

Київський національний університет ім. Тараса Шевченка

Кафедра мережевих та інтернет технологій

Лабораторна робота №2

Дисципліна: Базы даних та інформаційні системи

Тема: Проектування та створення бази даних

Мета: Ознайомитися з принципами моделювання баз даних, виконати опис бізнес-процесу, створити ER-діаграму, реалізувати структуру бази даних у реляційній СУБД та виконати базові SQL-запити для аналізу даних.

Виконала студентка групи МІТ-31

Пась Олександра

Хід виконання

Варіант 15. База даних для страхових полісів

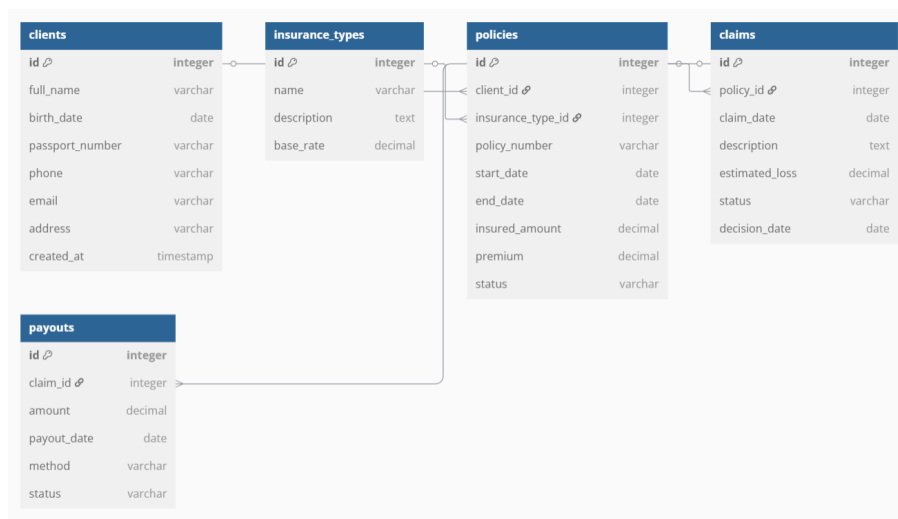
1. Опис бізнес-процесу.

Основні сутності системи:

1. Клієнт - фізична або юридична особа, яка укладає договір страхування
 - Атрибути: ID, ПІБ, контактні дані, паспортні дані, адреса
2. Поліс - договір страхування
 - Атрибути: номер полісу, тип страхування, дата початку/кінця, сума, премія
3. Тип страхування - категорія страхового продукту
 - Атрибути: ID, назва, опис, стандартні умови
4. Страховий випадок - подія, яка призвела до збитків
 - Атрибути: ID, дата, опис, сума збитку, статус розгляду
5. Виплата - компенсація клієнту за страховий випадок
 - Атрибути: ID, сума, дата, спосіб оплати

Взаємодії:

- Клієнт укладає Поліс певного Типу страхування
 - При настанні Страхового випадку клієнт подає заявку
 - Компанія розглядає випадок та ініціює Виплату при підтвердженні
 - Система веде облік активних/закритих полісів та виплат
2. Створення ER-діаграми.



3. Реалізація бази даних та користувачів.

General Columns Advanced Constraints Partitions Parameters Security SQL

Privileges				+
	Grantee	Privileges	Grantor	
	admin ▾	damxrtDw	postgres	
	moderator ▾	raw	postgres	
	postgres ▾	txrdDawm	postgres	
	user ▾	r	postgres	

4. Створення таблиць.

```

-- Таблиця клієнтів
CREATE TABLE clients (
    id SERIAL PRIMARY KEY,
    full_name VARCHAR(100) NOT NULL,
    birth_date DATE,
    passport_number VARCHAR(20) UNIQUE NOT NULL,
    phone VARCHAR(20),
    email VARCHAR(50),
    address TEXT,
    created_at TIMESTAMPTZ DEFAULT CURRENT_TIMESTAMP
);

-- Типи страхування
CREATE TABLE insurance_types (
    id SERIAL PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    description TEXT,
    base_rate DECIMAL(5,2) NOT NULL
);

-- Страхові поліси
CREATE TABLE policies (
    id SERIAL PRIMARY KEY,
    client_id INTEGER REFERENCES clients(id) ON DELETE CASCADE,
    insurance_type_id INTEGER REFERENCES insurance_types(id),
    policy_number VARCHAR(20) UNIQUE NOT NULL,
    start_date DATE NOT NULL,
    end_date DATE NOT NULL,
    insured_amount DECIMAL(12,2) NOT NULL,
    premium DECIMAL(10,2) NOT NULL,
    status VARCHAR(20) DEFAULT 'active',
    CHECK (end_date > start_date)
);
  
