open();

Console.WriteLine("Посилаємо запит");

cmdCreateDataBase.ExecuteNonQuery();

conn.Close();

Thread.Sleep(5000);

conn = new SqlConnection(connStr);

conn.Open();

Console.WriteLine("БД створена");

}

}

return conn;

}

Код 1 – функція створення з’єднання з базою даних

static void CloseConnection(SqlConnection connection)

{

Console.WriteLine("З’єднання успішно завершено");

connection.Close();

connection.Dispose();

}

Код 2 – функція закриття з’єднання з базою даних

static void Main(string[] args)

{

SqlConnection connection = OpenOrCreateDataBase();

CloseConnection(connection);

Console.ReadKey();

}

Код 3 – код створення бази даних

Зображення, що містить текст

Автоматично згенерований опис

Рисунок 1 – результат роботи програми

1. Створення таблиць

static void CreateTable(string tableCreateCommand, SqlConnection connection)

{

SqlCommand createTableCommand = new SqlCommand(tableCreateCommand, connection);

try

{

createTableCommand.ExecuteNonQuery();

}

catch (Exception e)

{

Console.WriteLine("Помилка створення таблиці " + e.Message);

return;

}

Console.WriteLine("Таблиця створена успішно");

}

Код 4 – функція створення таблиці

static void Main(string[] args)

{

SqlConnection connection = OpenOrCreateDataBase();

CreateTable(@"CREATE TABLE Contracts(

Number INT IDENTITY(1, 1) NOT NULL PRIMARY KEY,

ClientSurname NVARCHAR(30) NOT NULL,

ClientName NVARCHAR(20) NOT NULL,

ClientPatronymic NVARCHAR(20) NOT NULL,

TelephoneNumber BIGINT NOT NULL,

Cost MONEY,

Document NVARCHAR(200),

AdditionalDescription NVARCHAR)", connection);

CreateTable(@"CREATE TABLE Sellers(

SellerName NVARCHAR(100) NOT NULL PRIMARY KEY,

TelephoneNumber BIGINT NOT NULL,

AdditionalDescription NVARCHAR(200))", connection);

CreateTable(@"CREATE TABLE MaterialsInSellers(

SellerName NVARCHAR(100) NOT NULL REFERENCES Sellers(SellerName),

MaterialName NVARCHAR(100) NOT NULL PRIMARY KEY,

Prise MONEY NOT NULL,

AdditionalDescription NVARCHAR(200))", connection);

CreateTable(@"CREATE TABLE UsedMaterials(

ContractNumber INT NOT NULL REFERENCES Contracts(Number),

MaterialName NVARCHAR(100) REFERENCES MaterialsInSellers(MaterialName),

Cost MONEY NOT NULL,

MaterialCount INT NOT NULL

PRIMARY KEY (ContractNumber, MaterialName))", connection);

CreateTable(@"CREATE TABLE Employe(

RegistrationNumber INT IDENTITY(1,1) PRIMARY KEY,

Surname NVARCHAR(30) NOT NULL,

Name\_ NVARCHAR(20) NOT NULL,

Patronymic NVARCHAR(20) NOT NULL,

Team NVARCHAR(50),

Position NVARCHAR(100),

Salary MONEY NOT NULL,

DateStartWork DATE NOT NULL,

DateStopWork DATE,

AdditionalDescription NVARCHAR(200),

TelephoneNumber BIGINT NOT NULL)", connection);

CreateTable(@"CREATE TABLE Teams(

TeamName NVARCHAR(50) NOT NULL PRIMARY KEY,

Cheif INT REFERENCES Employe(RegistrationNumber),

AdditionalDescription NVARCHAR(200))", connection);

CreateTable(@"CREATE TABLE workSchedule(

ContractNumber INT NOT NULL FOREIGN KEY REFERENCES Contracts(Number),

WorkDescription NVARCHAR(100) NOT NULL,

Team NVARCHAR(50) NOT NULL REFERENCES Teams(TeamName),

StartDate DATE NOT NULL,

EndDate DATE NOT NULL,

AdditionalDescription NVARCHAR(200))", connection);

