

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ
УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту



Лабораторна робота №3

з дисципліни

«Організація баз даних та знань»

Виконав:

студент групи ШІ-23

Полуліх Ю.Ю.

Викладач:

Поберейко П. Б.

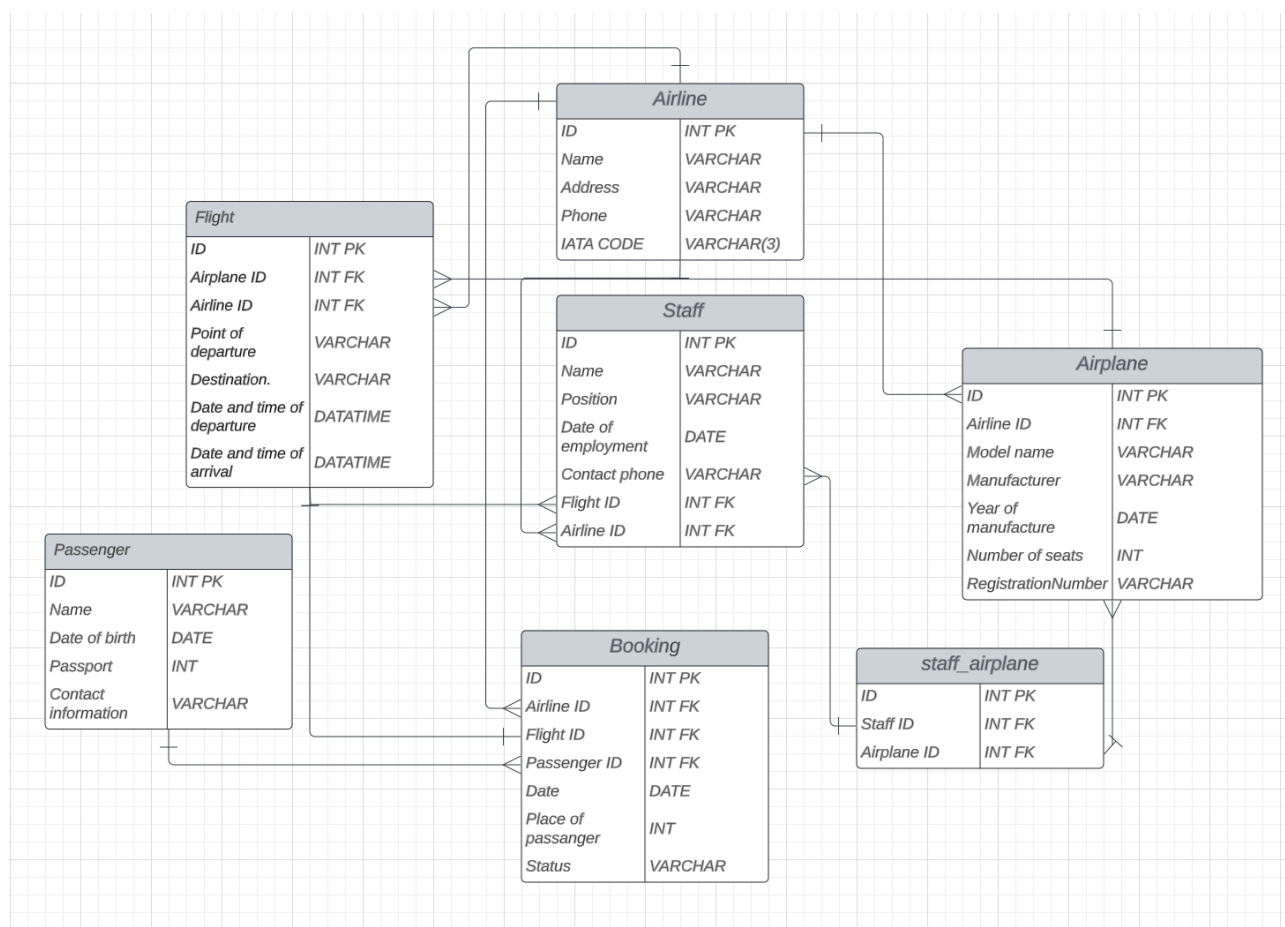
Львів – 2023 р.

Тема: Створення запитів щодо опрацювання даних БД засобами SQL

Мета роботи: Розробити SQL запити для виконання теоретико-множинних операцій реляційної алгебри: об'єднання, перетину, різниці, декартового добутку, проєкції, селекції, натурального з'єднання, умовного з'єднання, а також сортування, фільтрування даних, групування даних, об'єднання даних (JOIN, UNION), статистика, підзапитів (IN, ANY, ALL, EXISTS), обчислювальні стовпці, аналітичних та підсумкових запитів.

