Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Факультет «Информационных технологий и анализа больших данных»

**ОТЧЕТ**

**по практической работе № 9**

**по теме «Работа с PostgreSQL в Python»**

**по дисциплине Б.1.1.3.4 Системы управления базами данных**

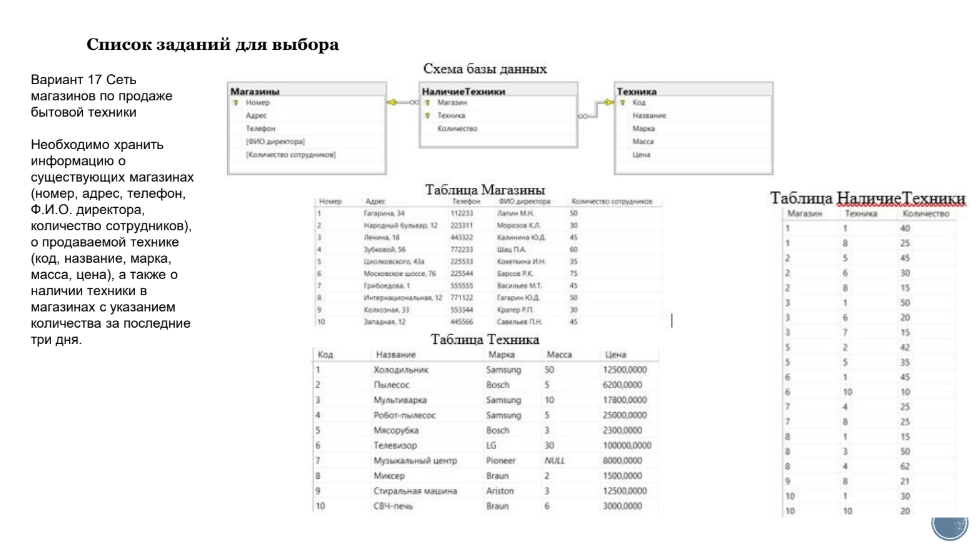
**Вариант 17**

Выполнил: студент группы ИД23-1

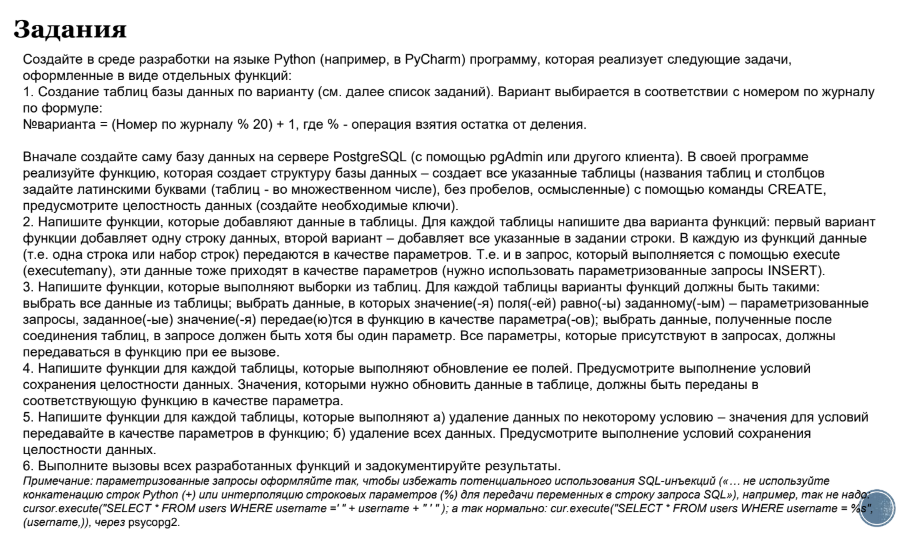
Маслов Александр Николаевич

Проверил преподаватель: Цветков И.Е.

**Москва – 2025**

****

**Задания:**

****

**1.**

def create\_tables(conn):

    cur = conn.cursor()

    cur.execute("""

        CREATE TABLE IF NOT EXISTS Магазины (

            Номер SERIAL PRIMARY KEY,

            Адрес VARCHAR(255),

            Телефон VARCHAR(20),

            ФИО VARCHAR(100),

            Количество\_сотрудников INTEGER

        );

    """)

    cur.execute("""

        CREATE TABLE IF NOT EXISTS Техника (

            Код SERIAL PRIMARY KEY,

            Название VARCHAR(100),

            Марка VARCHAR(50),

            Масса INTEGER,

            Цена NUMERIC

        );

    """)

    cur.execute("""

        CREATE TABLE IF NOT EXISTS НаличиеТехники (

            Магазин INTEGER REFERENCES Магазины(Номер) ON DELETE CASCADE,

            Техника INTEGER REFERENCES Техника(Код) ON DELETE CASCADE,

            Количество INTEGER,

            PRIMARY KEY (Магазин, Техника)

