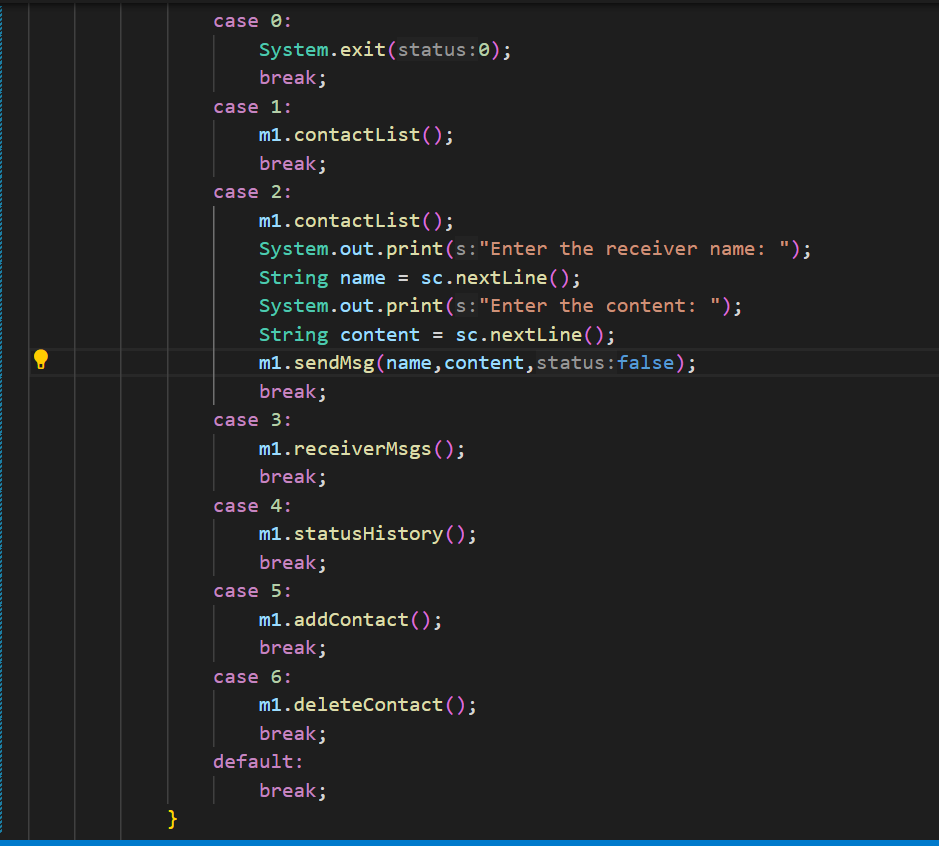
**Messaging System Assignment**

**Class Descriptions**

**1. Demo Class**

* **Purpose**: This is the main class that runs the application.
* **Functionality**:
  + Initializes the messaging system and default messages.
  + Displays a menu for the user to interact with the system.



**2. Message Class**

* **Purpose**: Represents an individual message sent in the system.
* **Attributes**:
  + receiverNo: The receiver's name .
  + msgContent: The content of the message.
  + messageId: A unique identifier for the message.
  + status: Indicates if the message has been read or not.
  + dateTime: The date and time when the message was created.
* **Functionality**:
  + Contains methods to format the message details for display.
  + Manages the status of the message.

**3. MsgSystem Class**

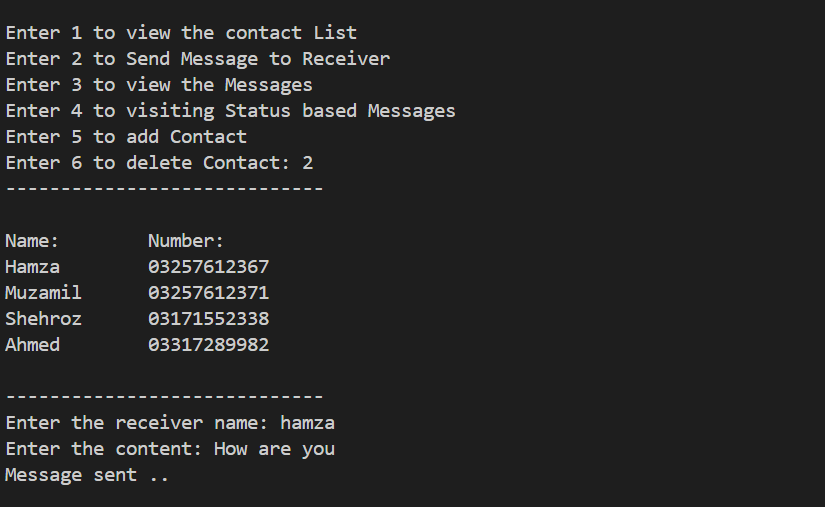
### Fields:

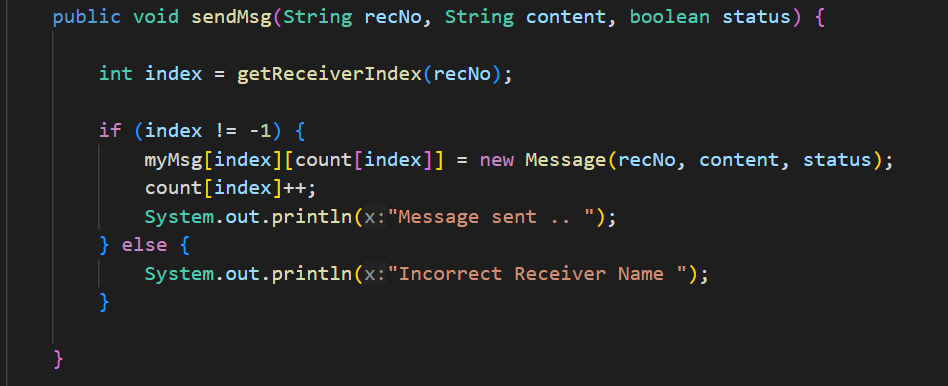
* **Scanner sc:** Used for taking user input.
* **String[][] recNo:** A 2D array storing contact names and their corresponding phone numbers.
* **static int[] count:** Tracks the number of messages sent to each contact.
* **Message[][] myMsg:** A 2D array that stores messages for each contact.

### Methods:

#### **sendMsg(String recNo, String content, boolean status)**

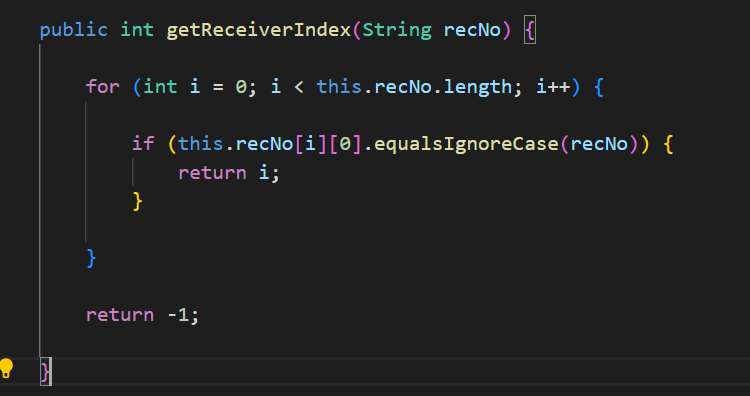
* **Purpose**: Sends a message to a specified contact.
* **Functionality**:
  + Uses **getReceiverIndex** to find the contact's index.
  + Creates a new Message object and stores it in the appropriate index.
  + Increments the message count for that contact.





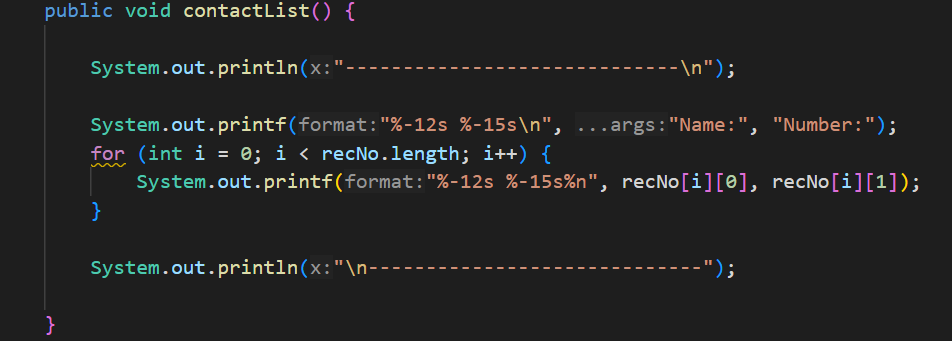
#### **getReceiverIndex(String recNo)**

* **Purpose**: Finds the index of a contact in the recNo array.
* **Functionality**: Iterates through the recNo array and returns the index of the matching name, or -1 if not found.



