

Практическое занятие № 14

Тема: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, работы с БД в IDE PyCharm Community.

Постановка задачи. Приложение **ТОРГОВАЯ ФИРМА** для автоматизированного контроля продаж товаров торговой фирмы. БД должна содержать таблицу Продажа товаров со следующей структурой записи: Дата продажи, Товар, Сумма, Скидка, Филиал, Менеджер.

Тип алгоритма:циклический

Текст программы:

```
import sqlite3
from sqlite3 import Error
from datetime import datetime

def create_connection(db_file):
    conn = None
    try:
        conn = sqlite3.connect(db_file)
        return conn
    except Error as e:
        print(e)
    return conn

def create_table(conn):
    create_table_sql = """
```

```
CREATE TABLE IF NOT EXISTS sales (  
    id INTEGER PRIMARY KEY,  
    sale_date TEXT NOT NULL,  
    product TEXT NOT NULL,  
    amount REAL NOT NULL,  
    discount REAL,  
    branch TEXT NOT NULL,  
    manager TEXT NOT NULL  
);  
"""
```

```
try:  
    c = conn.cursor()  
    c.execute(create_table_sql)  
except Error as e:  
    print(e)
```

```
def add_sale(conn, sale):  
    sql = '''INSERT INTO sales(sale_date, product, amount, discount, branch, manager)  
        VALUES(?,?,?,?,?,?)'''  
    try:  
        cur = conn.cursor()  
        cur.execute(sql, sale)  
        conn.commit()  
        print("Запись добавлена успешно")  
    except Error as e:  
        print(e)
```

```
def search_sales(conn, column, value):  
    sql = f"SELECT * FROM sales WHERE {column}=?"  
    try:  
        cur = conn.cursor()  
        cur.execute(sql, (value,))  
        rows = cur.fetchall()  
        for row in rows:  
            print(row)
```

```
except Error as e:
    print(e)

def delete_sale(conn, sale_id):
    sql = 'DELETE FROM sales WHERE id=?'
    try:
        cur = conn.cursor()
        cur.execute(sql, (sale_id,))
        conn.commit()
        print("Запись удалена успешно")
    except Error as e:
        print(e)

def update_sale(conn, sale_id, column, value):
    sql = f'UPDATE sales SET {column} = ? WHERE id = ?'
    try:
        cur = conn.cursor()
        cur.execute(sql, (value, sale_id))
        conn.commit()
        print("Запись обновлена успешно")
    except Error as e:
        print(e)

def main():
    database = "sales.db"
    conn = create_connection(database)
    if conn is not None:
        create_table(conn)
    else:
        print("Ошибка! Невозможно создать соединение с базой данных.")
    sales = [
        ('2024-05-01', 'Товар1', 100.0, 10.0, 'Филиал1', 'Менеджер1'),
        ('2024-05-02', 'Товар2', 200.0, 20.0, 'Филиал2', 'Менеджер2'),
        ('2024-05-03', 'Товар3', 300.0, 30.0, 'Филиал3', 'Менеджер3'),
```

```

        ('2024-05-04', 'Товар4', 400.0, 40.0, 'Филиал4', 'Менеджер4'),
        ('2024-05-05', 'Товар5', 500.0, 50.0, 'Филиал5', 'Менеджер5'),
        ('2024-05-06', 'Товар6', 600.0, 60.0, 'Филиал6', 'Менеджер6'),
        ('2024-05-07', 'Товар7', 700.0, 70.0, 'Филиал7', 'Менеджер7'),
        ('2024-05-08', 'Товар8', 800.0, 80.0, 'Филиал8', 'Менеджер8'),
        ('2024-05-09', 'Товар9', 900.0, 90.0, 'Филиал9', 'Менеджер9'),
        ('2024-05-10', 'Товар10', 1000.0, 100.0, 'Филиал10', 'Менеджер10')
    ]

    for sale in sales:
        add_sale(conn, sale)

    print("Поиск записей о продаже:")
    search_sales(conn, 'product', 'Товар5')

    print("Удаление записи о продаже:")
    delete_sale(conn, 5)

    print("Редактирование записи о продаже:")
    update_sale(conn, 3, 'amount', 350.0)

    conn.close()

if __name__ == '__main__':
    main()

```

Протокол работы программы:

Process finished with exit code 0

Вывод: В процессе выполнения закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ с БД в IDE PyCharm Community. Были использованы языковые конструкции `import`, `for`

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на [GitHub](#)