

# LAB211 Assignment

Type:  
Code:  
LOC:  
Slot(s):

Short Assignment  
J1.S.P0011  
64  
2

## Title

Develop the Contact Management Program.

## Background

N/A

## Program Specifications

Write a program to manage contacts as follows:

### CONTACT MANAGEMENT

1. Add a contact
2. Display all contacts
3. Delete a contact
4. Exit

Please choose:

### User's options:

1. If user chooses number 1, let them add a contact include **ID** (int), **fullname** (String), **group** (String), **address** (String), **phone** (String), **lastName** (String), **firstname** (String), In there **Id is automatically inscreasing** (means new contact has ID equal to last ID contact + 1), the first contact has ID: 1, **firstname**, **lastname** are taken from name by the first space .

2. If user chooses number 2, program will display the list of following data:

ID	Name	First Name	Last Name	Group	Address	Phone
1	Iker Casillas	Iker	Casillas	Star	Spain	1234567890
2	John Terry	John	Terry	Star	England	1234567890
3	Raul Gonzalez	Raul	Gonzalez	Star	Spain	1234567890

3. If user chooses 3, program will require to enter the contact ID which needs to be deleted, if ID does not exist, program will display a message as "No found contact".

4. If user chooses number 4: Exit the program.

Note: A phone number only has the following format:

- 1234567890
- 123-456-7890
- 123-456-7890 x1234
- 123-456-7890 ext1234
- (123)-456-7890
- 123.456.7890
- 123 456 7890

### Function details:

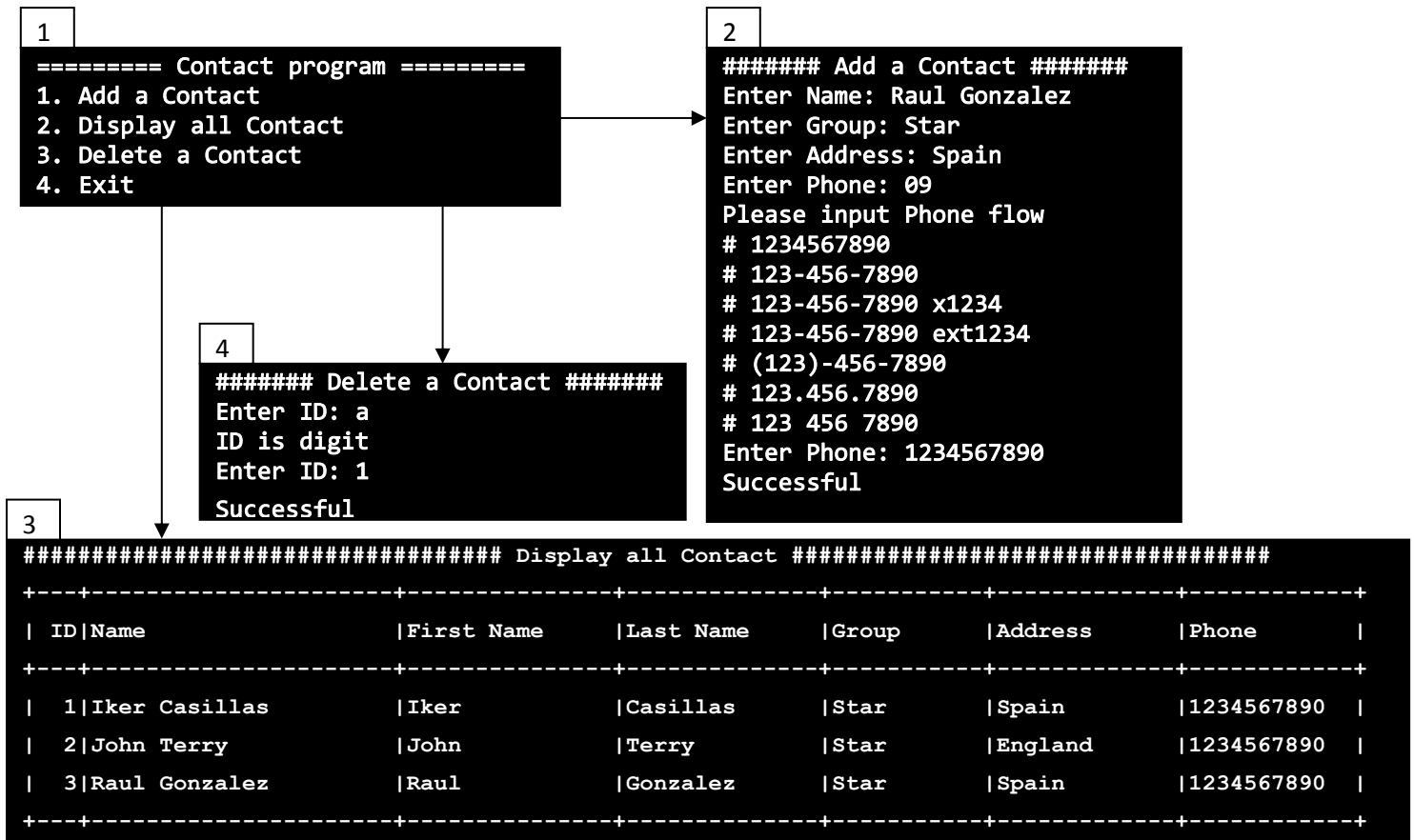
**Function 1:** Display a menu and ask users to select an option.

- User runs the program. The program prompts users to select an option.
- User selects an option, perform **Function 2**.

**Function 2:** Perform function based on the selected option.

- Option 1: Add contact
  - Require to input task information including “name, group, address, phone”.
  - Check valid data with the conditions below:
    - Phone number has to input the right format
  - Add contact to the program.
  - Return to the main screen.
- Option 2: Display the contact
  - Display all the contacts
  - Return to the program
- Option 3: Delete contact
  - Require the ID which you need to delete
  - Check valid data with the conditions below:
    - ID has to exist and is a number
- Option 4: Exit the program.

### Expectation of User interface:



## Guidelines

**Student must implement methods**

- addContact
- displayAll
- deleteContact

**in startup code.**

**Suggestion:**

**Function 1:** Add a contact.

- Implement the function: `public boolean addContact(List<Contact> list, Contact contact)`
  - Input:
    - list: where contains the contacts
    - contact: Contact's information.
  - Return value: to be more one contact status.

**Function 2:** Display the contacts list.

- Implement the function: `public void displayAll(List< Contact > list)`
  - Input:
    - list: Contact list
  - Return value: void.

**Function 3:** Delete contact

- Implement the function: `public boolean deleteContact(List<Contact> list, Contact contact)`
  - Input:
    - list: list of contact.
    - contact: the contact which we want to delete.
  - Return value: Delete contact status