**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ** НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, КОМП’ЮТЕРНОЇ ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

КУРСОВИЙ ПРОЕКТ

з дисципліни «Бази даних»

Тема: База даних поштового відділення

Виконала студентка 3 курсу 322 гр. кафедри інженерії програмного забезпечення Лясковська Варвара Юріївна

Керівник курсової роботи:

Ст.викл. кафедри ІПЗ Марченко О.І.

Київ 2020

***Зміст***

[**ЗАВДАННЯ 3**](#_Toc59449838)

[**МЕТА, ЦІЛІ ТА ЗАДАЧІ СТВОРЕННЯ БД 4**](#_Toc59449839)

[**АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ 5**](#_Toc59449840)

[**ПРОЕКТУВАННЯ БД 6**](#_Toc59449841)

[**4.1. Логічне проектування 6**](#_Toc59449842)

[**4.2. Фізичне проектування 9**](#_Toc59449844)

[**ЗАПИТИ ДО БД 15**](#_Toc59449845)

[**ВИСНОВОК 17**](#_Toc59449846)

ЗАВДАННЯ

**Варіант 4:** База даних поштового відділення

**Інформація:** абоненти, абонентськи скриньки, підписні видання, праівники, поштова скринька, поштове відділення, каталог(ходовик)

**Функції:** пошук абонента, пошук поштаря, список абонентів, выписуючих визначене видання, список абонентів, обслуговуваемих поштарьом, список абонентів, які мають абонентськи скриньки, обчислення абонентів, проживаючих на вказаної вулиці (у заданому домі).

Завданням курсового проекту з дисципліни «Бази даних» є створення бази даних поштового відділення для покращення навичок побудови та розуміння самої БД. Також завданням є створення зв’язків між сутностями і атрибутів, що будуть мати правильний функціонал та дозволять користувачам даної БД отримувати усі необхідні дані.

МЕТА, ЦІЛІ ТА ЗАДАЧІ СТВОРЕННЯ БД

Дана БД створена для проведення оптимізації роботи працівників поштового відділення, поліпшення зберігання даних працівників(№ посвідчення, посада, заробітня плата, дата прийому на роботу)

Також за допомогою БД зручніше буде проводити необхідні запити та пошуки, додавати або видаляти робітників.

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

Поштове відділення або відділення зв'язку — об'єкт поштового зв'язку, який здійснює прийом, сортування, відправку, обробку і доставку пошти.

Створення БД поштового відділення допоможе працівникам орієнтуватися в процусі роботи на відділення, поліпшить методи пошуку необхідної інформації та поліпшить роботу із клієнтами.

