**PhoneBook Manipulation - implement it with Stream API**

Airone mobile services needs to store their customer details in the company portal. The details are customer's first and last name, phone number, and email address. Help the company develop an application to maintain the details of their customer systematically.   
  
You are provided with a class Contact with the following attributes as private.

String firstName

String lastName

long  phoneNumber

String emailId

A 4 argument constructor and appropriate setters and getters to store and retrieve the details are also provided.

Create a class **PhoneBook** with a private attribute

List<Contact> phoneBook = new ArrayList<Contact>();

Write the getters and setters.

Write the following methods in the PhoneBook class.

public void addContact(Contact contactObj) – This method should add the contact object to the phoneBook list.

public List<Contact> viewAllContacts() – This method should return the list of all contacts available.

public Contact viewContactGivenPhone(long phoneNumber) -  This method should return the contact details which has the phoneNumber given as parameter.

public boolean removeContact(long phoneNumber) -  This method should remove the contact details which has the phoneNumber given as parameter.  If removed return true.  Else if the phone number is not available return false.

**Write a class Main with the main method.**  Create the menu as shown in the Sample Input and Output and invoke the corresponding methods as per the choice provided.

**Sample Input and Output 1:**

Menu

1.Add Contact

2.Display all contacts {{apply Stream API}}

3.Search contact by phone {{apply Stream API}}

4. Display all First Names {{apply Stream API}}

5.Remove contact

6.Exit

Enter your choice: 1

Add a Contact:

Enter the First Name: John

Enter the Last Name: Michael

Enter the Phone No.: 9787878900

Enter the Email: [John@gmail.com](mailto:John@gmail.com)

**<<success or failure message>>**

Menu

1.Add Contact

2.Display all contacts {{apply Stream API}}

3.Search contact by phone {{apply Stream API}}

4. Display all First Names {{apply Stream API}}

5.Remove contact

6.Exit

Enter your choice: 1

Add a Contact:

First Name: Jhonty

Last Name: Rhodes

Phone No.: 9787888900

Email: [Jhonty@gmail.com](mailto:Jhonty@gmail.com)

Menu

1.Add Contact

2.Display all contacts {{apply Stream API}}

3.Search contact by phone {{apply Stream API}}

4. Display all First Names {{apply Stream API}}

5.Remove contact

6.Exit

Enter your choice: 2

The contacts in the List are:

First Name: John

Last Name: Michael

Phone No.: 9787878900

Email: [John@gmail.com](mailto:John@gmail.com)

First Name: Jhonty

Last Name: Rhodes

Phone No.: 9787888900

Email: [Jhonty@gmail.com](mailto:Jhonty@gmail.com)

Menu

1.Add Contact

2.Display all contacts {{apply Stream API}}

3.Search contact by phone {{apply Stream API}}

4. Display all First Names {{apply Stream API}}

5.Remove contact

6.Exit

**Enter your choice: 3**

**Enter the Phone number to search contact:9787888900**

**The contact is:**

**First Name: Jhonty**

**Last Name: Rhodes**

**Phone No.: 9787888900**

**Email:**[**Jhonty@gmail.com**](mailto:Jhonty@gmail.com)

Menu

1.Add Contact

2.Display all contacts {{apply Stream API}}

3.Search contact by phone {{apply Stream API}}

4. Display all First Names {{apply Stream API}}

5.Remove contact

6.Exit

**Enter your choice: 4**

**---- First Names in the Phone Book are ----**

1. **John**
2. **Jhonty**

Menu

1.Add Contact

2.Display all contacts {{apply Stream API}}

3.Search contact by phone {{apply Stream API}}

4. Display all First Names {{apply Stream API}}

5.Remove contact

6.Exit

Enter your choice: 5

**Enter the Phone number to remove :9787888900**

**Do you want to remove the contact (Y/N): Y**

**The contact is successfully deleted.**