```
package EmailClient;

// Index number:200045U
// Name        :H.W.K.Aravinda

//import libraries

import java.io.IOException;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;
import java.util.InputMismatchException;
import java.util.Scanner;
import java.io.Serializable;
import java.io.*;
import java.util.ArrayList;
import javax.mail.*;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;
import java.util.Properties;
import java.io.FileOutputStream;
import java.io.ObjectOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.BufferedWriter;
import java.io.FileWriter;

class Email_Client {

    public static void main(String[] args) throws IOException {
        boolean Loop=true;
        while (Loop){
            Scanner scanner = new Scanner(System.in);
            System.out.println("""
                    Enter option type:\s
                    1 - Adding a new recipient
                    2 - Sending an email
                    3 - Printing out all the recipients who have birthdays
                    4 - Printing out details of all the emails sent
                    5 - Printing out the number of recipient objects in the
application""");
            try{
                int option = scanner.nextInt();

            //Taking current date by importing java library.
            DateTimeFormatter currentDate =
DateTimeFormatter.ofPattern("yyyy/MM/dd");
            LocalDateTime now = LocalDateTime.now();
            boolean addRecipient=false;
            SetMailClient.setMailClient();
            switch (option) {
                case 1 -> {
                    System.out.println("""
                    Enter recipients as following input:\s
                    Official:nimal,nimal@gmail.com,ceo
                    Office_friend:kamal,kamal@gmail.com,clerk,2000/12/12
                    Personal:sunil,<nick-name>,sunil@gmail.com,2000/10/10""");
                    scanner.nextLine();
                    String input1 = scanner.nextLine();
                    WriteFile.write("clientList.txt", input1);
                    System.out.println("The new recipient is successfully added to
the clientList.txt file.");
                    addRecipient=true;
                }
                case 2 -> {
                    System.out.println("Enter the email,subject and content using
coma seperated: ");
                    scanner.nextLine();
                    String input2 = scanner.nextLine();
                    //Spliting the input
                    String[] array2 = input2.split(",");

                    //Sending the mail using email,subject and content given by the
user.
                    try{
                        SendEmail newMail = new SendEmail(currentDate.format(now),
array2[0], array2[1], array2[2]);

                        newMail.sendMail();

                        //Serialization newMail objects
                        SerializationObject serObject = new
SerializationObject("obj.ser", newMail);
                        serObject.serialization();
                        System.out.println("The email was sent successfully.");
                    }catch(ArrayIndexOutOfBoundsException error){
                        System.out.println("Input format is false.");
                    }

                }
                case 3 -> {
                    System.out.println("Enter the date [input format - yyyy/MM/dd
(ex: 2018/09/17)]");
                    scanner.nextLine();
                    String input3 = scanner.nextLine();
                    //Taking the array list as recipients who have birthdays on the
given date.
                    ReadFile.read("clientList.txt", input3);
                    ReadFile.printDetails(input3);
                }
                case 4 -> {
                    System.out.println("Enter the date [input format - yyyy/MM/dd 
(ex: 2018/09/17)]");
                    scanner.nextLine();
                    String input4 = scanner.nextLine();
                    //Making array list for saving SendMail objects after 
deserializing objects from text file.
                    ReadObjectFile.read("obj.ser");
                    ReadObjectFile.printMail(input4);
                }
                case 5 -> {
                    System.out.println("Number of recipients objects in the 
application: ");
                    ReadFile.read("clientList.txt", "             ");
                    System.out.println(Recipient.getNumberOfRecipients());
                }
            }
            SetMailClient.setMailClient();
            //Check whether that there are recipients who have birthday on current 
date.