**Поштове відділення надає такі послуги:**

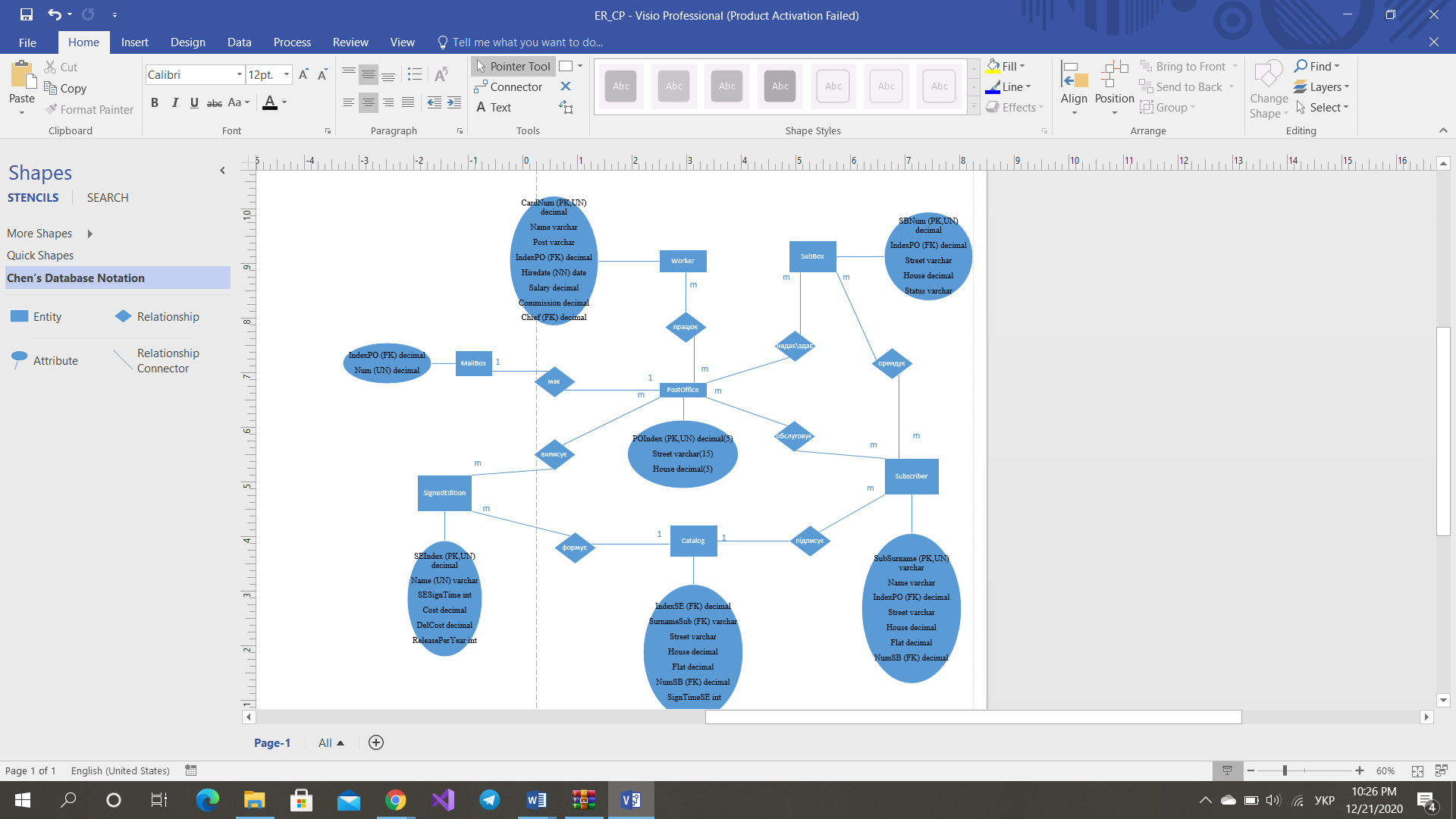
* Супутні послуги поштового зв'язку
  + знайдення та/або зазначення [поштового індексу](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D1%88%D1%82%D0%BE%D0%B2%D0%B8%D0%B9_%D1%96%D0%BD%D0%B4%D0%B5%D0%BA%D1%81)
  + надання у користування абонементних скриньок;
  + повідомлення — запит про вручення поштових відправлень (поштових переказів);
* Передплата та розповсюдження періодичних видань

При роботі з системою працівник обов’язково повинен мати можливість виконувати наступні задачі:

* пошук абонента,
* пошук поштаря,
* список абонентів, виписуючих періодисне видання,
* список абонентів, які мають абонентськи скриньки,
* обчислення абонентів, проживаючих на вказаної вулиці (у заданому домі).
* обчислення : мінімальних, максимальних, середніх значень та для звітування роботи поштового відділення.
* Підрахунок кількості видань, абонскриньок тощо.

ПРОЕКТУВАННЯ БД

* 1. Логічне проектування



1. ***Список сутностей***

Сильні сутності : абонент, періодичне видання, працівник, поштове відділення

Слабкі сутності : абонентьска скринька, каталог, поштова скринька

1. ***Список атрибутів сутностей***

Абонент:

* Фамілія - первинний ключ, унікальний атрибут, текстовий тип
* Ім’я - простий атрибут, текстовий тип
* Індекс ПВ – зовнішній ключ, цілочисельне значення
* Вулиця - простий атрибут, текстовий тип
* Дім - простий атрибут, цілочисельне значення
* Квартира - простий атрибут, цілочисельне значення
* Номер абонскриньки - зовнішній ключ, цілочисельне значення

Періодичне видання:

* Індекс ПВ - первинний ключ, унікальний атрибут, цілочисельне значення
* Назва - унікальний атрибут, текстовий тип
* Період підписки – простий атрибут, цілочисельне значення
* Вартість – простий атрибут, цілочисельне значення
* Вартість доставки – простий атрибут, цілочисельне значення
* Періодичність виходу на рік – простий атрибут, цілочисельне значення

Працівник:

* № посвідчення - первинний ключ, унікальний атрибут, цілочисельне значення
* Ім’я – простий атрибут, текстовий тип
* Посада – простий атрибут, текстовий тип
* Індекс ПВ - зовнішній ключ, цілочисельне значення
* Дата прийому - простий атрибут, тип Дата
* Зарплата - простий атрибут, цілочисельне значення
* Надбавка - простий атрибут, цілочисельне значення
* Керівник - зовнішній ключ, цілочисельне значення