        );

    """)

    conn.commit()

    cur.close()

**2.**

def add\_product(conn, products):

    with conn.cursor() as cur:

        cur.executemany("""

            INSERT INTO Техника (Название, Марка, Масса, Цена)

            VALUES (%s, %s, %s, %s);

        """, products)

    conn.commit()

def add\_all\_products(conn):

    with conn.cursor() as cur:

        cur.execute("""

            INSERT INTO Техника(Название, Марка, Масса, Цена)

            VALUES ('Холодильник','Samsung',50,12500),

                   ('Пылесос','Bosch',5,6200),

                   ('Мультиварка','Samsung',10,17800),

                   ('Робот-пылесос','Samsung',5,25000),

                   ('Мясорубка','Bosch',3,2300),

                   ('Телевизор','LG',30,100000),

                   ('Музыкальный центр','Pioneer',null,8000),

                   ('Миксер','Braun',2,1500),

                   ('Стиральная машина','Ariston',3,12500),

                   ('СВЧ-печь','Braun',6,3000);

        """)

    conn.commit()

def add\_store(conn, stores):

    with conn.cursor() as cur:

        cur.executemany("""

            INSERT INTO Магазины(Адрес, Телефон, ФИО, Количество\_сотрудников)

            VALUES (%s, %s, %s, %s);

        """, stores)

    conn.commit()

def add\_all\_stores(conn):

    with conn.cursor() as cur:

        cur.execute("""

            INSERT INTO Магазины(Адрес, Телефон, ФИО, Количество\_сотрудников)

            VALUES ('Гагарина, 34', '112233', 'Лапин М.К.', 50),

                   ('Народный бульвар, 12', '223311', 'Морозов К.Л.', 30),

                   ('Ленина, 18', '443322', 'Калинина Ю.Д.', 45),

                   ('Зубковой, 56', '772233', 'Шац П.А.', 60),

                   ('Циолковского, 43а', '225533', 'Кокеткина И.Н.', 35),

                   ('Московское шоссе, 76', '225544', 'Барсов Р.К.', 75),

                   ('Грибоедова, 1', '555555', 'Васильев М.Т.', 45),

                   ('Интернациональная, 12', '771122', 'Гагарин Ю.Д.', 50),

                   ('Колхозная, 33', '553344', 'Кратер Р.П.', 30),

                   ('Западная, 12', '445566', 'Савельев П.Н.', 45);

        """)

    conn.commit()

def add\_inventory(conn, inventories):

    with conn.cursor() as cur:

        cur.executemany("""

            INSERT INTO НаличиеТехники(Магазин, Техника, Количество)

            VALUES (%s, %s, %s);

        """, inventories)

    conn.commit()

def add\_all\_inventory(conn):

    with conn.cursor() as cur:

        cur.execute("""

            INSERT INTO НаличиеТехники

            VALUES (1,1,40),

                   (1,8,25),

                   (2,5,45),

                   (2,6,30),

                   (2,8,15),

                   (3,1,50),

                   (3,6,20),

                   (3,7,15),

                   (5,2,42),

                   (5,5,35),

                   (6,1,45),

                   (6,10,10),

                   (7,4,25),

                   (7,8,25),

                   (8,1,15),

                   (8,3,50),

                   (8,4,62),

                   (9,8,21),

                   (10,1,30),

                   (10,10,20);

        """)

    conn.commit()

**3.**

def get\_all(conn, table):

    with conn.cursor() as cur:

        cur.execute("SELECT \* FROM " + table + ";")

        return cur.fetchall()

def get\_one\_by\_name(conn, table, column, target):

    with conn.cursor() as cur:

        cur.execute("SELECT \* FROM "+ table +" WHERE "+ column + " = %s;", (target,))

        return cur.fetchone()

def get\_inventory\_by\_store(conn, id):

    with conn.cursor() as cur:

        cur.execute("""

            SELECT s.Адрес, p.Название, i.Количество

            FROM НаличиеТехники i

            JOIN Магазины s ON i.Магазин = s.Номер

            JOIN Техника p ON i.Техника = p.Код

            WHERE s.Номер = %s;

        """, (id,))

        return cur.fetchall()

**4.**

def update\_product\_price(conn, product\_id, new\_price):

    with conn.cursor() as cur:

        cur.execute("""

            UPDATE Техника

            SET Цена = %s

            WHERE Код = %s;

        """, (new\_price, product\_id))

    conn.commit()

def update\_store\_employees(conn, store\_id, new\_count):

    with conn.cursor() as cur:

        cur.execute("""