```

```
-- Страхові випадки
CREATE TABLE claims (
    id SERIAL PRIMARY KEY,
    policy_id INTEGER REFERENCES policies(id),
    claim_date DATE NOT NULL,
    description TEXT NOT NULL,
    estimated_loss DECIMAL(12,2),
    status VARCHAR(20) DEFAULT 'pending',
    decision_date DATE
);

-- Виплати
CREATE TABLE payouts (
    id SERIAL PRIMARY KEY,
    claim_id INTEGER REFERENCES claims(id),
    amount DECIMAL(12,2) NOT NULL,
    payout_date DATE NOT NULL,
    method VARCHAR(20) NOT NULL,
    status VARCHAR(20) DEFAULT 'completed'
);
```

5. Заповнення таблиць.

```
-- Додавання типів страхування
INSERT INTO insurance_types (name, description, base_rate) VALUES
('Авто-КАСКО', 'Страхування транспортних засобів від пошкоджень та крадіжок', 3.5),
('ОСАГО', 'Обовязкове страхування цивільної відповідальності', 1.2),
('Медичне', 'Страхування медичних витрат', 2.8),
('Майно', 'Страхування нерухомості та домашнього майна', 1.5),
('Життя', 'Страхування життя та від нещасних випадків', 4.0);
```

	id [PK] integer	name character varying (50)	description text	base_rate numeric (5,2)
1	1	Авто-КАСКО	Страхування транспортних засобів від пошкоджень та крадіжок	3.50
2	2	ОСАГО	Обовязкове страхування цивільної відповідальності	1.20
3	3	Медичне	Страхування медичних витрат	2.80
4	4	Майно	Страхування нерухомості та домашнього майна	1.50
5	5	Життя	Страхування життя та від нещасних випадків	4.00

```
-- Додавання клієнтів
INSERT INTO clients (full_name, birth_date, passport_number, phone, email, address) VALUES
('Іванов Петро Сидорович', '1985-05-15', 'KM123456', '+380501234567', 'ivanov@example.com', 'м. Київ, вул. Хрещатик, 1'),
('Петрова Марія Іванівна', '1990-08-22', 'KM654321', '+380671234567', 'petrova@example.com', 'м. Львів, вул. Свободи, 10'),
('Сидоренко Олексій Володимирович', '1978-03-10', 'KM987654', '+380631234567', 'sydorenko@example.com', 'м. Одеса, вул. Дерибасівська, 5');
```

	id [PK] integer	full_name character varying (100)	birth_date date	passport_number character varying (20)	phone character varying (20)	email character varying (50)	address text
1	1	Іванов Петро Сидорович	1985-05-15	KM123456	+380501234567	ivanov@example.com	м. Київ, вул. Хрещатик, 1
2	2	Петрова Марія Іванівна	1990-08-22	KM654321	+380671234567	petrova@example.com	м. Львів, вул. Свободи, 10
3	3	Сидоренко Олексій Володимирович	1978-03-10	KM987654	+380631234567	sydorenko@example.com	м. Одеса, вул. Дерибасівська, 5

```
-- Додавання полісів
INSERT INTO policies (client_id, insurance_type_id, policy_number, start_date, end_date, insured_amount, premium) VALUES
(1, 1, 'AUTO-2023-001', '2023-01-15', '2024-01-14', 250000.00, 8750.00),
(1, 3, 'MED-2023-001', '2023-02-01', '2024-01-31', 100000.00, 2800.00),
(2, 2, 'OSAGO-2023-001', '2023-01-10', '2024-01-09', 50000.00, 600.00),
(3, 4, 'PROP-2023-001', '2023-03-01', '2024-02-28', 500000.00, 7500.00);
```