Поштове відділення:

* Індекс ПВ - первинний ключ, унікальний атрибут, цілочисельне значення
* Вулиця - простий атрибут, текстовий тип
* Дім - простий атрибут, цілочисельне значення

Абонентська скринька:

* Номер абонскриньки - первинний ключ, унікальний атрибут, цілочисельне значення
* Індекс ПВ - зовнішній ключ, цілочисельне значення
* Вулиця - простий атрибут, текстовий тип
* Дім - простий атрибут, цілочисельне значення
* Статус - простий атрибут, текстовий тип

Каталог(Ходовик):

* Індекс - зовнішній ключ, цілочисельне значення Назва - унікальний атрибут, текстовий тип
* Фамілія - зовнішній ключ, текстовий тип
* Вулиця - простий атрибут, текстовий тип
* Дім - простий атрибут, цілочисельне значення
* Квартира - простий атрибут, цілочисельне значення
* Номер абонскриньки - зовнішній ключ, цілочисельне значення
* Періодичність виходу на рік – простий атрибут, цілочисельне значення

Поштова скринька :

* Індекс ПВ – зовнішній ключ, цілочисельне значення
* № поштової скриньки - унікальний атрибут, цілочисельне значення

1. ***Опис взаємозв’язків між сутностями***

* ***«працює»***  - тип зв’язку М:М. У місті є багато поштових відділень і працює там декілька робітників.
* ***«має»*** - тип зв’язку 1:1. Кожне поштове відділення має лише свою поштову скриньку.
* ***«виписує»*** - тип зв’язку М:М. Поштове відділення виписує безліч періодичних видань. Водночас це роблять декілька поштових відділень
* ***«формує»*** - тип зв’язку 1:М. Періодичних видань багато, але разом вони формують єдиний документ Ходовик, де знаходиться інформація про всі видання, які підписуються абонентами.
* ***«підписує»*** - тип зв’язку 1:М. Багато користувачів підписують періодичні видання, але інформація зберігається в єдиному документі.
* ***«обслуговує»*** - тип зв’язку М:М. Безліч поштових відділень обслуговують безліч абонентів
* ***«надає\здає»*** - тип зв’язку М:М. У безлічі поштових відділень є багато абоненскриньок***.***
* ***«орендує»*** - тип зв’язку М:М. 2 варіанти: 1 – один абонент орендує одну сриньку і таких абонентів безліч. 2 – один абонент може орендувати декілька абонскриньок і таких абонентів безліч.
  1. Фізичне проектування

create database CP;

use CP;

**TABLE SUBSCRIBER**

create table Subscriber(

SubSurname varchar(15) primary key,

Name varchar(15),

IndexPO decimal(5),

Street varchar(15),

House decimal(5),

Flat decimal(5),

NumSB decimal(3),

constraint SubIndex foreign key (IndexPO) references PostOffice (POIndex),

constraint SubNum foreign key (NumSB) references SubBox (SBNum)

);

**TABLE SUBBOX**

create table SubBox

(

SBNum decimal(3) primary key unique check(SBNum between 1 and 125),

IndexPO decimal(5),

Street varchar(15),

House decimal(5),

Status varchar(5) check(Status in ('free', 'busy')),

constraint BoxIndex foreign key (IndexPO) references PostOffice (POIndex),

constraint IndexNum UNIQUE (IndexPO, SBNum)

);

**TABLE POSTOFFICE**

create table PostOffice

(

POIndex decimal(5) primary key unique not null,

Street varchar(15),

House decimal(5)

);

**TABLE WORKER**

create table Worker

(

CardNum decimal(6) primary key unique,

Name varchar(15),

Post varchar(15),

IndexPO decimal(5),

Hiredate date not null check(Hiredate > date'1990-01-01'),

Salary decimal(6,2),

Commission decimal(4,2),

Chief decimal(6),

constraint WorkIndex foreign key (IndexPO) references PostOffice (POIndex),

constraint Chief foreign key (Chief) references Worker (CardNum),

constraint post\_restriction check(Post in('Head', 'Dean', 'operator', 'postman')),

constraint salary\_restriction check(Salary > 4000),

constraint comm\_restriction check(Commission > -1),

constraint chief\_restriction check(Chief != CardNum)

);

**TABLE MAILBOX**

create table MailBox

(

IndexPO decimal(5),

Num decimal(3) check(Num between 01 and 099),

constraint MailIndex foreign key (IndexPO) references PostOffice (POIndex)

