

**Отчет**

по лабораторной работе №4  
по дисциплине "Базы данных"

Выполнили:

Михайлов Андрей Алексеевич

Факультет: ПИиКТ

Группа: Р33212

Преподаватель: Шешуков Д.М.



## Задание

Для выполнения лабораторной работы №4 необходимо:

Реализовать разработанную в рамках лабораторной работы №3 даталогическую модель в реляционной СУБД PostgreSQL.

Заполнить созданные таблицы данными.

Обеспечить целостность данных при помощи средств языка DDL.

В рамках лабораторной работы должны быть разработаны скрипты для создания/удаления требуемых объектов базы данных, заполнения/удаления содержимого созданных таблиц.

## Описание предметной области

Была выбрана предметная область: "Инвестиционный портфель и управление активами". Данная база данных предназначена для учета и анализа инвестиционных портфелей, а также эффективного управления финансовыми активами.

Информационная система предоставляет возможность эффективного управления инвестиционными ресурсами, анализа доходности, рисков и формирования оптимальных стратегий управления активами для инвесторов и финансовых аналитиков.

Задачи, которые решает данная информационная система и ее бизнес процессы:

1. Учет и анализ инвестиционных портфелей (активы, распределение и стоимость)
  - **Инвесторы** предоставляют информацию о себе (физическое/юридическое лицо).
  - Выбирают **инвестиционные инструменты** для включения в свой портфель.
  - Формируется **инвестиционный портфель** из выбранных **финансовых инструментов**, и они становятся текущими **активами** в портфеле.
  - Финансовые инструменты имеют принадлежность к конкретной бирже по **индексу**
  - Идет учет **транзакций** в портфеле
2. Анализ доходности (**отчетность**)
  - Использование показателей **доходности и рисков**
3. Формирование оптимальных стратегий (на основе целей пользователя, рыночных условий и оптимальной доходности)
  - Инвесторы определяют **стратегию управления активами** (консервативный, агрессивный, сбалансированный).
4. Отслеживание дивидендов (дата и доходность)
  - Регистрация **дивидендов** согласно выбранным инвестиционным

инструментам.

5. Мониторинг показателей (эмитента)

- Отслеживание основных **экономических и финансовых показателей компании**, влияющих на инвестиционные решения.

6. Показатели налогов и учет их уплаты

- Учет **налоговых выплат и отчетности**

## DDL скрипты

CREATE:

```
create table investor (  
    id serial primary key,  
    type varchar(50),  
    name varchar(50) NOT NULL,  
    experience varchar(50)  
);  
create table index (  
    id serial primary key,  
    stock_market varchar(50),  
    country varchar(50),  
    value_in_points int  
);  
create table issuer (  
    id serial primary key,  
    name varchar(50) NOT NULL,  
    currency char(3),  
    capitalization varchar(50),  
    p_e numeric,  
    p_s numeric  
);  
create table tool (  
    id serial primary key,  
    name varchar(50) NOT NULL,  
    type varchar(50),  
    ticker varchar(6),  
    price numeric,  
    issuer_id int NOT NULL,  
    index_id int NOT NULL,  
    FOREIGN KEY(issuer_id) REFERENCES issuer(ID),  
    FOREIGN KEY(index_id) REFERENCES index(ID)  
);  
create table strategy (  
    id serial primary key,  
    name varchar(50),  
    description varchar(50),  
    pofitability_forecast varchar(50),  
);  
create table investment_portfolio (  
    id serial primary key,  
    name varchar(50) NOT NULL,  
    type varchar(50),  
    balance int,  
    strategy_id int NOT NULL,  
    investor_id int NOT NULL,  
    FOREIGN KEY(strategy_id) REFERENCES strategy(ID),  
    FOREIGN KEY(investor_id) REFERENCES investor(ID)  
);  
create table active (  
    id serial primary key,
```

```

        quantity int,
        total_price numeric,
        tool_id int NOT NULL,
        investment_portfolio_id int NOT NULL,
        FOREIGN KEY(tool_id) REFERENCES tool(ID),
        FOREIGN KEY(investment_portfolio_id) REFERENCES
investment_portfolio(ID)
);
create table transaction (
    id serial primary key,
    trans_type varchar(50),
    quantity int,
    price int,
    date_t date,
    tool_id int NOT NULL,
    investment_portfolio_id int NOT NULL,
    FOREIGN KEY(tool_id) REFERENCES tool(ID)
    FOREIGN KEY(investment_portfolio_id) REFERENCES
investment_portfolio(ID)
);
create table report (
    id serial primary key,
    period interval,
    income int,
    turnover int,
    deposits int,
    tax varchar(50),
    investment_portfolio_id int NOT NULL,
    FOREIGN KEY(investment_portfolio_id) REFERENCES
investment_portfolio(ID)
);
create table dividends (
    id serial primary key,
    payment_date date,
    tool_id int,
    payment_amount numeric,
    FOREIGN KEY(tool_id) REFERENCES tool(ID)
);
create table tool_strategy (
    tool_id INT NOT NULL,
    strategy_id INT NOT NULL,
    PRIMARY KEY (tool_id, strategy_id),
    FOREIGN KEY (tool_id) REFERENCES tool(ID),
    FOREIGN KEY (strategy_id) REFERENCES strategy(ID)
);