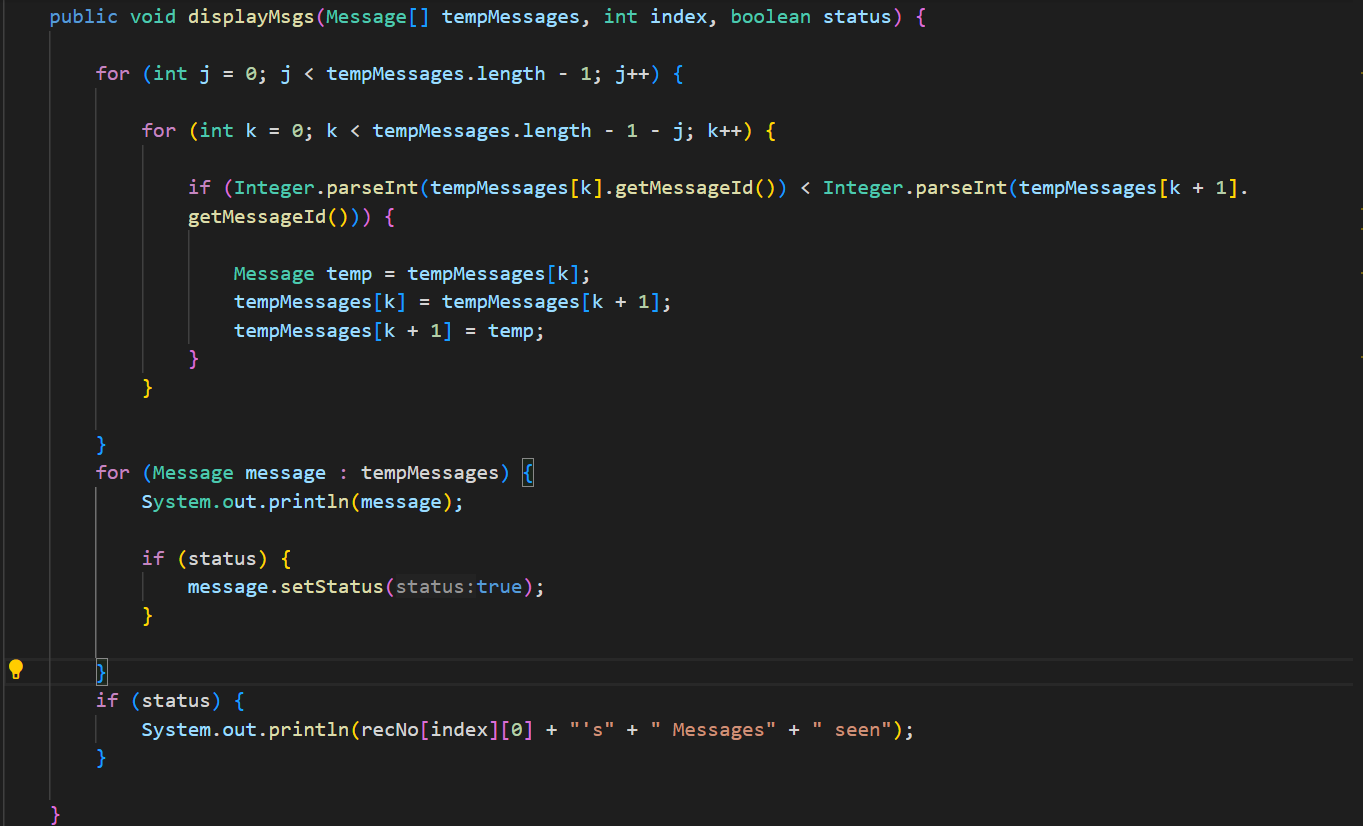
#### **contactList()**

* **Purpose**: Displays the list of contacts.
* **Functionality**: Formats and prints the names and phone numbers of all contacts in a readable manner.



