Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут прикладної математики та фундаментальних наук

Кафедра прикладної математики

**Звіт**

про виконання лабораторної роботи №2

з курсу «Організація баз даних та знань»

**Виконав:**

студент групи ПМ-22

Янів П.В.

**Перевірив:**

Пабирівський В.В.

Львів-2024

**Варіант №11 “Оплата праці”**

На підприємстві працюють працівники, кожен з яких належить до певного структурного підрозділу. Кожен структурний підрозділ має свою назву, код, керівника. Кожен працівник працює на певній посаді та характеризується табельним номером, прізвищем, ім'ям, по-батькові, паспортними даними, датою та місцем народження, домашньою адресою.

На підприємстві встановлені такі форми оплати праці

* основна заробітна плата у вигляді погодинної оплати за кожну відпрацьовану годину, причому для кожної посади існує свій тариф за годину,
* додаткова заробітна плата – оплата за роботу у понад встановлені норми, яка полягає в подвійній оплаті за кожну додатково відпрацьовану годину та потрійній – за кожну відпрацьовану годину у вихідний (субота, неділя) та святковий день.

Працівник кожного року має відпустку. Система повинна зберігати дані про всі відпустки працівників. Одночасно перебувати в відпустці можуть не більше 15% працівників одного відділу. Оплата відпусток проводиться так: S=M/(365-C)\*N, де S – сума відпускних, M – сумарний заробіток працівника за останні 12 місяців, 365 – кількість днів у році, С – число святкових днів, N – тривалість відпустки в календарних днях.

Система повинна вести облік робочого часу (фактично відпрацьовані години працівником за кожен день), нараховувати місячну зарплату всім працівникам, розраховувати для кожного працівника заробітну плату до видачі на руки (заробітна плата до видачі на руки – це нарахована заробітна плата мінус соціальний податок 20%).

Система повинна вести облік виплати зарплати працівника.

В системі повинні діяти такі додаткові обмеження: працівник не може працювати в декількох відділах та на декількох посадах одночасно; основна зарплата працівника не може бути меншою від встановленої мінімальної зарплати, працівнику забороняється нараховувати зарплату після його звільнення; виплата зарплати працівникам повинна здійснюватись не пізніше 10 числа наступного місяця, який слідує за відпрацьованим, при порушенні цього правила працівнику додається до зарплати за поточний місяць пеня в розмірі 0,1% від суми заборгованості за кожен прострочений день.

* розрахункову відомість заробітної плати для відділу,
* відомість на отримання заробітної плати для відділу,
* відомість про роботу працівника впродовж певного періоду,
* відомість про борг підприємства перед працівниками.

**Завдання**:

## Лабораторна робота № 2. “Написання SQL‑запитів”

Завдання лабораторної роботи виконувати в призначеній предметній області. Змістовне призначення запитів визначати самостійно виходячи з логіки предметної області.

Створити запити типу:

* SELECT на базі однієї таблиці з використанням сортування, накладенням умов зі зв’язками OR та AND.
* SELECT з виводом обчислюваних полів (виразів) в колонках результату.
* SELECT на базі кількох таблиць з використанням сортування, накладенням умов зі зв’язками OR та AND.
* SELECT на базі кількох таблиць з типом поєднання Outer Join.
* SELECT з використанням операторів Like, Between, In, Exists, All, Any.
* SELECT з використанням підсумовування та групування.
* SELECT з використанням під-запитів в частині Where.
* SELECT з використанням під-запитів в частині From.
* ієрархічний SELECT‑запит.
* SELECT‑запит типу CrossTab.
* UPDATE на базі однієї таблиці.
* UPDATE на базі кількох таблиць.
* Append (INSERT) для додавання записів з явно вказаними значеннями.
* Append (INSERT) для додавання записів з інших таблиць.
* DELETE для видалення всіх даних з таблиці.
* DELETE для видалення вибраних записів таблиці.