            BirthdayWishing.wish(addRecipient);

            }catch (InputMismatchException error){
                System.out.println("This input format is false.Give a correct 
format.");
            }
            System.out.println("""
                    Do you want to continue Email Client operations?\s
                    1:Yes
                    2:No""");
            int terminate = scanner.nextInt();
            switch (terminate){
                case 1->{
                }
                case 2-> Loop=false;
            }
        }
    }
}

//===========================================================================

package EmailClient;

class Recipient implements Serializable {
    private static int numberOfRecipients=0;
    private String name;
    private final String emailAddress;

    public Recipient(String name,String emailAddress) {
        this.emailAddress=emailAddress;
        this.name=name;
        numberOfRecipients++;
    }

    //Returning the number of recipients
    public static int getNumberOfRecipients() {
        return numberOfRecipients;
    }

    //Returing the names of the recipients
    public String getName() {
        return name;
    }

    //returning the mail address of the recipients
    public String getEmailAddress() {
        return emailAddress;
    }

    public static void setNumberOfRecipients(){
        numberOfRecipients=0;
    }

}

//===========================================================================

package EmailClient;

class Official extends Recipient {
    private String position;

    //Constructor of the Official class
    public Official(String[] content) {
        super(content[0],content[1]);
        this.position = content[2];
    }


}

//===========================================================================

package EmailClient;

class OfficialFriend extends Recipient implements EmailSendable {
    private String birthDay;
    private String position;
    public OfficialFriend(String[] content) {
        super(content[0],content[1]);
        this.birthDay = content[3].substring(5,10);
        this.position = content[2];
        addBirthdayList(this);
    }

    public String getBirthDay() {
        return birthDay;
    }

    //Sending mails for official friends
    public void sendEmail(String date) throws IOException {
        String subject="Birthday wishing";
        String content="Wish you a Happy Birthday. \n\n Aravinda";
        SendEmail newMail=new
SendEmail(date,this.getEmailAddress(),subject,content);
        newMail.sendMail();

        //Serialization send office friend birthday wishes
        SerializationObject serializationObject=new
SerializationObject("obj.ser",newMail);
        serializationObject.serialization();


    }

    //Add to the birthday
    public void addBirthdayList(OfficialFriend object){
        BirthdayList.createBirthdayList(object);
    }


}

//===========================================================================

package EmailClient;

class Personal extends Recipient implements EmailSendable{
    private String birthDay;
    private String nickName;

    public Personal(String[] content) {
        super(content[0],content[2]);
        this.birthDay = content[3].substring(5,10);
        this.nickName = content[1];
        addBirthdayList(this);
    }
    public String getBirthDay() {
        return birthDay;
    }

    public String getNickName() {
        return nickName;
    }

    public void sendEmail(String date) throws IOException {
        String subject="Birthday wishing";
        String content="Hugs and love on your birthday. \n\n Aravinda";
        SendEmail newMail=new
SendEmail(date,this.getEmailAddress(),subject,content);
        newMail.sendMail();

        //Serialization send personal birthday wishes
        SerializationObject serializationObject=new
SerializationObject("obj.ser",newMail);
        serializationObject.serialization();
    }

    public void addBirthdayList(Personal object){

        BirthdayList.createBirthdayList(object);
    }
}

//===========================================================================

package EmailClient;

interface EmailSendable{
    //Interface to send mails
    void sendEmail(String date) throws IOException;
}

//===========================================================================

package EmailClient;

class RecipientFactory {

    public void makeRecipient(String[] content) {
            String[] array1=content[1].split(",");

            //Factory to make recipients
            switch (content[0]){
                case "Official"->{
                    Official official = new Official(array1);
                }
                case "Office_friend"->{
                    OfficialFriend oofficialFriend = new OfficialFriend(array1);
                }
                case "Personal"->{
                    Personal personal = new Personal(array1);
                }
            }
    }
}


//===========================================================================

package EmailClient;

class BirthdayList {

    //Making array list of recipients who has birthdays on given date
    private static ArrayList<Recipient> objectList = new ArrayList<>();

    //Setting the array list
    public static void createBirthdayList(Recipient object){
        objectList.add(object);
    }

    public static ArrayList<Recipient> getObjectList() {
        return objectList;
    }

    public static void clearBirthdayList(){
        objectList.clear();
    }

}

//===========================================================================

package EmailClient;

class BirthdayWishing {

    public static void wish(boolean addRecipient) throws IOException {

        //Taking current date
        DateTimeFormatter currentDate = DateTimeFormatter.ofPattern("yyyy/MM/dd");
        LocalDateTime now = LocalDateTime.now();
        ReadFile.read("clientList.txt",currentDate.format(now));

        //Check the existence of recipient list file
        if (ReadFile.isExistFile()){
            ArrayList<Recipient> objList=ReadFile.getBirthdayObjectList();

            boolean Loop=true;
            ReadObjectFile.read("obj.ser");
            if (ReadObjectFile.isExistFile()){
                ArrayList<SendEmail> sendEmailObjects =
ReadObjectFile.getSendEmailsObjects();
                for (SendEmail mails : sendEmailObjects) {

                    //Check whether today the email client has send birthday wishes
                    if (mails.getSubject().equals("Birthday
wishing")&&mails.getDate().equals(currentDate.format(now))){
                        Loop=false;
                    }
                }
            }
            //Sending birthday wishes
            if (Loop){
                for (Recipient recipient : objList) {
                    if (recipient instanceof Personal){
                        //Send mail to a personal recipient.
                        EmailSendable personal = (Personal) recipient;
                        personal.sendEmail(currentDate.format(now));
                        System.out.println("Birthday wish was sent to the
"+recipient.getName());

                    }
                    if (recipient instanceof OfficialFriend){
                        //Sending mail to a office friend.