	id [PK] integer	client_id integer	insurance_type_id integer	policy_number character varying (20)	start_date date	end_date date	insured_amount numeric (12,2)	premium numeric (10,2)	status character varying (20)
1	1	1	1	AUTO-2023-001	2023-01-15	2024-01-14	250000.00	8750.00	active
2	2	1	3	MED-2023-001	2023-02-01	2024-01-31	100000.00	2800.00	active
3	3	2	2	OSAGO-2023-001	2023-01-10	2024-01-09	50000.00	600.00	active
4	4	3	4	PROP-2023-001	2023-03-01	2024-02-28	500000.00	7500.00	active

```
-- Додавання страхових випадків
INSERT INTO claims (policy_id, claim_date, description, estimated_loss, status, decision_date) VALUES
(1, '2023-05-20', 'ДТП, пошкодження переднього бампера', 15000.00, 'approved', '2023-05-25'),
(3, '2023-06-15', 'Пошкодження автомобіля третьої сторони', 8000.00, 'approved', '2023-06-20'),
(1, '2023-07-10', 'Крадіжка автомобіля', 250000.00, 'pending', NULL);
```

	id [PK] integer	policy_id integer	claim_date date	description text	estimated_loss numeric (12,2)	status character varying (20)	decision_date date
1	1	1	2023-05-20	ДТП, пошкодження переднього бампера	15000.00	approved	2023-05-25
2	2	3	2023-06-15	Пошкодження автомобіля третьої сторони	8000.00	approved	2023-06-20
3	3	1	2023-07-10	Крадіжка автомобіля	250000.00	pending	[null]

```
-- Додавання виплат
INSERT INTO payouts (claim_id, amount, payout_date, method, status) VALUES
(1, 15000.00, '2023-05-28', 'bank_transfer', 'completed'),
(2, 8000.00, '2023-06-22', 'bank_transfer', 'completed');
```

	id [PK] integer	claim_id integer	amount numeric (12,2)	payout_date date	method character varying (20)	status character varying (20)
1	1	1	15000.00	2023-05-28	bank_transfer	completed
2	2	2	8000.00	2023-06-22	bank_transfer	completed

6. Виконання SQL-запитів.

```
-- 1. Вибірка всіх клієнтів
SELECT * FROM clients;

-- 2. Сортуювання полісів за сумою страхування
SELECT policy_number, insured_amount, premium
FROM policies
ORDER BY insured_amount DESC;

-- 3. Групування полісів за типом страхування
SELECT it.name, COUNT(p.id) as policy_count, SUM(p.insured_amount) as total_insured
FROM policies p
JOIN insurance_types it ON p.insurance_type_id = it.id
GROUP BY it.name
HAVING COUNT(p.id) > 0;

-- 4. Об'єднання таблиць для аналізу виплат
SELECT c.full_name, p.policy_number, cl.claim_date, cl.estimated_loss, po.amount, po.payout_date
FROM payouts po
JOIN claims cl ON po.claim_id = cl.id
JOIN policies p ON cl.policy_id = p.id
JOIN clients c ON p.client_id = c.id;

-- 5. Агрегатні функції для аналізу виплат
SELECT
    COUNT(*) as total_claims,
    SUM(estimated_loss) as total_estimated_loss,
    AVG(estimated_loss) as average_claim,
    MAX(estimated_loss) as max_claim
FROM claims
WHERE status = 'approved';

-- 6. Пошук клієнтів з найбільшою кількістю полісів
SELECT c.full_name, COUNT(p.id) as policy_count
FROM clients c
LEFT JOIN policies p ON c.id = p.client_id
GROUP BY c.id
ORDER BY policy_count DESC;

-- 7. Аналіз виплат за місяцями
SELECT
    EXTRACT(MONTH FROM payout_date) as month,
    EXTRACT(YEAR FROM payout_date) as year,
    COUNT(*) as payout_count,
    SUM(amount) as total_payouts
FROM payouts
GROUP BY year, month
ORDER BY year, month;
```

Висновок: Під час виконання цієї лабораторної було створено базу даних, користувачів та таблиці. Для кожного таблиці користувачам було надані певні привілеї. Виконано SQL-запити. Ця робота є основою для подальшого вивчення взаємодії з базами даних.