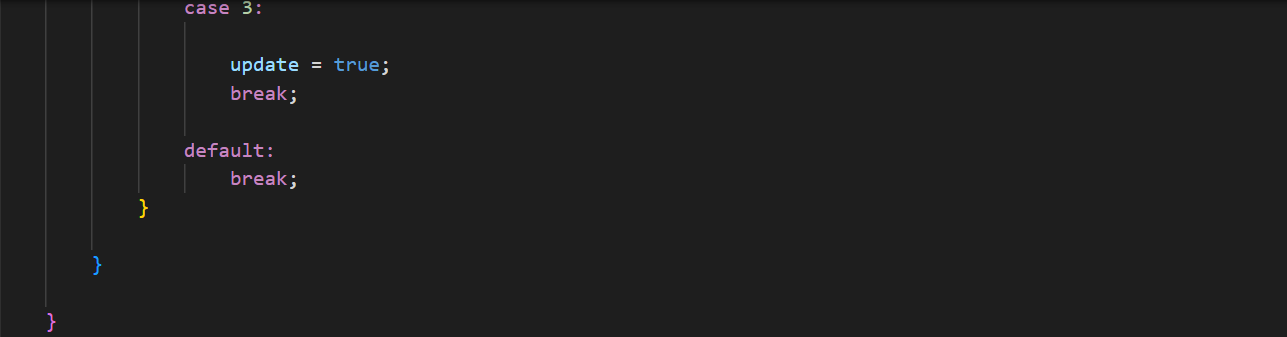
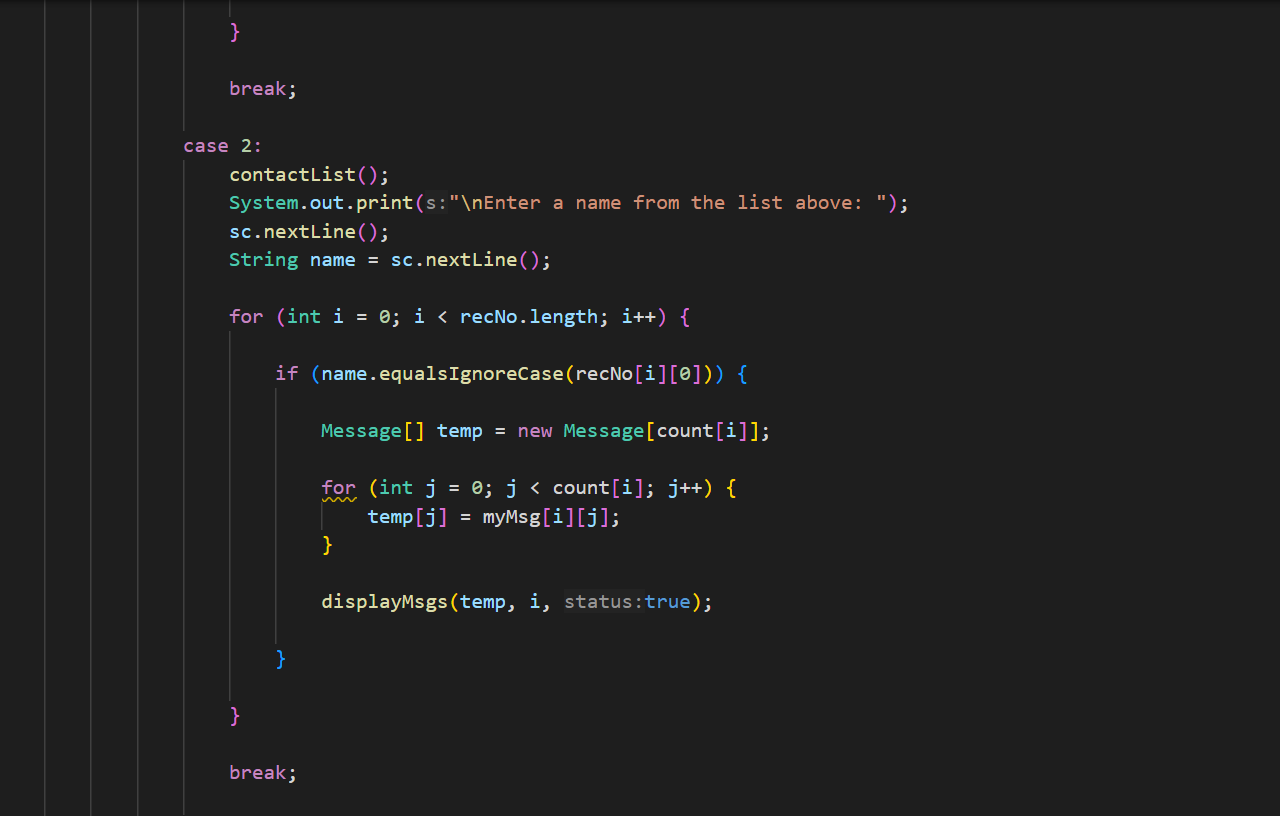
