Mini Phonebook Project

Author: Mousa Amran Alawadhi

# About Project

The project is a simple Contact Management System developed in Java. It simply allows users to create, retrieve, and manage contact information including name, phone number, and email address.

# Features

## Contact Creation

Users can create a new contact by providing a name, phone number, and email address. The system validates the email address to ensure it is in a valid format.

## Contact Retrieval

Users can retrieve contact information by using the getName(), getPhoneNumber(), and getEmail() methods.

# Project classes

## Contact Class

This class the name, phone number and email address of a contact and provides getter methods for each field. It also overrides the toString() method to provide a string representation of the contact, which can be useful for debugging and logging purposes.

The Contact class represents a single contact in the Contact Management System. It contains the following:

### Fields

* name: A string representing the name of the contact.
* phoneNumber: A string representing the phone number of the contact.
* email: A string representing the email address of the contact.

### Methods

#### public Contact(String name, String phoneNumber, String email)

The constructor for the Contact class. It takes a name, phone number, and email address as input and creates a new Contact object.

#### public String getName()

A method that returns the name of the contact.

#### public String getPhoneNumber()

A method that returns the phone number of the contact.

#### public String getEmail()

A method that returns the email address of the contact.

#### public String toString()

A method that returns a string representation of the Contact object.

## MobilePhone class

The MobilePhone class represents a simple mobile phone that can store a list of contacts. It provides methods to add, remove, and search for contacts in the phone's contact list.

### Fields

* contacts: A DoublyLinkedList of Contact objects that stores the phone's contact list.

### Methods

#### MobilePhone()

Constructor for creating a new MobilePhone object. It initializes the contacts field with a new instance of DoublyLinkedList.

#### public void addContact(Contact contact)

Adds a new Contact object to the contacts list.

Parameters:

* **contact:** The Contact object to be added to the list.

#### public void removeContact(Contact contact)

Removes the given Contact object from the contacts list.

Parameters:

* **contact:** The Contact object to be removed from the list.

#### public Contact searchContactByPhoneNumber(String phoneNumber)

Searches for a Contact object in the contacts list by its phone number.

Parameters:

* **phoneNumber:** The phone number of the Contact object to be searched.

Returns:

The Contact object with the given phone number, or null if no such contact is found.

#### public Contact searchContactByName(String name)

Searches for a Contact object in the contacts list by its name.

Parameters:

* **name:** The name of the Contact object to be searched.

Returns:

The Contact object with the given name, or null if no such contact is found.

#### public void printContacts()

Prints all the Contact objects in the contacts list.