CreateTable(@"CREATE TABLE TeamComposition(

RegistrationNumber INT NOT NULL FOREIGN KEY REFERENCES Employe(RegistrationNumber),

TeamName NVARCHAR(50) NOT NULL FOREIGN KEY REFERENCES Teams(TeamName))", connection);

CloseConnection(connection);

Console.ReadKey();

}

Код 5 – код створення таблиць

Зображення, що містить текст

Автоматично згенерований опис

Рисунок 2 – результат роботи програми

1. Заповнення таблиць даними

static void Main(string[] args)

{

SqlConnection connection = OpenOrCreateDataBase();

SqlCommand cmd = new SqlCommand(@"INSERT INTO Employe (Surname, Name\_, Patronymic, Team, Position, Salary, DateStartWork, TelephoneNumber) VALUES

('Дейнеко', 'Василь', 'Вікторович', 'бригадир каменярів', 1, 20000, CAST('2016-05-27' as datetime), 380671122333),

('Кравець', 'Іван', 'Петрович', 'каменяр', 1, 8000, CAST('2016-06-13' as datetime), 380671122331),

('Кучеренко', 'Петро', 'Васильович', 'каменяр', 1, 8000, CAST('2016-06-10' as datetime), 380671122332),

('Амосов', 'Микола', 'Сергійович', 'каменяр', 1, 8000, CAST('2016-06-17' as datetime), 380671122334),

('Москальов', 'Федір', 'Олегович', 'каменяр', 1, 8000, CAST('2016-06-12' as datetime), 380671122335),

('Дорошенко', 'Сергій', 'Вікторович', 'каменяр', 1, 8000, CAST('2016-06-13' as datetime), 380671122336),

('Тимченко', 'Володимир', 'Микитович', 'бригадир малярів', 1, 12000, CAST('2016-05-17' as datetime), 380671223331),

('Іванець', 'Микола', 'Сергійович', 'маляр', 1, 8000, CAST('2016-05-29' as datetime), 380671223332),

('Коношевич', 'Ольга', 'Володимирівна', 'маляр', 1, 8000, CAST('2016-06-07' as datetime), 380671223333),

('Коваль', 'Мирослав', 'Всеволодович', 'бригадир покрівельників', 1, 15000, CAST('2016-06-9' as datetime), 380671122355),

('Шмигаль', 'Юрій', 'Юрійович', 'покрівельник', 1, 8000, CAST('2016-06-28' as datetime), 380671122365),

('Заєць', 'Микита', 'Васильович', 'покрівельник', 1, 8000, CAST('2016-07-21' as datetime), 380671122375),

('Куцихвіст', 'Данило', 'Андрійович', 'сантехнік', 1, 15000, CAST('2016-07-03' as datetime), 380671122433),

('Пальчун', 'Назар', 'Миколаєвич', 'електрик', 1, 12000, CAST('2016-06-20' as datetime), 380671122533),

('Почечун', 'Павло', 'Некіфорович', 'директор фірми', 1, 30000, CAST('2016-05-01' as datetime), 380681122333),

('Некрасова', 'Любов', 'Іванівна', 'бухгалтер', 1, 15000, CAST('2016-05-13' as datetime), 380681222333),

('Підгірна', 'Людмила', 'Миколаєвна', 'юрист', 1, 13000, CAST('2016-05-23' as datetime), 380681522333),

('Непийпиво', 'Наталія', 'Петрівна', 'менеджер по роботі з клієнтами', 1, 20000, CAST('2016-05-27' as datetime), 380671122733);

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO Teams (TeamName, Cheif) VALUES

('каменярі', 1),

('малярі', 7),

('покрівельники', 10),

('електрик/сантехнік', 14),

('невиробничий відділ', 15);

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO TeamComposition (RegistrationNumber, TeamName) VALUES

(1, 'каменярі'),

(2, 'каменярі'),

(3, 'каменярі'),

(4, 'каменярі'),

(5, 'каменярі'),

(6, 'каменярі'),

(7, 'малярі'),

(8, 'малярі'),

(9, 'малярі'),

(10, 'покрівельники'),

(11, 'покрівельники'),

(12, 'покрівельники'),

(13, 'електрик/сантехнік'),

(14, 'електрик/сантехнік'),

(15, 'невиробничий відділ'),

(16, 'невиробничий відділ'),

(17, 'невиробничий відділ'),

(18, 'невиробничий відділ');