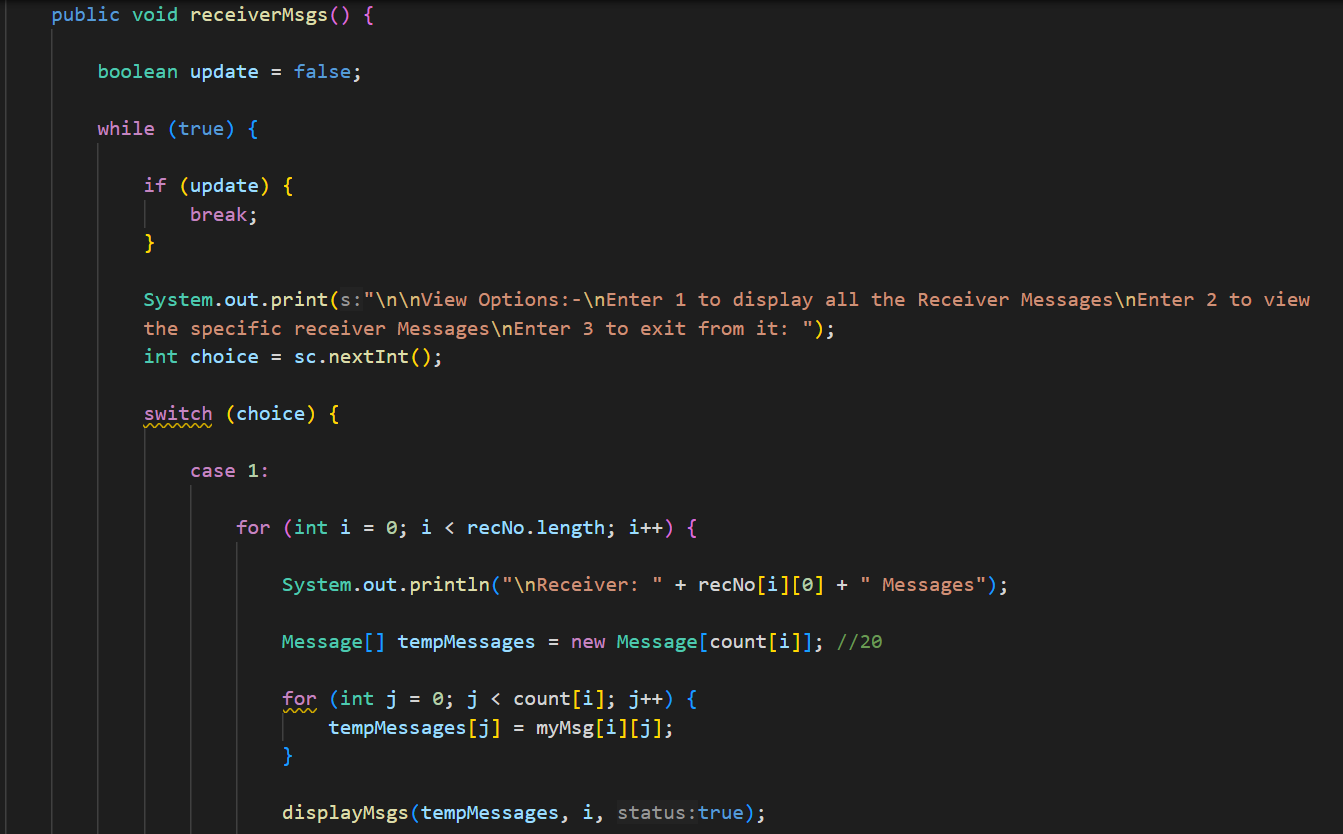
#### **displayMsgs(Message[] tempMessages, int index, boolean status)**