Хід роботи

Графічна модель бази даних:



Об'єднання Union:

```
SELECT id,Name FROM Passengers
```

UNION

```
SELECT id,Name FROM Staff;
```

	id	Name
Employee	1	John Doe
	2	Jane Smith
	1	Michael Johnson
	2	Emily Williams

Перетин INNER JOIN:

Passenger:

	Id	Name	DateOfBirth	Passport	ContactInformation
	1	John Doe	1980-05-15	123456789	john@example.com
	2	Jane Smith	1995-10-20	987654321	jane@example.com
	3	David Lee	1988-07-12	456789123	david@example.com

Staff:

	Id	Name	Position	DateOfEmploye...	ContactPhone	FlightId	AirlineId
	1	Michael Johnson	Pilot	2018-01-01	111-222-3333	1	1
	2	Emily Williams	Flight Attendant	2019-02-01	222-333-4444	2	2
	3	David Lee	Cabin Crew	2020-03-01	333-444-5555	2	2

```
SELECT p.Id, p.Name
FROM Passengers p
INNER JOIN Staff s
ON p.Name = s.Name;
```

	id	Name
	3	David Lee

Різниця (EXCEPT або MINUS):

```
SELECT Id, Name
FROM Passengers
WHERE Name NOT IN (
    SELECT Name FROM Staff
);
```

	Id	Name
	1	John Doe
	2	Jane Smith

Є в першій таблиці і немає і другій.

Відбірка (SELECTION):

```
SELECT * FROM Passengers;
```

	Id	Name	DateOfBirth	Passport	ContactInformation
	1	John Doe	1980-05-15	123456789	john@example.com
	2	Jane Smith	1995-10-20	987654321	jane@example.com
	3	David Lee	1988-07-12	456789123	david@example.com

```
SELECT *
FROM Passengers
WHERE DateOfBirth > ,1990-01-01';
```

	Id	Name	DateOfBirth	Passport	ContactInformation
	2	Jane Smith	1995-10-20	987654321	jane@example.com

Проекція (PROJECTION):

```
SELECT Name
FROM Passengers;
```

	Name
	John Doe
	Jane Smith
	David Lee

Об'єднання (SET UNION або UNION ALL):

```
SELECT Name FROM Passengers
UNION ALL
SELECT Name FROM Staff;
```

	Name
	John Doe
	Jane Smith
	David Lee
	Michael Johnson
	Emily Williams
	David Lee

Використовуючи UNION ALL ми об'єднуємо всі рядки таблиць навіть з дублікатами.

Декартовий добуток (CARTESIAN PRODUCT):

```
SELECT p.Name, s.Position FROM Passengers p
CROSS JOIN Staff s;
```

	Name	Position
	David Lee	Pilot
	Jane Smith	Pilot
	John Doe	Pilot
	David Lee	Flight Attendant
	Jane Smith	Flight Attendant
	John Doe	Flight Attendant
	David Lee	Cabin Crew
	Jane Smith	Cabin Crew
	John Doe	Cabin Crew

Умовне з'єднання (WHERE):

```
SELECT *  
FROM Passengers  
WHERE DateOfBirth > '1990-12-31';
```

	Id	Name	DateOfBirth	Passport	ContactInformation
	2	Jane Smith	1995-10-20	987654321	jane@example.com
	NULL	NULL	NULL	NULL	NULL

Об'єднання

(внутрішнє з'єднання) (JOIN):

```
SELECT Passengers.Name, Booking.Date  
FROM Passengers  
INNER JOIN Booking ON Passengers.Id = Booking.PassengerId;
```

	Name	Date
	Jane Smith	2023-10-14

Агрегування даних (GROUP BY):

```
SELECT YEAR(DateOfBirth) AS BirthYear, COUNT(*) AS PassengerCount  
FROM Passengers  
GROUP BY BirthYear  
ORDER BY BirthYear;
```

	BirthYear	PassengerCount
	1980	1
	1988	1
	1995	1

Умова при агрегації (HAVING):

```
SELECT AirlineId, COUNT(*) as FlightCount  
FROM Flights  
GROUP BY AirlineId  
HAVING COUNT(*) > 2;
```

	AirlineId	FlightCount
	1	3

Сортування даних (ORDER BY):

Посортуємо за датою народження:

	Id	Name	DateOfBirth	Passport	ContactInformation
	1	John Doe	1980-05-15	123456789	john@example.com
	3	David Lee	1988-07-12	456789123	david@example.com
	2	Jane Smith	1995-10-20	987654321	jane@example.com

Підзапити (Subqueries):

```
SELECT Name
FROM Passengers
WHERE Id IN (SELECT PassengerId FROM Booking);
```

	Name
	John Doe
	Jane Smith

Висновок: На даному практичному занятті було розглянуто методи використання операцій реляційної алгебри та їх реалізацію на мові SQL. В ході роботи були виконані операції об'єднання, перетину, різниці та декартового добутку для двох таблиць. Також була розглянута проекція, селекція, натуральне та умовне з'єднання таблиць. Додатково, було використано агрегатні функції для обробки результатів вибірки даних з бази даних.

У процесі вивчення також були розглянуті методи сортування та групування при виборі даних з таблиць бази даних. Натомість, найбільший акцент було зроблено на виборі даних зі з'єднаних таблиць бази даних з використанням різних типів з'єднань та складних критеріїв в умові вибірки. Всі запити до бази даних були виконані з використанням директив SELECT та різних видів операцій JOIN.

Це дозволило отримати розуміння основних методів маніпулювання даними в реляційних базах даних за допомогою мови SQL.