

**Практическое задание №15**

**Тема:** составление программ для работы с базами данных в IDE PyCharm Community.

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

**Постановка задачи:** Приложение ЗАКАЗЫ ТОВАРОВ для автоматизированного контроля заказов торговой фирмы. Таблица Заказы должна содержать информацию о товарах со следующей структурой записи: Код товара, Наименование товара, Заказчик (наименование организации, возможны повторяющиеся значения), Дата заказа, Срок исполнения (от 1 до 10 дней), Стоимость заказа.

**Тип алгоритма:** линейный

**Текст программы:**

1)

```
import sqlite3 as sq

order = [
    (1, 'Стол письменный', 'ООО «Мебель»', '18.08.2023', 7, 281000),
    (2, 'Стул', 'ООО «Офис»', '19.08.2023', 5, 30000),
    (3, 'Шкаф', 'ООО «Дом»', '20.08.2023', 10, 50000),
    (4, 'Кресло', 'ООО «Комфорт»', '21.08.2023', 7, 45000),
    (5, 'Компьютерный стол', 'ООО «Техника»', '22.08.2023', 8,
60000),
    (6, 'Офисная тумба', 'ООО «Стиль»', '23.08.2023', 10, 35000),
    (7, 'Стеллаж', 'ООО «Офис»', '24.08.2023', 3, 75000),
    (8, 'Принтер', 'ООО «Печать»', '25.08.2023', 1, 20000),
    (9, 'МФУ', 'ООО «Сервис»', '26.08.2023', 3, 40000),
    (10, 'Ноутбук', 'ООО «Техника»', '27.08.2023', 5, 80000)
]

with sq.connect('orders.db') as con:
    cur = con.cursor()
    cur.execute("DROP TABLE IF EXISTS users")
    cur.execute("""CREATE TABLE IF NOT EXISTS users (
        id INTEGER PRIMARY KEY,
        name TEXT NOT NULL,
        customer TEXT,
        date DATE,
        deadline INTEGER VARCHAR(10),
        cost INTEGER
    )""")

with sq.connect('orders.db') as con:
```

```
cur = con.cursor()
cur.executemany("INSERT INTO users VALUES (?, ?, ?, ?, ?, ?)",
order)
```

#3 операции поиска

```
def search1():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute("SELECT * FROM users WHERE cost>50000")
        result = cur.fetchall()
        print(result)
```

```
def search2():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute("SELECT * FROM users WHERE id=5")
        result = cur.fetchall()
        print(result)
```

```
def search3():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute("SELECT * FROM users WHERE deadline>5")
        result = cur.fetchall()
        print(result)
```

#3 операции удаления

```
def delete1():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute('DELETE FROM users WHERE id=1')
        cur.execute("SELECT * FROM users")
        result = cur.fetchall()
        print(result)
```

```
def delete2():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute('DELETE FROM users WHERE cost<70000')
        cur.execute("SELECT * FROM users")
        result = cur.fetchall()
        print(result)
```

```
def delete3():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute('DELETE FROM users WHERE deadline<6')
        cur.execute("SELECT * FROM users")
```

```

        result = cur.fetchall()
        print(result)

def update1():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute('UPDATE users SET deadline=1 WHERE cost<50000')
        cur.execute("SELECT * FROM users")
        result = cur.fetchall()
        print(result)

def update2():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute('UPDATE users SET cost=1000000 WHERE id=5')
        cur.execute("SELECT * FROM users")
        result = cur.fetchall()
        print(result)

def update3():
    with sq.connect('orders.db') as con:
        cur = con.cursor()
        cur.execute('UPDATE users SET deadline=10 WHERE cost>50000')
        cur.execute("SELECT * FROM users")
        result = cur.fetchall()
        print(result)

```

## Протокол работы программы:

### 1) search1

/home/student/Документы/uc27klipan/bin/python

/home/student/Документы/uc27klipan/pz/pz 15/15.py

[(1, 'Стол письменный', 'ООО «Мебель»', '18.08.2023', 7, 281000), (5, 'Компьютерный стол', 'ООО «Техника»', '22.08.2023', 8, 60000), (7, 'Стеллаж', 'ООО «Офис»', '24.08.2023', 3, 75000), (10, 'Ноутбук', 'ООО «Техника»', '27.08.2023', 5, 80000)]

Process finished with exit code 0

### 2) delete1

```
/home/student/Документы/uc27klipan/bin/python
/home/student/Документы/uc27klipan/pz/pz 15/15.py
[(2, 'Стул', 'ООО «Офис»', '19.08.2023', 5, 30000), (3, 'Шкаф', 'ООО «Дом»', '20.08.2023',
10, 50000), (4, 'Кресло', 'ООО «Комфорт»', '21.08.2023', 7, 45000), (5, 'Компьютерный
стол', 'ООО «Техника»', '22.08.2023', 8, 60000), (6, 'Офисная тумба', 'ООО «Стиль»',
'23.08.2023', 10, 35000), (7, 'Стеллаж', 'ООО «Офис»', '24.08.2023', 3, 75000), (8,
'Принтер', 'ООО «Печать»', '25.08.2023', 1, 20000), (9, 'МФУ', 'ООО «Сервис»',
'26.08.2023', 3, 40000), (10, 'Ноутбук', 'ООО «Техника»', '27.08.2023', 5, 80000)]
```

Process finished with exit code 0

### 3) update1

```
/home/student/Документы/uc27klipan/bin/python
/home/student/Документы/uc27klipan/pz/pz 15/15.py
[(1, 'Стол письменный', 'ООО «Мебель»', '18.08.2023', 7, 281000), (2, 'Стул', 'ООО
«Офис»', '19.08.2023', 1, 30000), (3, 'Шкаф', 'ООО «Дом»', '20.08.2023', 10, 50000), (4,
'Кресло', 'ООО «Комфорт»', '21.08.2023', 1, 45000), (5, 'Компьютерный стол', 'ООО
«Техника»', '22.08.2023', 8, 60000), (6, 'Офисная тумба', 'ООО «Стиль»', '23.08.2023', 1,
35000), (7, 'Стеллаж', 'ООО «Офис»', '24.08.2023', 3, 75000), (8, 'Принтер', 'ООО
«Печать»', '25.08.2023', 1, 20000), (9, 'МФУ', 'ООО «Сервис»', '26.08.2023', 1, 40000), (10,
'Ноутбук', 'ООО «Техника»', '27.08.2023', 5, 80000)]
```

Process finished with exit code 0

### Вывод:

Мы закрепили усвоенные знания, понятия, алгоритмы, основные принципы составления программ, работу с БД в IDE PyCharm Community.