* **Purpose**: Displays messages for a specific contact in a sorted form like the highest id on the first number and so on . By default, messages appear in ascending order .
* **Functionality**:
  + Sorts the messages by their ID using a bubble sort algorithm.
  + Prints each message and updates the status if specified.



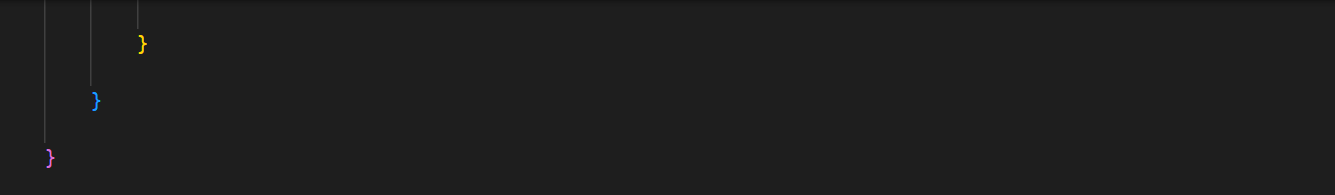
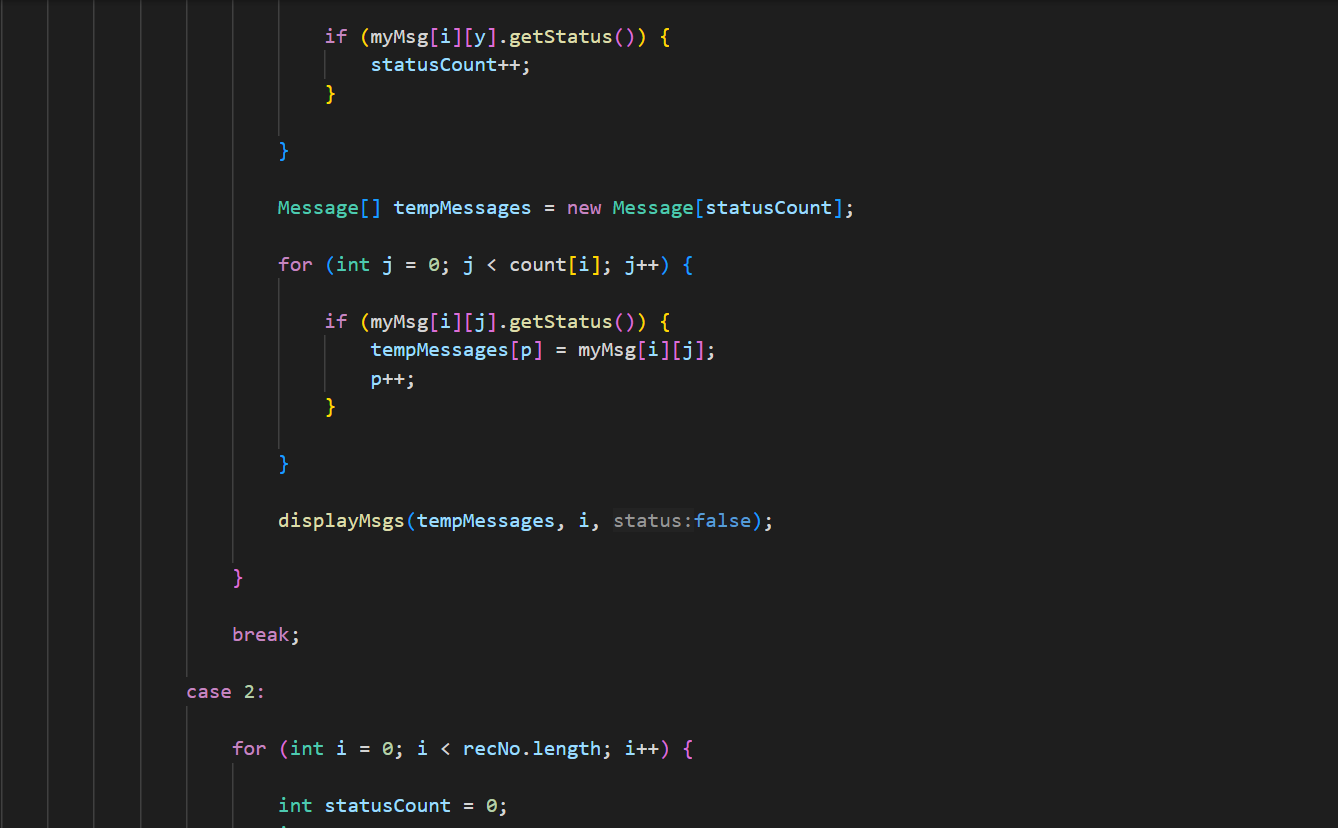
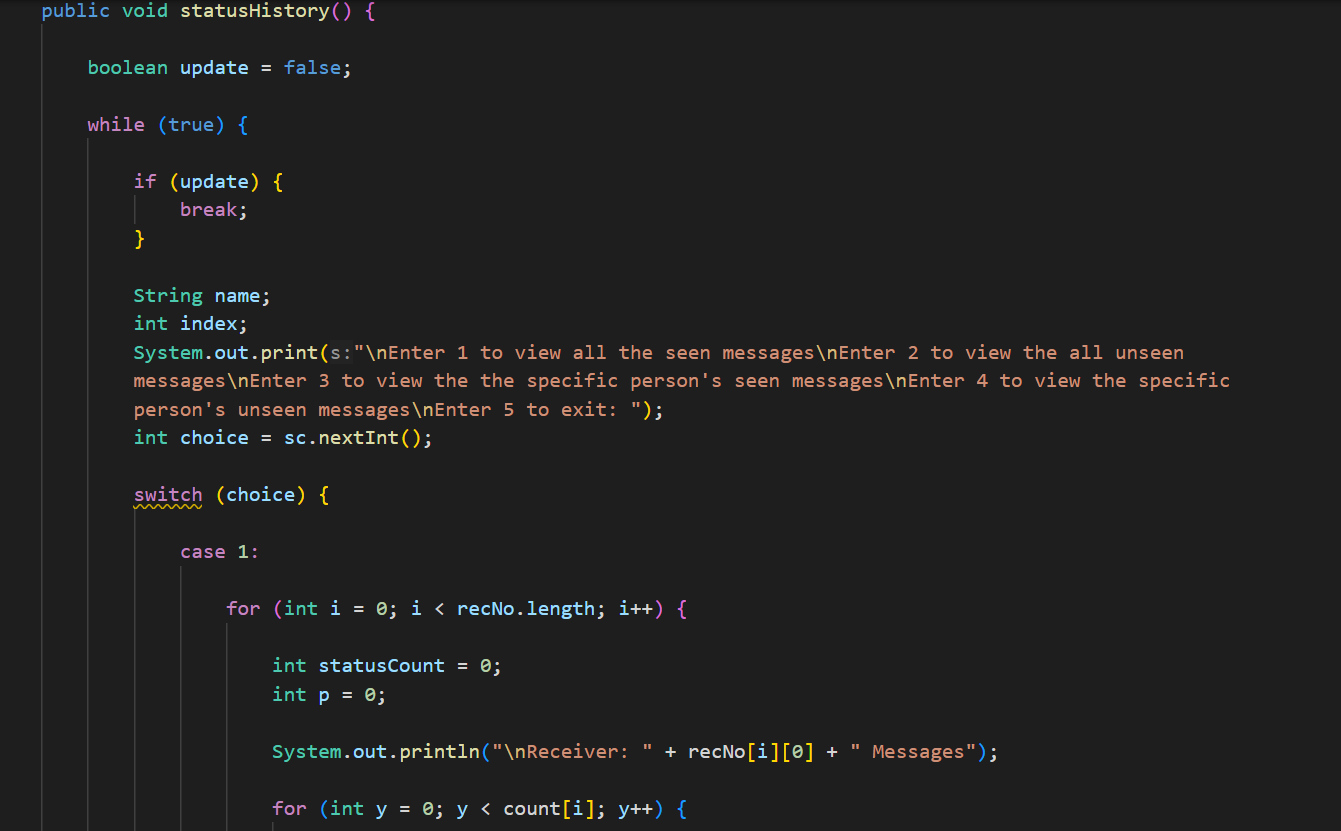
#### **receiverMsgs()**