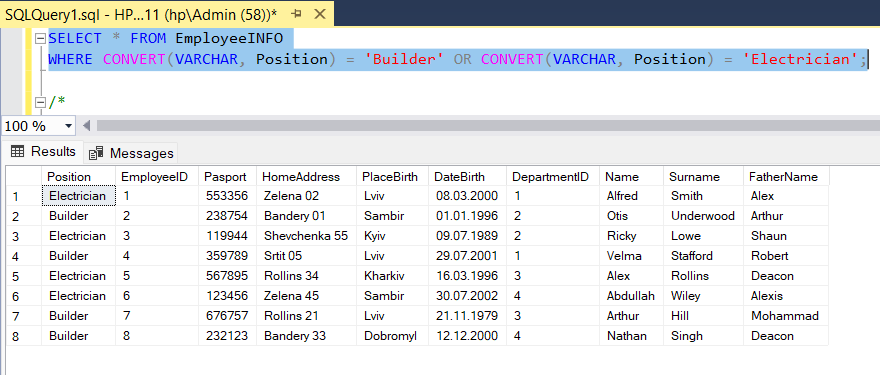
Запити можуть бути складними і поєднувати у собі одразу декілька типів (згідно завдання). У роботі має бути не менше двох таких складних запитів. При цьому обов’язково необхідно відобразити у звіті відповідні особливості запиту.

**SQl запити:**

SELECT \* FROM EmployeeINFO

WHERE CONVERT(VARCHAR, Position) = 'Builder' OR CONVERT(VARCHAR, Position) = 'Electrician';

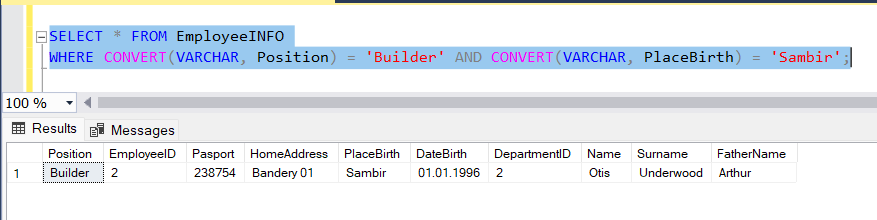
Цей запит виводить працівників з посадою електрик та будівельник

****

SELECT \* FROM EmployeeINFO

WHERE CONVERT(VARCHAR, Position) = 'Builder' AND CONVERT(VARCHAR, PlaceBirth) = 'Sambir';

Цей запит виводить працівників з посадою будівельник та місцем народження Самбір

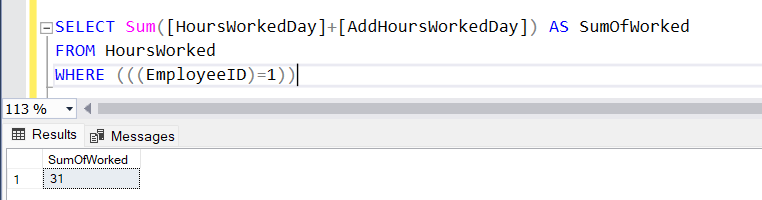


SELECT Sum([HoursWorkedDay]+[AddHoursWorkedDay]) AS SumOfWorked

FROM HoursWorked

WHERE (((EmployeeID)=1))

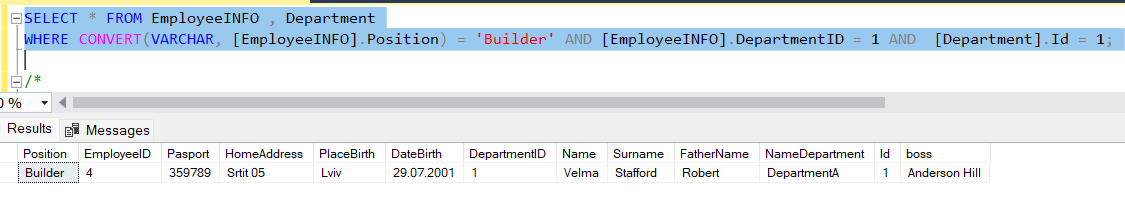
Цей запит виводить суму відпрацьованих годин працівника з табельним номером 1



SELECT \* FROM EmployeeINFO , Department

WHERE CONVERT(VARCHAR, [EmployeeINFO].Position) = 'Builder' AND [EmployeeINFO].DepartmentID = 1 AND [Department].Id = 1;