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO Sellers (SellerName, TelephoneNumber, AdditionalDescription) VALUES

('ЛісБуд', 380975156019, 'строительный лес'),

('БудІстПром', 380971714428, 'блоки стен, все для бетона'),

('ТеплоДім', 380977199845, 'смеси, утеплители, монтажные мелочи'),

('СерцеДому', 380972219793, 'сантехника, електрика, утеплители, монтажные мелочи'),

('Лісовик', 380973474814, 'строительный лес, монтажные мелочи');

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO Contracts(ClientSurname, ClientName, ClientPatronymic, TelephoneNumber, Cost) VALUES

('Бондар', 'Артем', 'Тимофійович', 380675874528, 1000000),

('Поліщук', 'Іван', 'Трохимович', 380678254781, 4000000);

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO MaterialsInSellers (SellerName, MaterialName, Prise) VALUES

('ЛісБуд', 'брус 100\*100 6м ЛісБуд', 230),

('ЛісБуд', 'брус 100\*50 6м ЛісБуд', 120),

('ЛісБуд', 'брус 50\*50 6м ЛісБуд', 60),

('ЛісБуд', 'брус 200\*100 6м ЛісБуд', 450),

('БудІстПром', 'цегла червона БудІстПром', 6.25),

('БудІстПром', 'цегла біла силікатна БудІстПром', 8.35),

('БудІстПром', 'цегла вогнетривка БудІстПром', 15),

('БудІстПром', 'пінобетон 600\*300\*200 БудІстПром', 54),

('БудІстПром', 'пінобетон 600\*200\*100 БудІстПром ', 21),

('БудІстПром', 'портланд цемент 400 БудІстПром', 65),

('БудІстПром', 'портланд цемент 500 БудІстПром', 80),

('БудІстПром', 'портланд цемент 1000 БудІстПром', 145),

('ТеплоДім', 'цемент ТеплоДім', 50),

('ТеплоДім', 'клей для плитки ТеплоДім', 123),

('ТеплоДім', 'сатенгіпс ТеплоДім', 157),

('ТеплоДім', 'клей для плитки екстра ТеплоДім', 170),

('ТеплоДім', 'пінопласт 1000\*1000\*150 ТеплоДім', 208),

('ТеплоДім', 'пінопласт 1000\*1000\*100 ТеплоДім', 140),

('ТеплоДім', 'пінопласт 1000\*1000\*50 ТеплоДім', 68),

('ТеплоДім', 'пінопласт 1000\*1000\*20 ТеплоДім', 17.60),

('СерцеДому', 'унітаз модель1 СерцеДому', 1200),

('СерцеДому', 'унітаз модель2 СерцеДому', 1600),

('СерцеДому', 'унітаз модель3 СерцеДому', 3400),

('СерцеДому', 'ванна модель1 СерцеДому', 2700),

('СерцеДому', 'ванна модель2 СерцеДому', 3500),

('СерцеДому', 'ванна модель3 СерцеДому', 5400),

('СерцеДому', 'умивальник модель1 СерцеДому', 500),

('СерцеДому', 'умивальник модель2 СерцеДому', 700),

('СерцеДому', 'пінопласт 1000\*1000\*150 СерцеДому', 192),

('СерцеДому', 'пінопласт 1000\*1000\*100 СерцеДому', 138),

('СерцеДому', 'пінопласт 1000\*1000\*50 СерцеДому', 63),

('СерцеДому', 'пінопласт 1000\*1000\*20 СерцеДому', 14),

('Лісовик', 'брус 100\*100 6м Лісовик', 220),

('Лісовик', 'брус 100\*50 6м Лісовик', 115),

('Лісовик', 'брус 50\*50 6м Лісовик', 57),

('Лісовик', 'брус 200\*100 6м Лісовик', 450);