* **Purpose**: Manages viewing of messages for contacts.
* **Functionality**:
  + Provides options to view all messages, specific messages, or exit.
  + Calls **displayMsgs** to show messages based on user choice.



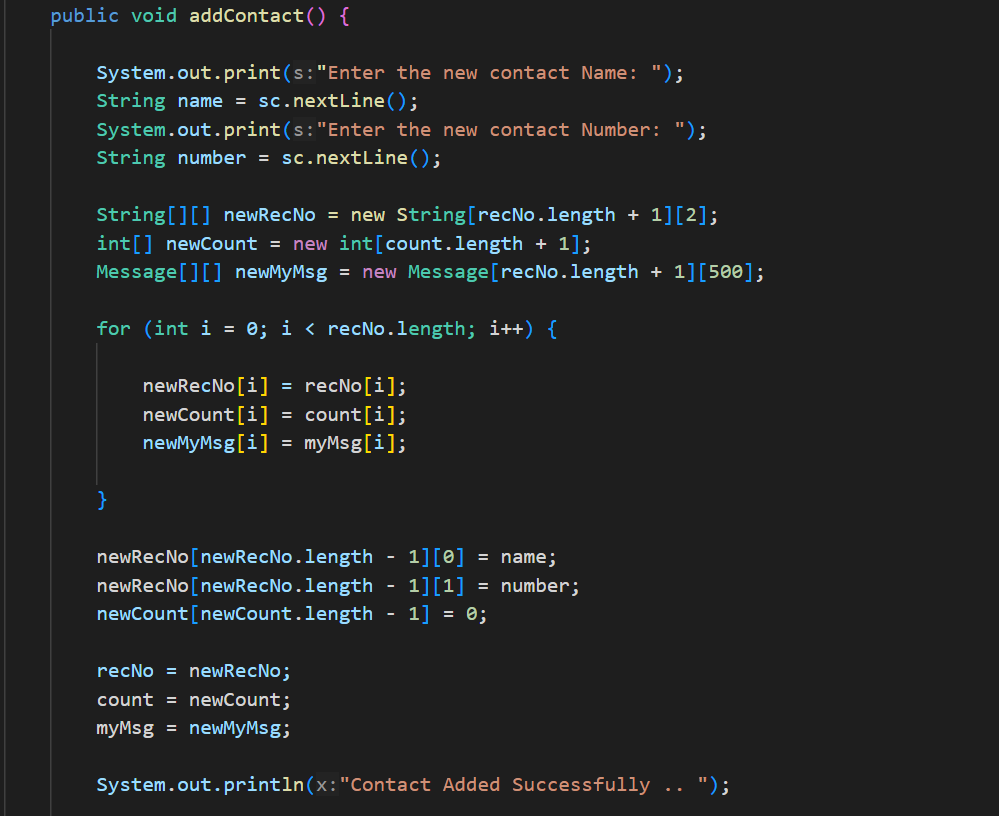
#### **statusHistory()**

* **Purpose**: Manages viewing of message statuses (seen/unseen).
* **Functionality**:
  + Provides options to view seen/unseen messages for all contacts or specific contacts.
  + Calls **displayMsgs** to show the relevant messages.