Цей запит виводить працівників з посадою будівельник та табельним номером 1



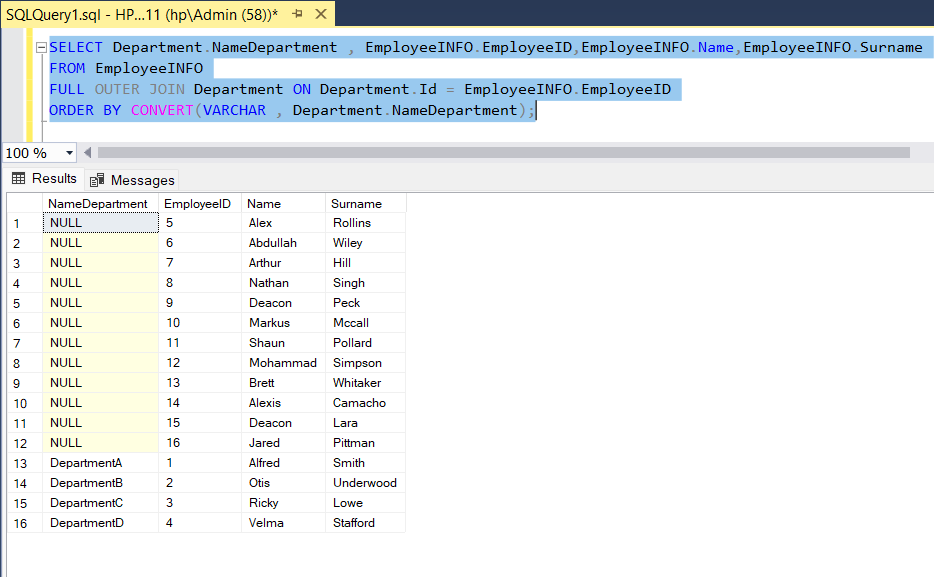
SELECT Department.NameDepartment , EmployeeINFO.EmployeeID,EmployeeINFO.Name,EmployeeINFO.Surname

FROM EmployeeINFO

FULL OUTER JOIN Department ON Department.Id = EmployeeINFO.EmployeeID

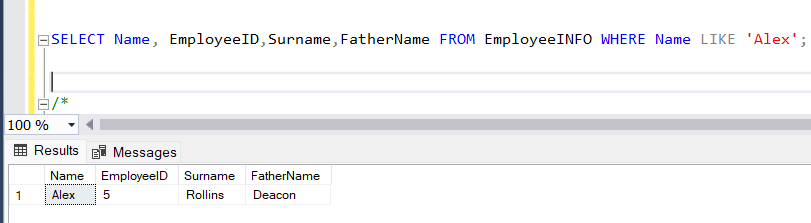
ORDER BY CONVERT(VARCHAR , Department.NameDepartment);

Цей запит виводить працівників у яких табельний номер співпадає з номером структурного підрозділу



SELECT Name, EmployeeID,Surname,FatherName FROM EmployeeINFO WHERE Name LIKE 'Alex';

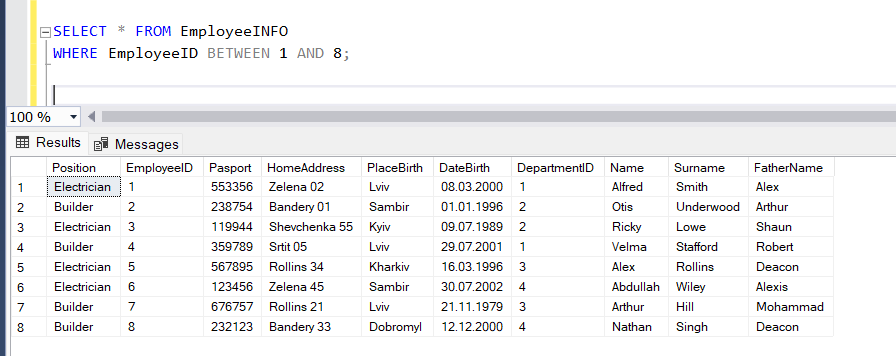
Цей запит виводить працівників з іменем Алекс



SELECT \* FROM EmployeeINFO

WHERE EmployeeID BETWEEN 1 AND 8;

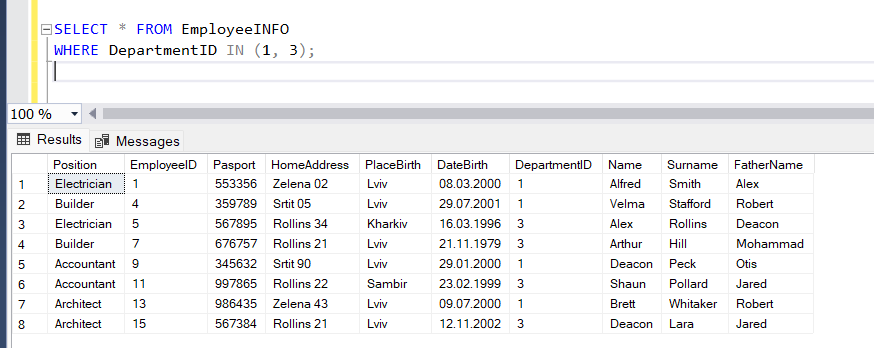
Цей запит виводить працівників з табельним номером між 1 та 8



