

## Практическое занятие № 15

**Тема:** Составление программ для работы с БД в IDE PyCharm Community.

**Цель:** Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ и работы с БД в IDE PyCharm Community

### Постановка задачи.

Написать БД учёта товаров на складе, таблица Товарный запас должна содержать – код товара, торговая марка, тип, цена, кол-во на складе, минимальный запас

**Текст программы:**

1.

#Написать БД учёта товаров на складе, таблица Товарный запас должна содержать –  
#код товара, торговая марка, тип, цена, кол-во на складе, минимальный запас

```
import sqlite3
import add
import brandsearch
import codesearch
import minchange
import minsearch
import pricechange
import quantitychange
import output
import deletecode

# Соединяемся с БД
conn = sqlite3.connect('товарный_запас.db')
c = conn.cursor()

# Создаем таблицу "Товары"
c.execute('''
    CREATE TABLE IF NOT EXISTS Stock (
        code INTEGER PRIMARY KEY AUTOINCREMENT,
        brand TEXT NOT NULL,
        stype TEXT NOT NULL,
        price REAL NOT NULL,
        quantity INTEGER NOT NULL,
        minquantity INTEGER NOT NULL
    )
''')

def find(c):
    print('Выберите действие:')
    print('1. Поиск по марке')
    print('2. Поиск по минимальному кол-ву')
    print('3. Поиск по коду')
    print('4. Вернуться')
    choice = int(input('Введите номер действия: '))

    if choice == 1:
        brandsearch.brandsearch(c)
    elif choice == 2:
```

```

        minsearch.minsearch(c)
    elif choice == 3:
        codesearch.codesearch(c)
    elif choice == 4:
        return
    else:
        print('Неверный номер действия')

def change(c):
    print('Выберите действие:')
    print('1. Изменить цену')
    print('2. Измененить минимальное кол-во товаров')
    print('3. Изменить кол-во на складе')
    print('4. Вернуться')
    choice = int(input('Введите номер действия: '))

    if choice == 1:
        pricechange.pricechange(c)
    elif choice == 2:
        minchange.quantitychange(c)
    elif choice == 3:
        quantitychange.quantitychange(c)
    elif choice == 4:
        return
    else:
        print('Неверный номер действия')

while True:
    print('Выберите действие:')
    print('1. Добавить товар')
    print('2. Найти товары')
    print('3. Редактировать товары')
    print('4. Удалить товары')
    print('5. Вывести БД')
    print('6. Выйти из программы')
    choice = int(input('Введите номер действия: '))

    if choice == 1:
        add.inputdata(c)
    elif choice == 2:
        find(c)
    elif choice == 3:
        change(c)
    elif choice == 4:
        print('Введите код товара на удаление')
        code = input()
        deletetcode.delete_entry(c,code)
    elif choice == 5:
        output.output(c)
    elif choice == 6:
        break
    else:

```

```

        print('Неверный номер действия')

# Закрываем соединение с БД
conn.commit()
conn.close()

```

```

import sqlite3

def inputdata(c):
    mark = input('Введите торговую марку: ')
    type = input('Введите тип: ')
    price = float(input('Введите цену: '))
    quantity = int(input('Введите количество на складе: '))
    minquantity = int(input('Введите минимальный запас: '))

    c.execute('''
        INSERT INTO Stock (brand, stype, price, quantity, minquantity)
        VALUES (?, ?, ?, ?, ?)
    ''', (mark, type, price, quantity, minquantity))
    code = c.lastrowid
    print(f'Успешно добавленно код: {code} марка: {mark} тип товара: {type} цена: {price} кол-во на складе: {quantity} минимально допустимое кол-во: {minquantity}')

```

```

import sqlite3

def brandsearch(c):
    mark = input('Введите торговую марку для поиска: ')

    c.execute('''
        SELECT * FROM Stock
        WHERE brand = ?
    ''', (mark,))

    print('Результаты поиска:')
    for row in c:
        print(row)

```

```

import sqlite3

def codesearch(c):
    code = input('Введите код для поиска: ')

    c.execute('''
        SELECT * FROM Stock
        WHERE code = ?
    ''', (code,))

    print('Результаты поиска:')
    for row in c:
        print(row)

```

```

import sqlite3

```

```

conn = sqlite3.connect('товарный_запас.db')

def delete_entry(c, code):
    c.execute("DELETE from Stock WHERE code=?", (code,))
    conn.commit()
    print("Успешно удалено")

```

```

import sqlite3

def quantitychange(c):
    code = int(input('Введите код товара для редактирования: '))
    newminquantity = int(input('Введите новое значение минимального запаса: '))

    c.execute('''
        UPDATE Stock
        SET minquantity = ?
        WHERE code = ?
    ''', (newminquantity, code))

```

```

import sqlite3

def minsearch(c):
    c.execute('''
        SELECT * FROM Stock
        WHERE quantity < minquantity
    ''')

    print('Результаты поиска:')
    for row in c:
        print(row)

```

```

import sqlite3

def output(c):
    c.execute('''
        SELECT * FROM Stock
    ''')

    print('Содержимое таблицы "Товары":')
    for row in c:
        print(row)

```

```

import sqlite3

def pricechange(c):
    code = int(input('Введите код товара для редактирования: '))
    newprice = float(input('Введите новую цену: '))

    c.execute('''
        UPDATE Stock
        SET price = ?
    ''', (newprice, code))

```

```
WHERE code = ?  
'', (newprice, code))
```

```
import sqlite3  
  
def quantitychange(c):  
    code = int(input('Введите код товара для редактирования: '))  
    newquantity = int(input('Введите новое количество на складе: '))  
  
    c.execute('''  
        UPDATE Stock  
        SET quantity = ?  
        WHERE code = ?  
    ''', (newquantity, code))
```

Вывод: Были закреплены усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ и работы с БД в IDE PyCharm Community