

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

Цель: закрепить усвоенные знания, понятия, база данных основные принципы составления программ, приобрести навыки составления программ с использованием баз данных в IDE PyCharm Community.

Постановка задачи.

Приложение для туристического агентства ТУР. Таблица Турист должна следующую информацию о клиентах турфирмы: Код клиента, Клиент (Фамилия), Телефон, Название страны, Регион, Продолжительность поездки, Стоимость путевки.

Текст программы:

```
import sqlite3
import os

def db_connect():
    db_name = 'tour.db'
    if not os.path.exists(db_name):
        conn = sqlite3.connect(db_name)
        cursor = conn.cursor()
        cursor.execute("""CREATE TABLE IF NOT EXISTS Tourist
                        (Client_code INTEGER PRIMARY KEY,
                         Client_name TEXT NOT NULL,
                         Phone TEXT NOT NULL,
                         Country_name TEXT NOT NULL,
                         Region TEXT NOT NULL,
                         Duration INTEGER NOT NULL,
                         Cost INTEGER NOT NULL)""")
        conn.commit()
        conn.close()
    conn = sqlite3.connect(db_name)
    return conn

def db_add(conn):
    cursor = conn.cursor()
    client_code = input('Введите код клиента: ')
    client_name = input('Введите фамилию клиента: ')
    phone = input('Введите телефон клиента: ')
    country_name = input('Введите название страны: ')
    region = input('Введите регион: ')
    duration = input('Введите продолжительность поездки: ')
    cost = input('Введите стоимость путевки: ')
```

```

    cursor.execute("INSERT INTO Tourist (Client_code, Client_name, Phone,
Country_name, Region, Duration, Cost)
                VALUES (?, ?, ?, ?, ?, ?, ?)", (client_code, client_name, phone,
country_name, region, duration, cost))
    conn.commit()
    cursor.close()

def db_search(conn):
    cursor = conn.cursor()
    search_type = input('Введите тип поиска (код клиента, фамилия, телефон,
название страны, регион, продолжительность, стоимость): ')
    search_value = input('Введите значение для поиска: ')
    if search_type == 'код клиента':
        cursor.execute("SELECT * FROM Tourist WHERE Client_code = ?",
(search_value,))
    elif search_type == 'фамилия':
        cursor.execute("SELECT * FROM Tourist WHERE Client_name = ?",
(search_value,))
    elif search_type == 'телефон':
        cursor.execute("SELECT * FROM Tourist WHERE Phone = ?",
(search_value,))
    elif search_type == 'название страны':
        cursor.execute("SELECT * FROM Tourist WHERE Country_name = ?",
(search_value,))
    elif search_type == 'регион':
        cursor.execute("SELECT * FROM Tourist WHERE Region = ?",
(search_value,))
    elif search_type == 'продолжительность':
        cursor.execute("SELECT * FROM Tourist WHERE Duration = ?",
(search_value,))
    elif search_type == 'стоимость':
        cursor.execute("SELECT * FROM Tourist WHERE Cost = ?",
(search_value,))
    else:
        print('Неверный тип поиска')
        cursor.close()
    return
result = cursor.fetchall()
for row in result:
    print(row)
cursor.close()

```

```

def db_delete(conn):
    cursor = conn.cursor()
    delete_type = input('Введите тип удаления (код клиента, фамилия,
телефон, название страны, регион, продолжительность, стоимость): ')
    delete_value = input('Введите значение для удаления: ')
    if delete_type == 'код клиента':
        cursor.execute("DELETE FROM Tourist WHERE Client_code = ?",
(delete_value,))
    elif delete_type == 'фамилия':
        cursor.execute("DELETE FROM Tourist WHERE Client_name = ?",
(delete_value,))
    elif delete_type == 'телефон':
        cursor.execute("DELETE FROM Tourist WHERE Phone = ?",
(delete_value,))
    elif delete_type == 'название страны':
        cursor.execute("DELETE FROM Tourist WHERE Country_name = ?",
(delete_value,))
    elif delete_type == 'регион':
        cursor.execute("DELETE FROM Tourist WHERE Region = ?",
(delete_value,))
    elif delete_type == 'продолжительность':
        cursor.execute("DELETE FROM Tourist WHERE Duration = ?",
(delete_value,))
    elif delete_type == 'стоимость':
        cursor.execute("DELETE FROM Tourist WHERE Cost = ?",
(delete_value,))
    else:
        print('Неверный тип удаления')
        cursor.close()
        return
    conn.commit()
    print('Данные удалены')
    cursor.close()

def db_edit(conn):
    cursor = conn.cursor()
    edit_type = input('Введите тип редактирования (код клиента, фамилия,
телефон, название страны, регион, продолжительность, стоимость): ')
    edit_value = input('Введите значение для редактирования: ')
    if edit_type == 'код клиента':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Client_code = ? WHERE

```

```

Client_code = ?", (new_value, edit_value))
    elif edit_type == 'фамилия':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Client_name = ? WHERE
Client_name = ?", (new_value, edit_value))
    elif edit_type == 'телефон':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Phone = ? WHERE Phone = ?",
(new_value, edit_value))
    elif edit_type == 'название страны':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Country_name = ? WHERE
Country_name = ?", (new_value, edit_value))
    elif edit_type == 'регион':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Region = ? WHERE Region = ?",
(new_value, edit_value))
    elif edit_type == 'продолжительность':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Duration = ? WHERE Duration =
?", (new_value, edit_value))
    elif edit_type == 'стоимость':
        new_value = input('Введите новое значение: ')
        cursor.execute("UPDATE Tourist SET Cost = ? WHERE Cost = ?",
(new_value, edit_value))
    else:
        print('Неверный тип редактирования')
        cursor.close()
        return
conn.commit()
print('Данные отредактированы')
cursor.close()

conn = db_connect()
while True:
    print('Выберите действие: ')
    print('1. Добавить данные')
    print('2. Найти данные')
    print('3. Удалить данные')
    print('4. Отредактировать данные')
    print('5. Выйти из программы')
    choice = input()

```

```
if choice == '1':
    db_add(conn)
elif choice == '2':
    db_search(conn)
elif choice == '3':
    db_delete(conn)
elif choice == '4':
    db_edit(conn)
elif choice == '5':
    conn.close()
    break
else:
    print('Неверный выбор')
```

Протокол работы программы:

Выберите действие:

- 1. Добавить данные**
- 2. Найти данные**
- 3. Удалить данные**
- 4. Отредактировать данные**
- 5. Выйти из программы**

1

Введите код клиента: 13

Введите фамилию клиента: Смагин

Введите телефон клиента: 89287598591

Введите название страны: Казахстан

Введите регион: 2

Введите продолжительность поездки: 1200

Введите стоимость путевки: 15700

2

Введите тип поиска (код клиента, фамилия, телефон, название страны, регион, продолжительность, стоимость): код клиента

Введите значение для поиска: 13

(13, 'Смагин', '89287598591', 'Казахстан', '2', 1200, 15700)

3

Введите тип удаления (код клиента, фамилия, телефон, название страны, регион, продолжительность, стоимость): код клиента

Введите значение для удаления: 13

Данные удалены

4

Введите тип редактирования (код клиента, фамилия, телефон, название страны, регион, продолжительность, стоимость): код клиента

Введите значение для редактирования: 6

Введите новое значение: 14

Данные отредактированы

5

Process finished with exit code 0

Вывод:закрепила усвоенные знания, понятия, база данных основные принципы составления программ, приобрела навыки составление программ с использованием баз данных в IDE PyCharm Community.