            UPDATE Магазины

            SET Количество\_сотрудников = %s

            WHERE Номер = %s;

        """, (new\_count, store\_id))

    conn.commit()

def update\_inventory\_amount(conn, store\_id, product\_id, new\_count):

    with conn.cursor() as cur:

        cur.execute("""

            UPDATE НаличиеТехники

            SET Количество = %s

            WHERE Магазин = %s AND Техника = %s;

        """, (new\_count, store\_id, product\_id))

    conn.commit()

**5.**

def delete\_store\_by\_id(conn, store\_id):

    with conn.cursor() as cur:

        cur.execute("DELETE FROM Магазины WHERE Номер = %s;", (store\_id,))

    conn.commit()

def delete\_all\_stores(conn):

    with conn.cursor() as cur:

        cur.execute("DELETE FROM Магазины;")

    conn.commit()

def delete\_product\_by\_id(conn, product\_id):

    with conn.cursor() as cur:

        cur.execute("DELETE FROM Техника WHERE Код = %s;", (product\_id,))

    conn.commit()

def delete\_all\_products(conn):

    with conn.cursor() as cur:

        cur.execute("DELETE FROM Техника;")

    conn.commit()

def delete\_inventory\_by\_id(conn, store\_id, product\_id):

    with conn.cursor() as cur:

        cur.execute("DELETE FROM НаличиеТехники WHERE Магазин = %s AND Техника = %s;", (store\_id, product\_id))

    conn.commit()

def delete\_all\_inventories(conn):

    with conn.cursor() as cur:

        cur.execute("DELETE FROM НаличиеТехники;")

    conn.commit()

**6.**

conn = psycopg2.connect(dbname="SUBD\_PR9", user="postgres", password="postgres", host="localhost")

create\_tables(conn)

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

add\_all\_stores(conn)

add\_all\_products(conn)

add\_all\_inventory(conn)

add\_store(conn, [('Пушкино, 23', '123456', 'Гагаренко И.И', 24)])

add\_product(conn, [('Мультиварка', 'Sony', 4, 5000), ('Пароварка', 'Bosch', 5, 10000)])

add\_inventory(conn,[(11, 4, 50)])

Изображение выглядит как текст, снимок экрана, число, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

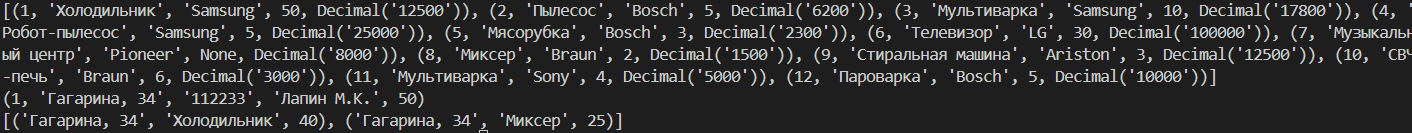
Изображение выглядит как текст, снимок экрана, число, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

print(get\_all(conn, "Техника"))

print(get\_one\_by\_name(conn, "Магазины", "Номер", 1))

print(get\_inventory\_by\_store(conn, 1))



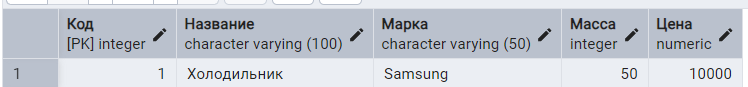
update\_inventory\_amount(conn, 1, 1, 50)

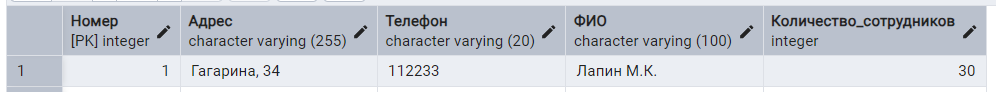
update\_product\_price(conn, 1, 10000)

update\_store\_employees(conn, 1, 30)



2





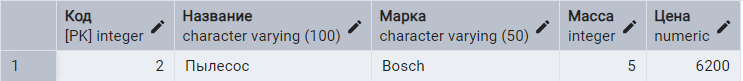
delete\_inventory\_by\_id(conn, 1, 1)

delete\_product\_by\_id(conn, 1)

delete\_store\_by\_id(conn, 1)

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.





delete\_all\_inventories(conn)

delete\_all\_products(conn)

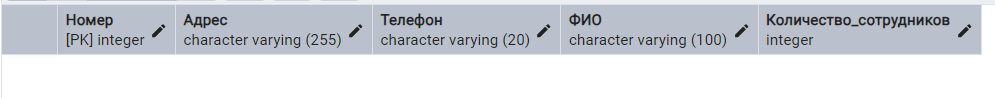
delete\_all\_stores(conn)

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.



conn.close()