SELECT \* FROM EmployeeINFO

WHERE DepartmentID IN (1, 3);

Цей запит виводить працівників з кодом структурного підрозділу 1 або 3

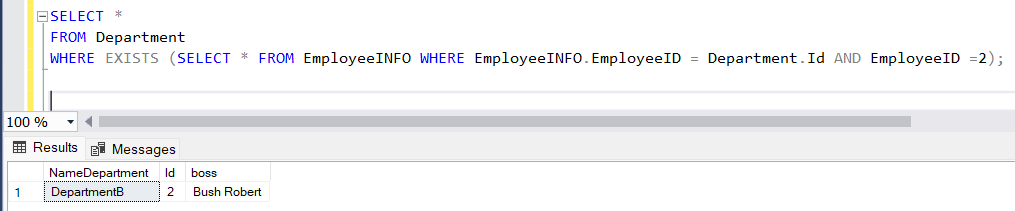


SELECT \*

FROM Department

WHERE EXISTS (SELECT \* FROM EmployeeINFO WHERE EmployeeINFO.EmployeeID = Department.Id AND EmployeeID =2);

Цей запит виводить назву структурного підрозділу з кодом 2

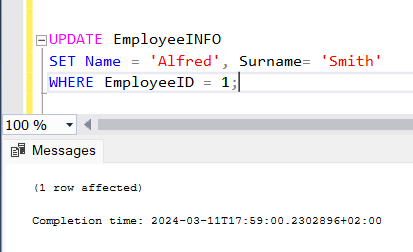


UPDATE EmployeeINFO

SET Name = 'Alfred', Surname= 'Smith'

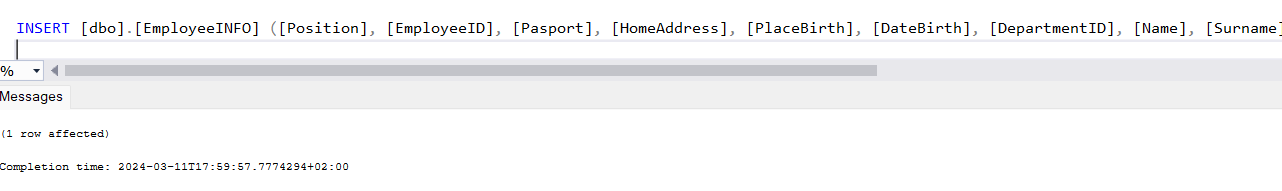
WHERE EmployeeID = 1;

Цей запит переписує інформацію про працівника з табельним номером 1



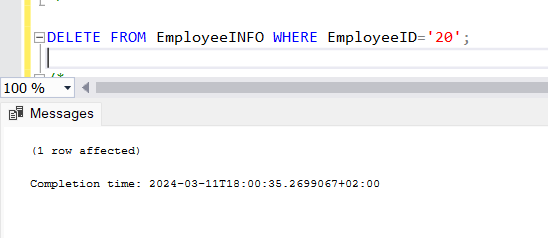
INSERT [dbo].[EmployeeINFO] ([Position], [EmployeeID], [Pasport], [HomeAddress], [PlaceBirth], [DateBirth], [DepartmentID], [Name], [Surname], [FatherName]) VALUES (N'Electrician', 20, 553356, N'Zelena 02', N'Lviv', N'08.03.2000', 1, N'Robert ', N'Schaefer', N'Alex ')

Цей запит записує в таблицю нового працівника



DELETE FROM EmployeeINFO WHERE EmployeeID='20';

Цей запит видаляє з таблиці працівника з табельним номером 20



**GitHub репозиторій:** https://github.com/PetroYaniv/DBSQLYanivPetroPM22.git

**Висновок:** Написав SQL-запити які наведені у завданні до лабораторної роботи №2 для бази даних створеної у лабораторній роботі №1. До кожного запиту зробив скріншот екрану та описав їх призначення.