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO MaterialsInSellers (SellerName, MaterialName, Prise, AdditionalDescription) VALUES

('БудІстПром', 'пісок річковий БудІстПром', 170, 'ціна за тонну'),

('ТеплоДім', 'саморізи ТеплоДім', 130, 'за кілограм'),

('ТеплоДім', 'шурупи ТеплоДім', 80, 'за кілограм'),

('ТеплоДім', 'дюбель для пинопласта ТеплоДім', 92, 'за 100шт'),

('СерцеДому', 'водяний насос Дельта СерцеДому', 1600, '2.16 м3/час'),

('СерцеДому', 'водяний насос PEDROLLO JSWm 2CX СерцеДому', 2150, '3.3 м3/час'),

('СерцеДому', 'саморізи СерцеДому', 130, 'за кілограм'),

('СерцеДому', 'шурупи СерцеДому', 90, 'за кілограм'),

('СерцеДому', 'дюбель для пинопласта СерцеДому', 103, 'за 100шт'),

('Лісовик', 'саморізи Лісовик', 140, 'за кілограм'),

('Лісовик', 'шурупи Лісовик', 85, 'за кілограм');

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO UsedMaterials(ContractNumber, MaterialName, Cost, MaterialCount) VALUES

(1, 'цегла червона БудІстПром', 6.25, 10000),

(1, 'пісок річковий БудІстПром', 170, 10),

(1, 'портланд цемент 500 БудІстПром', 80, 100),

(2, 'пінобетон 600\*300\*200 БудІстПром', 6.25, 300),

(2, 'портланд цемент 500 БудІстПром', 80, 100),

(2, 'брус 200\*100 6м ЛісБуд', 450, 2),

(2, 'брус 100\*100 6м ЛісБуд', 230, 6),

(2, 'брус 50\*50 6м ЛісБуд', 60, 30),

(2, 'пінопласт 1000\*1000\*100 СерцеДому', 138, 200),

(2, 'дюбель для пинопласта СерцеДому', 103, 2);

", connection);

cmd.ExecuteNonQuery();

cmd = new SqlCommand(@"INSERT INTO workSchedule(ContractNumber, WorkDescription, Team, StartDate, EndDate) VALUES

(1, 'викладка стін', 'каменярі', CAST('2018-05-13' as datetime), CAST('2018-06-1' as datetime)),

(2, 'викладка стін', 'каменярі', CAST('2018-06-5' as datetime), CAST('2018-06-18' as datetime)),

(2, 'будівництво криші', 'покрівельники', CAST('2018-06-19' as datetime), CAST('2016-06-21' as datetime)),

(2, 'утеплення стін', 'каменярі', CAST('2018-06-19' as datetime), CAST('2018-06-26' as datetime)),

(2, 'утеплення стін', 'каменярі', CAST('2018-06-19' as datetime), CAST('2018-06-26' as datetime));

", connection);

cmd.ExecuteNonQuery();

CloseConnection(connection);

Console.ReadKey();

}

Код 6 – код заповнення таблиць

Зображення, що містить текст

Автоматично згенерований опис

Рисунок 3 – результат виконання коду

**Висновок:** в результаті виконання лабораторної роботи було створено і наповнено базу даних по темі «Будівельна організація».

**Контрольні питання**

1. Що таке ADO.NET?

Інструментарій для роботи з даними з джерела, зокрема з базами даних.

1. Яке призначення наступних об’єктів?

* Connection;

Встановлює з’єднання з конкретним джерелом даних

* Command;

Виконує команду у джерелі даних

* DataReader;

Зчитує з джерела одиночний потік даних

* DataAdapter;

Заповнює DataSet та виконує обновлення в джерелі даних

* Transaction;

Прикріплює команди до транзакцій в джерелі даних

* Parameter.

Визначає вхідні, вихідні, повертаємі параметри команд та збережених процедур

1. Перерахуйте базові класи ADO.NET та вкажіть їх призначення?

SqlConnection – надає відкрите підключення до бази даних.

SqlCommand – виконує команду у джерелі даних

SqlDataReader – надає можливість зчитування потоку з БД

SqlParameter – задає параметри виконання команди