);

**TABLE CATALOG**

create table Catalog

(

IndexSE decimal(5),

SurnameSub varchar(15),

Street varchar(15),

House decimal(5),

Flat decimal(5),

NumSB decimal(3),

SignTimeSE int check(SignTimeSE in ('1', '2', '3', '6', '12')),

constraint CatNum foreign key (NumSB) references SubBox (SBNum),

constraint CatSurname foreign key (SurnameSub) references Subscriber (SubSurname),

constraint IndexSE foreign key (IndexSE) references SignedEdition (SEIndex)

);

**TABLE SIGNEDEDITION**

create table SignedEdition

(

SEIndex decimal(5) primary key unique,

Name varchar(15) unique,

SESignTime int check(SESignTime in ('1', '2', '3', '6', '12')),

Cost decimal(4) check(Cost between 10 and 1000),

DelCost decimal(3) check(DelCost between 1 and 100),

ReleasePerYear int check(ReleasePerYear in ('1', '2', '4', '12', '24')),

constraint IndexName UNIQUE (SEIndex, Name)

);

**ПРИКЛАДИ ЗАПОВНЕННЯ КОРТЕЖЕЙ**

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `Flat`) VALUES ('Liaskovska', 'Varvara', '4107', 'Tatarska', '2', '104');

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `NumSB`) VALUES ('Grachova', 'Larysa', '4107', 'Tatarska', '38', '24');

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `Flat`) VALUES ('Syzyntseva', 'Elvira', '4050', 'Dehtiarivska', '6', '92');

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `NumSB`) VALUES ('Bukin', 'Henna', '4050', 'Illienka', '6', '2');

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `Flat`) VALUES ('Koba', 'Sasha', '4107', 'Tatarska', '20', '74');

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `NumSB`) VALUES ('Petrychenko', 'Iryna', '4107', 'Tatarska', '38', '13');

INSERT INTO subscriber (`SubSurname`, `Name`, `IndexPO`, `Street`, `House`, `NumSB`) VALUES ('Kyrychenko', 'Valeria', '1044', 'Kreschatyk', '36', '12');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('3', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('4', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('5', '4107', 'Tatarska', '38', 'busy');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('6', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('7', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('8', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('9', '4107', 'Tatarska', '38', 'busy');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('10', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('11', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('12', '4107', 'Tatarska', '38', 'free');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('13', '4107', 'Tatarska', '38', 'busy');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('14', '4107', 'Tatarska', '38', 'busy');

INSERT INTO `cp`.`subbox` (`SBNum`, `IndexPO`, `Street`, `House`, `Status`) VALUES ('15', '4107', 'Tatarska', '38', 'free');

insert into mailbox values ('4107','12');

insert into mailbox values ('4119','24');

insert into mailbox values ('4050','6');

insert into mailbox values ('1044','36');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16071', 'Limb1', '1', '20', '1', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16073', 'Limb3', '3', '60', '2', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16076', 'Limb6', '6', '100', '3', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('94421', 'Eva', '12', '350', '10', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('94422', 'Eva6', '6', '210', '5', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16056', 'Sprint6', '6', '190', '5', '24');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16058', 'Sprint', '12', '380', '10', '24');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16081', 'Oraky1', '1', '10', '1', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16083', 'Oraky3', '3', '33', '3', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16086', 'Oraky6', '6', '65', '6', '12');

INSERT INTO `cp`.`signededition` (`SEIndex`, `Name`, `SESignTime`, `Cost`, `DelCost`, `ReleasePerYear`) VALUES ('16085', 'Oraky', '12', '130', '12', '12');

insert into catalog values ('16071', 'Liaskovska', 'Tatarska', '2', '104', null, '1');

insert into catalog values ('16071', 'Grachova', 'Tatarska', '38', null, '14', '1');

insert into catalog values ('16076', 'Mnogolit', 'Derevlianska', '21', null, '24', '6');

insert into catalog values ('16083', 'Bukin', 'Illienka', '6', null, '16', '3');

insert into catalog values ('16058', 'Kyrychenko', 'Kreschatyk', '36', null, '17', '12');

insert into catalog values ('16085', 'Koba', 'Tatarska', '20', '74', null, '12');

insert into catalog values ('94421', 'Liaskovska', 'Tatarska', '2', '104', null, '12');

INSERT INTO `cp`.`worker` (`CardNum`, `Name`, `Post`, `IndexPO`, `Hiredate`, `Salary`, `Commission`) VALUES ('40101', 'Feder', 'Head', '1044', '2000-08-15', '10000.00', '2000.00');