```

## DML скрипты

### INSERT:

```
INSERT INTO investor (type, name, experience)
VALUES ('физлицо', 'Андрей', '3 года'),
       ('юрлицо', 'Банк СПб', '20 лет'),
       ('физлицо', 'Пётр', '1 год'),
       ('физлицо', 'Олег', '5 лет');

INSERT INTO index (stock_market, country, value_in_points)
VALUES ('MOEX', 'Russia', 3300),
       ('RTS', 'Russia', 3000),
       ('SP500', 'Usa', 5000);

INSERT INTO issuer (name, currency, capitalization, p_e, p_s)
VALUES ('Gazprom', 'RUB', 3821000000000, 12.9, 0.9),
       ('Sberbank', 'RUB', 6751000000000, 4.5, 1),
       ('VTB', 'RUB', 650000000000, 1.7, 2),
       ('Ozon', 'RUB', 577000000000, 0, 1.76),
       ('Yandex', 'RUB', 1370000000000, 48.7, 1.67);

INSERT INTO tool (name, type, ticker, price, issuer_id,
index_id)
VALUES ('Gazprom', 'Stock', 'GAZP', 200, 1, 1),
       ('Sberbank', 'Stock', 'SBER', 300, 2, 1),
       ('VTB', 'Stock', 'VTBR', 0.023, 3, 1),
       ('Ozon', 'Stock', 'OZON', 3400, 4, 1),
       ('Yandex', 'Stock', 'YNDX', 3700, 5, 1);

INSERT INTO strategy (name, description, pofitability_forecast)
VALUES ('долгосрочная', 'держатъ ценные бумаги сроком от года',
'50% годовых'),
       ('среднесрочная', 'держатъ ценные бумаги сроком от
месяца', '20% годовых'),
       ('краткосрочная', 'держатъ ценные бумаги сроком от
недели', '30% годовых'),
       ('спекулятивная', 'покупка/продажа бумаг в течении дня',
'не прогнозируется');

INSERT INTO tool_strategy (tool_id, strategy_id)
VALUES (2, 1),
       (2, 2),
       (1, 1),
       (3, 2),
       (3, 3),
       (5, 4),
       (5, 2),
       (4, 1),
       (4, 2),
       (4, 4);
```

```
INSERT INTO investment_portfolio (name, strategy_id,
investor_id, balance)
VALUES ('Большой портфель на год', 1, 1, 200000),
      ('Дневной портфель', 4, 1, 100000),
      ('Портфель банка СПб', 1, 2, 200000000),
      ('Портфель Петра', 2, 3, 50000),
      ('Олегс Портфель', 3, 4, 1000000);

INSERT INTO dividends (payment_date, tool_id, payment_amount)
VALUES ('2023-05-08', 2, 25),
      ('2022-10-07', 1, 51),
      ('2021-07-13', 3, 0.0014);

INSERT INTO transaction (trans_type, quantity, date_t, tool_id,
investment_portfolio_id)
VALUES ('покупка', 20, '2023-01-01', 2, 1),
      ('покупка', 100, '2023-01-02', 5, 5),
      ('покупка', 30, '2023-01-02', 4, 5),
      ('покупка', 1000, '2023-01-03', 1, 3),
      ('покупка', 100000, '2023-01-03', 3, 2),
      ('покупка', 1000000, '2023-01-03', 3, 4),
      ('покупка', 10, '2023-01-04', 5, 1),
      ('покупка', 100, '2023-01-04', 2, 3),
      ('покупка', 100, '2023-01-04', 4, 3),
      ('покупка', 1500000, '2023-01-04', 3, 3);

INSERT INTO report (period, income, deposits, turnover,
investment_portfolio_id)
VALUES ('1 month', 10000, 200000, 400000, 1),
      ('2 months', 0, 100000, 250000, 2),
      ('1 year', 2000000, 200000000, 300000000, 3),
      ('1 month', 1000, 50000, 200000, 4),
      ('2 months', -10000, 1000000, 5000000, 5);
```

## Обеспечение целостности данных с помощью языка DDL

Ограничение CHECK для проверки корректности ввода данных в таблице report:

Проверяет, что при значении депозита = 0, значение оборота тоже должен быть равен 0.

```
ALTER TABLE report  
ADD CONSTRAINT check_zero_turnover_if_zero_deposits  
CHECK (CASE WHEN deposits = 0 THEN turnover = 0 ELSE true END);
```

Правило RULE для фильтрации вводимых значений даты в таблице transaction:

Не позволяет создать транзакцию, если указана дата из будущего по отношению к текущей дате:

```
CREATE OR REPLACE RULE check_transaction_date AS  
ON INSERT TO transaction  
WHERE NEW.date_t > CURRENT_DATE  
DO INSTEAD NOTHING;
```