**OOP Assignment**

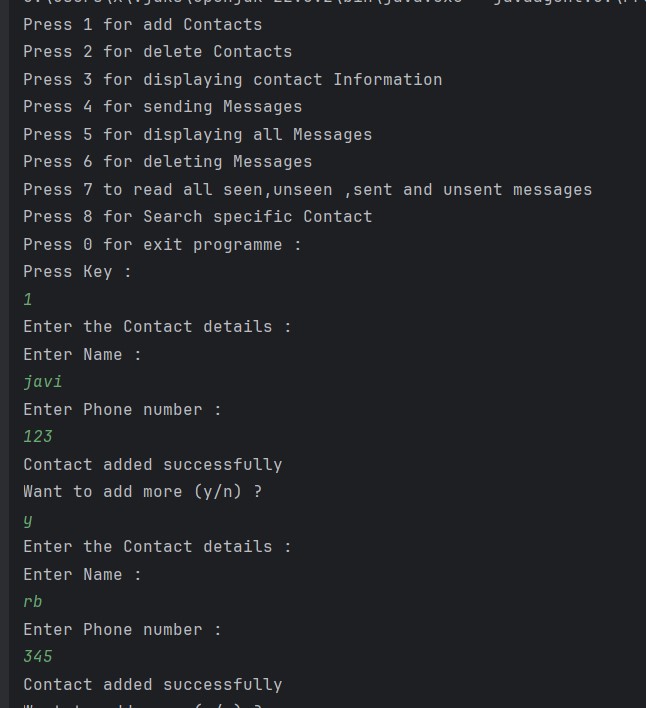
**Name : Areej Abrar**

**Roll NO : SP24-BSE-141**

**Section : B**

**Code Output : Step by Step**

**Firstly 4 classes are made in which There are 8 Functions :**



**1st Function : Add Contact**

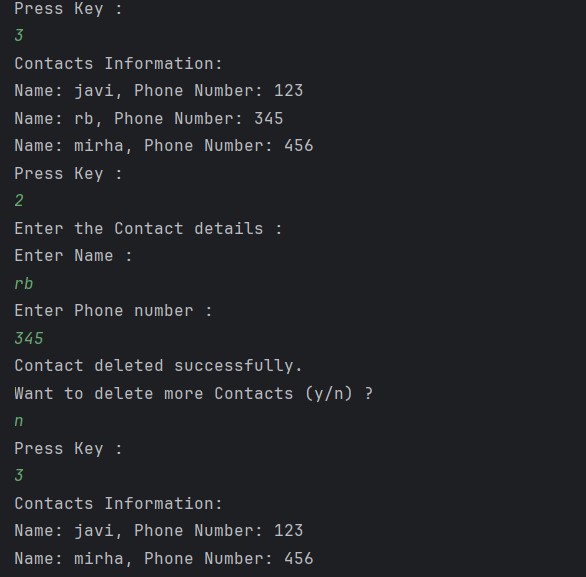
Name (variable) and phone Number(variable) of that Contact will be saved in the Contact List (List of that Contact made in Messaging App Class)….While Loop is used so how many user want can add the contacts.

**2nd Function : Delete Contact**

* Deleting Contact by asking the Contact Name and Phone Number .
* While loop also used here .
* That Contact will be equal to null means that Contact will be deleted.
* No of contacts (count--) will be decreased

**3rd Function : Display Contact**

All the Contacts will be displayed by the nested loop as Contact is in 2D array



**4th Function : Send Messages**

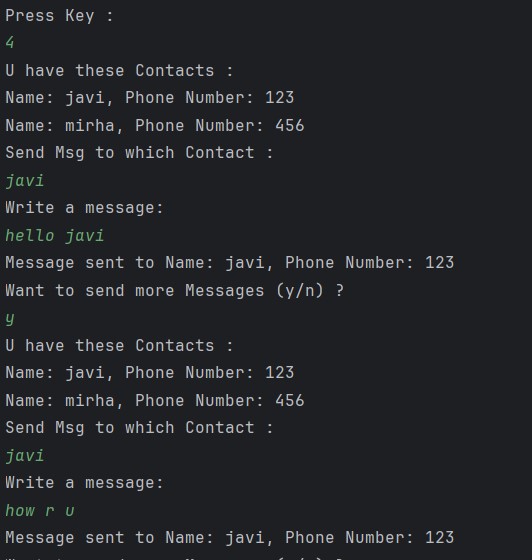
* First all the saved (add) Contacts will be displayed to the user that user can send to these contact otherwise it will show the error
* Use of Enhanced For Loop
* Use of condition to check if the name enter by user match with the contact(saved) names or not. If yes then it will ask the user to write message then the message will be sent to that contact(receiver at that time)…

**Some part of code :**

if (value != null && value.getName().equals(n)) {  
 System.*out*.println("Write a message: ");  
 String msg = input.nextLine();  
 if (*msg\_count* < messages.length) {  
  
 messages[*msg\_count*] = new Msg((id + 1), value, msg, true, "seen");  
  
 System.*out*.println("Message sent to " + value);  
 if (*msg\_count* % 4 == 0) {  
 messages[*msg\_count*].setStatus(false);  
 messages[*msg\_count*].setSeen("Msg not send so can't be seen");  
 }  
 if (*msg\_count* % 3 == 0 && messages[*msg\_count*].isStatus()) {  
 messages[*msg\_count*].setSeen("unseen");  
  
 }  
 *msg\_count*++;  
 id++;

Assigning the status to the messages that which messages will be sent or which not and also which will be seen and which will be not.