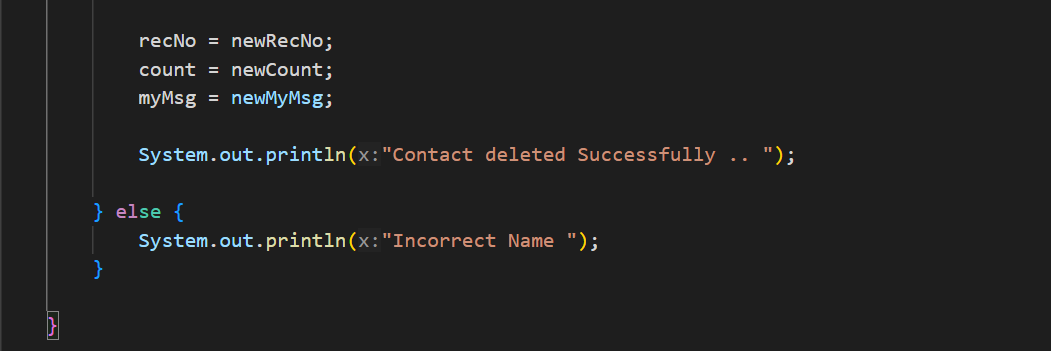
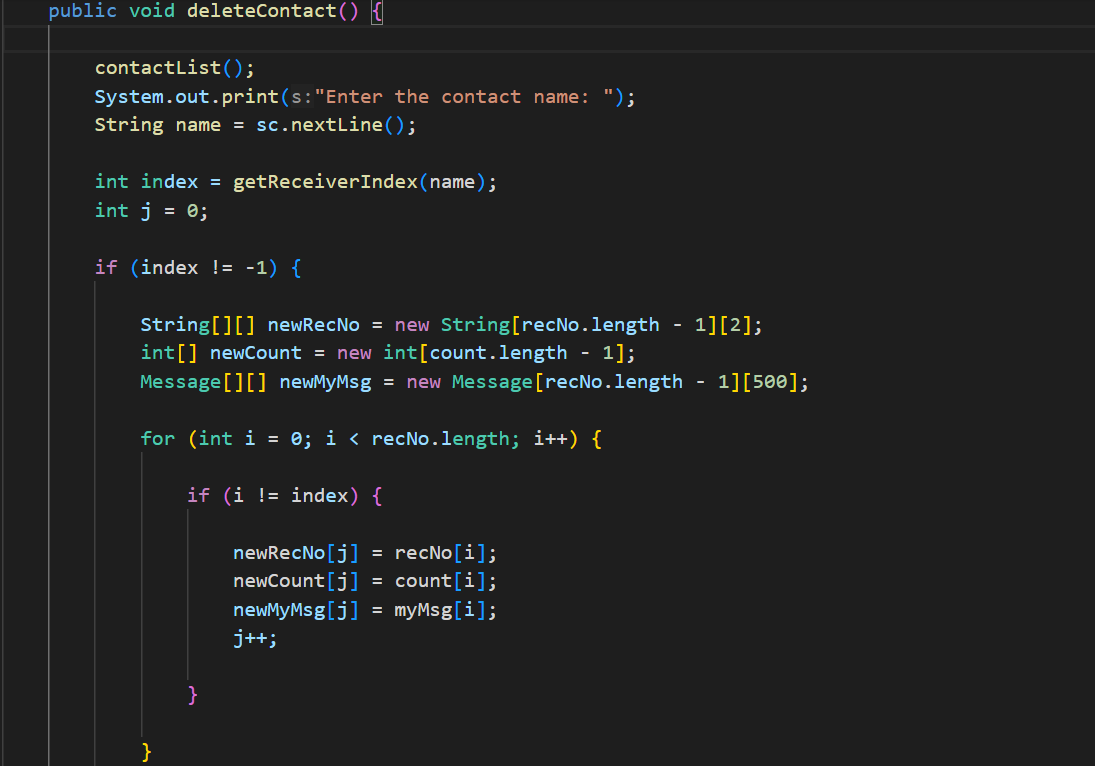
#### **addContact()**

* **Purpose**: Adds a new contact to the system.
* **Functionality**:
  + Prompts the user for a name and number, then creates new arrays to accommodate the new contact.
  + Copies existing contacts to the new array and adds the new contact at the end.



#### **deleteContact()**

* **Purpose**: Deletes an existing contact from the system.
* **Functionality**:
  + Prompts for a contact name, finds the index, and creates new arrays without the deleted contact.
  + Copies existing contacts to the new array, skipping the one to be deleted.



**4. DefaultMsgs Class**

* **Purpose**: Default messages for each contact.
* **Functionality**:
  + Uses to create default messages for each contact in the system.

