Лекция 11. Транзакции

Задание:

﻿﻿﻿Создайте таблицу для логов

﻿﻿﻿Создайте транзакцию на добавление информации в любую таблицу вашей БД. Одновременно с добавлением в таблицу с логами должна поступать информация о добавлении.

﻿﻿﻿Запустите транзакцию из предыдущего задания, посмотрите на изменения, а затем завершите выполнение транзакции ее откатом.

﻿﻿﻿запустите транзакцию еще раз, а потом подтвердите её выполнение. Изучите измененные данные.

Заблокируйте таблицы на чтение и в конкурирующей консоли запустите последовательно операторы SELECT и

UPDATE. Примечание: блокировка должна разрешить читать информацию в конкурирующей консоли, но не разрешить запись.

-- Удаляем существующую базу данных online\_banking, если таковая существует

DROP DATABASE IF EXISTS online\_banking;

-- Создаем новую базу данных online\_banking

CREATE DATABASE online\_banking;

-- Используем созданную базу данных

USE online\_banking;

-- Удаляем таблицу clients, если она существует

DROP TABLE IF EXISTS clients;

-- Создаем таблицу clients для хранения информации о клиентах

CREATE TABLE IF NOT EXISTS clients (

client\_id INT PRIMARY KEY AUTO\_INCREMENT, -- Уникальный идентификатор клиента

name VARCHAR(200) NOT NULL, -- Имя клиента

last\_name VARCHAR(200) NOT NULL, -- Фамилия клиента

patronymic VARCHAR(200) NOT NULL, -- Отчество клиента

phone\_number VARCHAR(10) NOT NULL, -- Номер телефона клиента

email VARCHAR(200) NOT NULL, -- Email клиента

birthday DATE NOT NULL, -- День рождения клиента

residential\_address VARCHAR(200) NOT NULL -- Адрес проживания клиента

);

-- Удаляем таблицу contract\_types, если она существует

DROP TABLE IF EXISTS contract\_types;

-- Создаем таблицу contract\_types для хранения типов контрактов

CREATE TABLE IF NOT EXISTS contract\_types (

contract\_type\_id INT PRIMARY KEY AUTO\_INCREMENT, -- Уникальный идентификатор типа контракта

contract\_type\_name VARCHAR(200) NOT NULL -- Название типа контракта

);

-- Вставляем значения типов контрактов в таблицу contract\_types

INSERT INTO contract\_types (contract\_type\_name)

VALUES

('Кредитный'),

('Депозитный'),

('Сберегательный'),

('Текущий');

Drop table if exists log\_table;

create table if not exists log\_table (

id int auto\_increment primary key,

log\_date datetime default current\_timestamp,

log\_type set("delete", "update","insert"),

table\_name varchar(50),

client\_id int not null

);

-- Удаляем таблицу accounts, если она существует

DROP TABLE IF EXISTS accounts;

-- Создаем таблицу accounts для хранения информации о счетах клиентов

CREATE TABLE IF NOT EXISTS accounts (

account\_number INT PRIMARY KEY AUTO\_INCREMENT, -- Уникальный номер счета

account\_type\_id INT, -- Идентификатор типа счета

currency VARCHAR(200) NOT NULL, -- Валюта счета

client\_id INT, -- Идентификатор клиента

CONSTRAINT fk\_client\_id FOREIGN KEY (client\_id) REFERENCES clients(client\_id), -- Внешний ключ на таблицу clients

CONSTRAINT fk\_account\_type\_id FOREIGN KEY (account\_type\_id) REFERENCES contract\_types(contract\_type\_id) -- Внешний ключ на таблицу contract\_types

);

-- Добавляем ограничение UNIQUE на поле email в таблице clients

ALTER TABLE clients

ADD CONSTRAINT unique\_email UNIQUE (email);

-- Вставляем данные о клиентах в таблицу clients

INSERT INTO clients (name, last\_name, patronymic, phone\_number, email, birthday, residential\_address)

VALUES

('Иван', 'Иванов', 'Иванович', '1234567890', 'ivan@example.com', '1990-01-01', 'ул. Ленина, д. 1'),

('Петр', 'Петров', 'Петрович', '9876543210', 'petr@example.com', '1985-05-05', 'ул. Пушкина, д. 10'),

('Мария', 'Сидорова', 'Ивановна', '5555555555', 'maria@example.com', '1995-10-10', 'ул. Гагарина, д. 5'),

('Алексей', 'Алексеев', 'Алексеевич', '1111111111', 'alex@example.com', '1988-08-08', 'ул. Кирова, д. 8'),

('Елена', 'Еленова', 'Еленовна', '2222222222', 'elena@example.com', '1992-02-02', 'ул. Сталина, д. 20'),

('Дмитрий', 'Дмитриев', 'Дмитриевич', '3333333333', 'dmitriy@example.com', '1980-12-12', 'ул. Лермонтова, д. 12'),

('Ольга', 'Ольгова', 'Ольговна', '4444444444', 'olga@example.com', '1983-03-03', 'ул. Ленина, д. 30'),

('Николай', 'Николаев', 'Николаевич', '6666666666', 'nikolay@example.com', '1975-07-07', 'ул. Гоголя, д. 7'),

('Анна', 'Иванова', 'Петровна', '7777777777', 'anna@example.com', '1987-04-15', 'ул. Кирова, д. 15'),

('Сергей', 'Сергеев', 'Сергеевич', '8888888888', 'sergey@example.com', '1978-11-25', 'ул. Пушкина, д. 5');

-- Вставляем данные о счетах клиентов в таблицу accounts

INSERT INTO accounts (account\_type\_id, currency, client\_id)

VALUES

(1, 'RUB', 1),

(4, 'USD', 2),

(1, 'EUR', 3),

(1, 'RUB', 4),

(3, 'USD', 5),

(4, 'EUR', 6),

(1, 'RUB', 7),

(3, 'USD', 8),

(4, 'EUR', 9),

(1, 'RUB', 10);

START TRANSACTION;

-- Выбираем клиентов, имена которых начинаются с 'А'

SELECT \* FROM clients

WHERE name LIKE 'А%';

-- Создаем точку сохранения

SAVEPOINT save\_1;

-- Обновляем адреса электронной почты для клиентов, имена которых начинаются с 'А'

UPDATE clients

SET email = CONCAT(email, '\_updated')

WHERE name LIKE 'А%';

-- Вставляем запись в журнал для операции обновления

INSERT INTO log\_table (log\_type, table\_name, client\_id)

SELECT 'update', 'clients', client\_id

FROM clients

WHERE name LIKE 'А%';

-- Фиксируем транзакцию

COMMIT;

-- Блокируем таблицы на чтение

LOCK TABLES clients READ, log\_table READ, accounts READ;

-- Выполняем оператор SELECT

SELECT \* FROM clients WHERE name LIKE 'А%';

-- Разблокируем таблицы

UNLOCK TABLES;