Within the each message msg\_count will be increased



**5th Function : Display Messages**

* Message will be displayed acc to timestamp
* Later messages will be shown first like in Messaging App
* For this Assorting Algorithm is used and “CompareTo” Method is used in condition while doing assorting (Assorting in descending order)
* And Obv for loop is used in displaying Messages …

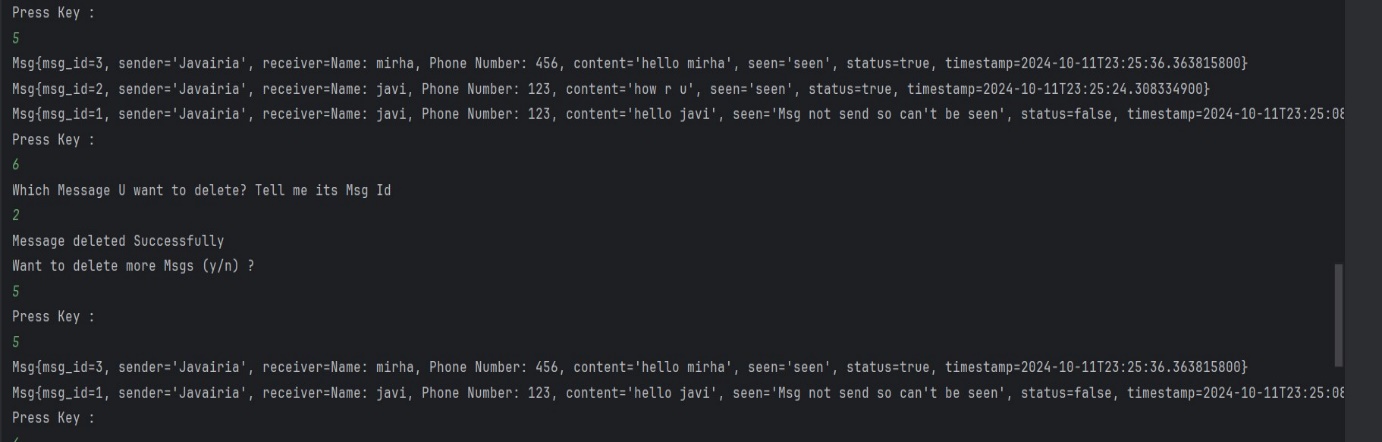
**Some main part of code in assorting :**

if(messages[j].getTimestamp().compareTo(messages[j+1].getTimestamp())<0)

**6th Function : Deleting Messages**

As the msg id has been assigned to all Messages (in send method)

Message will be deleted by asking msg\_id from the user and then that message will be deleted as U can see in the Output



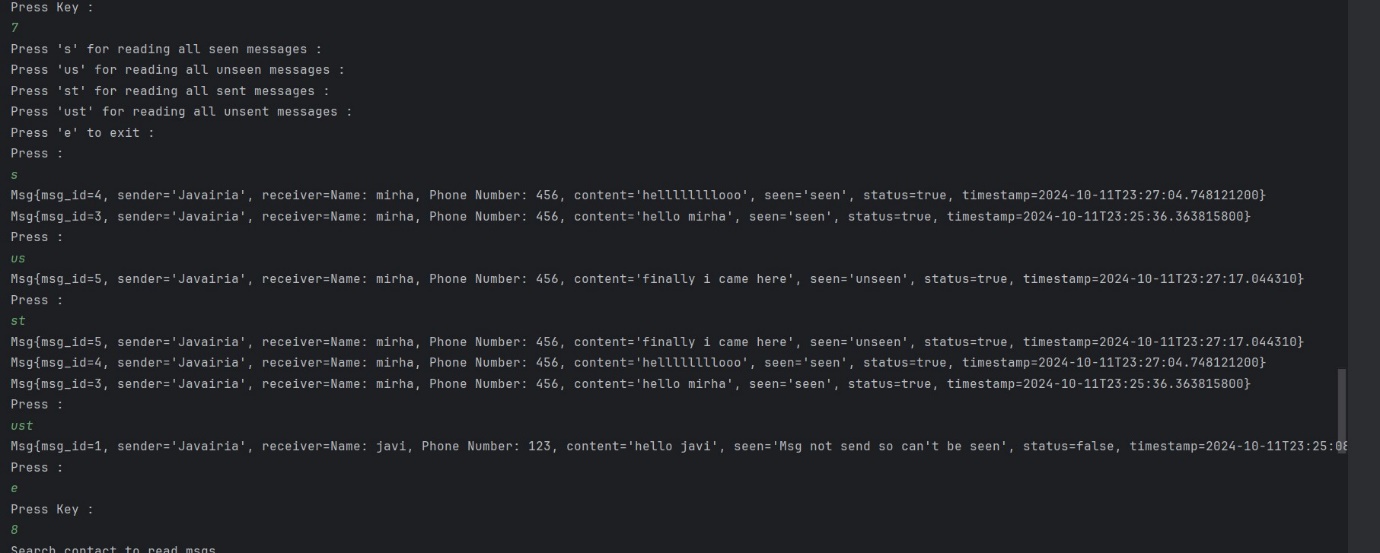
**7th Function : Want to see all Seen /Unseen/Sent/Unsent Messages**

There are 5 option has been given to user while using switch case that user want to see all seen msgs or unseen or sent or unsent and if not want see anything or wanted to exit from this function then simply press e (to exit)

**1st Picture :** As I have to show the working of 7th Function So I added more messages



**2nd Picture :** Showing Output acc to 7th mesthod



**8th Function : Searching For Specific Contact**

User will search the specific contacts from which he added before and then if the name matched then user apply all the method to that specific contact messages

* Like User can see :
* Seen Messages
* Unseen Messages
* Sent Messages
* Unsent Messages OF THAT SPECIFIC CONTACT
* AND ALSO can delete messeage from the messages of that Contact