INSERT INTO `cp`.`worker` (`CardNum`, `Name`, `Post`, `IndexPO`, `Hiredate`, `Salary`, `Commission`, `Chief`) VALUES ('20508', 'Guf', 'Dean', '1044', '2010-09-04', '9000.00', '1500.00', '40101');

INSERT INTO `cp`.`worker` (`CardNum`, `Name`, `Post`, `IndexPO`, `Hiredate`, `Salary`, `Commission`, `Chief`) VALUES ('14011', 'Micki', 'Operator', '1044', '2010-05-05', '5000.00', '500.00', '20508');

INSERT INTO `cp`.`worker` (`CardNum`, `Name`, `Post`, `IndexPO`, `Hiredate`, `Salary`, `Commission`, `Chief`) VALUES ('14012', 'Barbie', 'Postman', '1044', '2015-06-08', '6000.00', '500.00', '20508');

ЗАПИТИ ДО БД

1. **Пошук абонента за вулицею**

select \* from Subscriber where street = 'Tatarska';

1. **Пошук працівника за посадою**

select \* from Worker where post = 'postman';

1. **Пошук абонентів виписуючих періодичні видання**

select s.SubSurname, s.Name, s.IndexPO, s.Street, s.House, s.Flat, s.NumSB

from Subscriber s, Catalog c, SignedEdition se

where c.SurnameSub = s.SubSurname and se.SEIndex = c.IndexSE;

1. **Пошук абонентів, які мають абонскриньки**

select s.SubSurname, s.Name, s.IndexPO, s.Street, s.House, s.Flat, s.NumSB

from Subscriber s, SubBox sb

where sb.SBNum = s.NumSB;

1. **Вивести середні період підписки та періодичність видання на рік за умов, що вартість видання(50>Cost<350) та ці виданнявиписують абоненти.**

select avg(se.SESignTime) as SignTime, avg(se.ReleasePerYear) as ReleasePY

from Catalog c, SignedEdition se

where (se.Cost between 50 and 350) and se.SEIndex = c.IndexSE;

1. **Вивести працівника та індекс поштового відділення ня якому вони працюють при виконанні умов зар/плати.**

SELECT w.Name AS 'Name', po.POIndex

FROM postoffice po, worker w

WHERE po.POIndex = w.IndexPO and (w.Salary+w.Commission) > 7500

UNION

SELECT w.Name AS 'Name', po.POIndex

FROM postoffice po, worker w

WHERE po.POIndex = w.IndexPO and (w.Salary+w.Commission) < 5500

ORDER BY Name;

1. **Порахувати кількість видань та середню періодичність видання за умови, зо їх вартість більша за мінімальну вартість видання яке виписує абонент.**

select count(se.SEIndex) as CIndex, avg(ReleasePerYear) as ReleaseAVG

from signededition se where se.Cost > (select min(se.Cost)

from catalog c, signededition se where se.SEIndex = c.IndexSE);

1. **Деяка інформація про працівників та їх керівників, які прийняті на роботу в визначений період часу**

select w1.IndexPO as IndexPO, w1.name as 'Boss', w1.CardNum,

w.Chief, w.Name AS Subord, w.Post, w.CardNum as SubCard

from worker w, worker w1

where w1.CardNum = w.Chief and (w.Hiredate between '1999-10-01' AND '2015-12-31')

order by IndexPO, Boss;

1. **Пошук працівників зарплата+комісія, яких більші ніж половина зарплата+коміся** 'Skalova'

SELECT Name FROM Worker

WHERE Salary + Commission > (SELECT (Salary + Commission) / 2

FROM Worker WHERE Name like 'Skalova');

1. **Вивести Ім’я працівників, які працюють у конкретних ВПЗ**

select w.Name as Name

from worker w

join postoffice po on po.POIndex = w.IndexPO

join postoffice p on p.POIndex = w.IndexPO and p.POIndex in('4107', '4119')

group by Name;

ВИСНОВОК

Створила БД для проведення оптимізації роботи працівників поштового відділення, поліпшення зберігання даних працівників(№ посвідчення, посада, заробітня плата, дата прийому на роботу), для зручності проводення необхідних запитів та пошуки, додавати або видаляти робітників.; , створила зв’язки між сутностями і атрибутів, що мають правильний функціонал та дозволяють користувачам даної БД отримувати усі необхідні дані.

Також за допомогою БД зручніше буде проводити необхідні запити та пошуки, додавати або видаляти робітників.

,