

Практическое занятие № 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 ОПТОВАЯ БАЗА для автоматизированного контроля движения товаров на оптовой базе. Таблица Товары должна содержать следующие данные: Код товара, Наименование товара, Наименование магазина, Заявки магазина, Количество товара на складе, Единицы измерения, Оптовая цена.

Тип алгоритма: линейный

Текст программы:

```

"""Приложение ОПТОВАЯ БАЗА для автоматизированного контроля движения
товаров на оптовой базе. Таблица Товары должна содержать следующие данные:
Код
товара, Наименование товара, Наименование магазина, Заявки магазина,
Количество
товара на складе, Единицы измерения, Оптовая цена.
"""
import random
import sqlite3

product_names = ['Яблоко', 'Банан',
                  'Груша', 'Мандарин', 'Апельсин', 'Малина',
                  'Клубника', 'Помидор', 'Огурец', 'Баклажан']

magaz_names = ['Яблоко', 'Банан',
               'Груша', 'Мандарин', 'Апельсин', 'Малина',
               'Клубника', 'Помидор', 'Огурец', 'Баклажан'][::-1]

ed_izm = ['грамм (г)', 'килограмм (кг)', 'центнер (ц)', 'тонна (т)']

def create_database(connect, cursor):
    try:
        cursor.execute("""CREATE TABLE IF NOT EXISTS products (
                           id_p INTEGER PRIMARY KEY,
                           name_product VARCHAR(50),
                           name_magaz VARCHAR(200),
                           zayavki_magaz INT,
                           ostatok INT,
                           ed_izm VARCHAR(20),
                           cost INT);""")
        connect.commit()
    except sqlite3.Error as e:
        print("Ошибка при создании таблицы:", e)

def create_zap(connect, cursor):
    for i in range(10):
        cursor.execute(f"""INSERT INTO products (name_product, name_magaz,
zayavki_magaz, ostatok, ed_izm, cost)
VALUES('{product_names[i]}',
'{magaz_names[i]}', {random.randint(0, 2000)}, {random.randint(0,
5000)},

```

```

        '{random.choice(ed_izm)}', {random.randint(0, 200)}) """)
    connect.commit()

def view_zap(cursor):
    all_product = list(map(lambda x: x[0], cursor.execute("""SELECT DISTINCT
name_product
FROM products;""").fetchall()))
    all_magaz = list(map(lambda x: x[0], cursor.execute("""SELECT DISTINCT
name_magaz
FROM products;""").fetchall()))
    all_ed_izm = list(map(lambda x: x[0], cursor.execute("""SELECT DISTINCT
ed_izm
FROM products;""").fetchall()))
    print(all_product)
    print(all_magaz)
    print(all_ed_izm)

def update(connect, cursor):
    for _ in range(3):
        random_id = random.randint(1, 11)
        random_ed_izm = random.choice(ed_izm)
        cursor.execute(f"""UPDATE products SET ed_izm='{random_ed_izm}' WHERE
id_p={random_id}""")
        print(f"У записи №{random_id} единицы измерения изменены на
{random_ed_izm}")
    connect.commit()

def delete(connect, cursor):
    random_magaz = random.choice(magaz_names)
    random_product = random.choice(product_names)
    random_ed = random.choice(ed_izm)
    cursor.execute(f"""DELETE FROM products WHERE
name_product='{random_product}'""")
    cursor.execute(f"""DELETE FROM products WHERE
name_magaz='{random_magaz}'""")
    cursor.execute(f"""DELETE FROM products WHERE ed_izm='{random_ed}'""")
    print(f'Удалён магазин {random_magaz}')
    print(f'Удалён товар {random_product}')
    print(f'Удалены товары с единицей измерения {random_ed}')
    connect.commit()

def main():
    connect = sqlite3.connect('base.db')
    cursor = connect.cursor()
    create_database(connect, cursor)
    create_zap(connect, cursor)
    view_zap(cursor)

    # update(connect, cursor)
    # delete(connect, cursor)

if __name__ == '__main__':
    main()

```

Протокол работы программы:

['Яблоко', 'Банан', 'Груша', 'Мандарин', 'Апельсин', 'Малина', 'Клубника', 'Помидор', 'Огурец', 'Баклажан']

['Баклажан', 'Огурец', 'Помидор', 'Клубника', 'Малина', 'Апельсин', 'Мандарин', 'Груша', 'Банан', 'Яблоко']

['тонна (т)', 'центнер (ц)', 'килограмм (кг)', 'грамм (г)']

У записи №6 единицы измерения изменены на килограмм (кг)

У записи №2 единицы измерения изменены на центнер (ц)

У записи №5 единицы измерения изменены на грамм (г)

Удалён магазин Банан

Удалён товар Банан

Удалены товары с единицей измерения центнер (ц)

Process finished with exit code 0

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