**Source File of OOP Assignment No 1**

**Methods:**

1. public String getReciever(): Returns the receiver's name.
2. public String getcontent(): Returns the message content.
3. public boolean getseen(): Returns the seen status of the message.
4. public boolean isSeen(): Sets the seen status to true and returns it.
5. public String getmessageid(): Returns the unique message ID.
6. public LocalDateTime gettimestamp(): Returns the timestamp of the message.
7. @Override public String toString(): Returns a string representation of the message, including ID, receiver, content, status, and timestamp.

**Class: MessageApp**

**Attributes:**

1. String[] Contactlist: An array containing the names of contacts.
2. static int[] count: An array to track the number of messages for each contact.
3. Scanner sc: A scanner for user input.
4. Message message[][]: A 2D array to store messages for each contact.

**Constructor Block:**

* Initializes the message array to hold messages for each contact.

**Methods:**

1. public int getcontactindex(String contact): Returns the index of a contact in the Contactlist. Returns -1 if the contact is not found.
2. public void Sendmessage(String Contact, String content, boolean status): Creates a new Message object and stores it in the message array for the specified contact. Prints a confirmation message or an error if the contact is invalid.
3. public void displayContact(): Displays the list of contacts.
4. public void addcontact(): Prompts the user to enter a new contact name and adds it to the Contactlist, updating the relevant arrays.
5. public void deletecontact(): Prompts the user to enter a contact name to delete and removes it from the Contactlist, updating the relevant arrays.
6. public void Alldisplay(): Displays all messages for all contacts, iterating through the message array.
7. public void Specificdisplay(String name): Displays messages for a specific contact.
8. public void checkStatus(String id): Checks and displays the details of a message based on its ID.

**Class: Test3**

**Attributes:**

* This class serves as the entry point for the application.

**Methods:**

1. public static void main(String[] args):
   * Initializes the MessageApp and Storedmsg objects.
   * Contains a loop to interact with the user, prompting for various actions (sending messages, displaying contacts, etc.) and invoking the corresponding methods in MessageApp.

**Class: Storedmsg**

**Methods:**

1. public void storemsg(MessageApp str): Initializes the Message objects for each contact with a default message ("Hi") to demonstrate functionality.

Bottom of Form