                        EmailSendable officeFriend = (OfficialFriend) recipient;
                        officeFriend.sendEmail(currentDate.format(now));
                        System.out.println("Birthday wish was sent to the
"+recipient.getName());

                    }
                }
            }

            //Check whether the recipient who is added today and also his or her
birthday is today
            if (addRecipient&&!Loop){
                if (objList.get(objList.size() - 1) instanceof Personal personal){
                    if
(personal.getBirthDay().equals(currentDate.format(now).substring(5,10))){
                        EmailSendable personal1=(Personal)
objList.get(objList.size()-1);
                        personal1.sendEmail(currentDate.format(now));
                        System.out.println("Birthday wish was sent to the
"+personal.getName());
                    }
                }
                if (objList.get(objList.size() - 1) instanceof OfficialFriend
officialFriend){
                    if
(officialFriend.getBirthDay().equals(currentDate.format(now).substring(5,10))){
                        EmailSendable officialFriend1=(OfficialFriend)
objList.get(objList.size()-1);
                        officialFriend1.sendEmail(currentDate.format(now));
                        System.out.println("Birthday wish was sent to the
"+officialFriend.getName());
                    }
                }
            }

        }
    }
}

//=============================================================================

package EmailClient;

class SendEmail implements Serializable {

    private static final long serialVersionUID=-8307237345658518819L;
    private String date;
    private String email;
    private String subject;
    private String content;

    public SendEmail(String date, String email, String subject, String content) {
        this.date = date;
        this.email = email;
        this.subject = subject;
        this.content = content;
    }


    //Sending the mail method
    public void sendMail() {
        final String username = "aravindahwk@gmail.com";
        final String password = "zlck ihsk umbl zxmv";

        Properties prop = new Properties();
        prop.put("mail.smtp.host", "smtp.gmail.com");
        prop.put("mail.smtp.port", "587");
        prop.put("mail.smtp.auth", "true");
        prop.put("mail.smtp.starttls.enable", "true"); //TLS

        Session session = Session.getInstance(prop,
                new javax.mail.Authenticator() {
                    protected PasswordAuthentication getPasswordAuthentication() {
                        return new PasswordAuthentication(username, password);
                    }
                });

        try {

            javax.mail.Message message = new MimeMessage(session);
            message.setFrom(new InternetAddress("aravindahwk@gmail.com"));
            message.setRecipients(
                    Message.RecipientType.TO,
                    InternetAddress.parse(email)
            );
            message.setSubject(subject);
            message.setText(content);

            Transport.send(message);

        } catch (MessagingException e) {
            e.printStackTrace();
        }
    }

    //Returning the date
    public String getDate() {
        return date;
    }

    //Returning the email address
    public String getEmail() {
        return email;
    }

    //Returning the subject
    public String getSubject() {
        return subject;
    }

}

//===========================================================================

package EmailClient;

class SerializationObject {

    String outputFile;
    SendEmail mail;
    public SerializationObject(String outputFile, SendEmail mail) {
        this.outputFile = outputFile;
        this.mail = mail;
    }

    public void serialization() throws IOException {
        ObjectOutputStream objectOutputStream = null;
        FileOutputStream fileOutputStream = null;
        try {
            fileOutputStream = new FileOutputStream(outputFile, true);
            objectOutputStream = new ObjectOutputStream(fileOutputStream);
            objectOutputStream.writeObject(mail);
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            if (objectOutputStream!=null){
                objectOutputStream.close();
            }
            if (fileOutputStream!=null) {
                fileOutputStream.close();
            }
        }
    }
}

//===========================================================================

package EmailClient;

class DeserializationObject {

        //Making array list of send mails
        ArrayList<SendEmail> sendEmailObjects = new ArrayList<>();


        //Read the objects from the obj.ser file and add them to the array list.
        public void setSendEmailObjects() throws IOException {
            ObjectInputStream objectInputStream = null;
            FileInputStream fileInputStream = null;
            try {
                File filename = new File("obj.ser");
                fileInputStream = new FileInputStream(filename);
                SendEmail sendMail;
                while (true) {
                    try {
                        objectInputStream = new ObjectInputStream(fileInputStream);
                        sendMail = (SendEmail) objectInputStream.readObject();
                        sendEmailObjects.add(sendMail);
                    } catch (EOFException e) {
                        break;
                    }
                    if (sendMail == null) {
                        break;
                    }

                }

            } catch (ClassNotFoundException error) {
                System.out.println("Error: "+error.getMessage());
            } finally {
                assert objectInputStream != null;
                objectInputStream.close();
                fileInputStream.close();
            }
        }


        //Returning the send mail object array list
        public ArrayList<SendEmail> getSendEmailObjects() {
            return sendEmailObjects;
        }
}

//===========================================================================

package EmailClient;

class ReadFile {
    private static boolean existFile;
    private static ArrayList<Recipient> birthdayObjectList =new ArrayList<>();

    //Read the clientList.txt file
    public static void read(String fileName,String input) throws
FileNotFoundException {
        File myFile1=new File(fileName);
        if (myFile1.exists()){
            Scanner myReader = new Scanner(myFile1);
            while (myReader.hasNextLine()){
                String data = myReader.nextLine();
                String[] array = data.split(":");
                RecipientFactory factory = new RecipientFactory();
                factory.makeRecipient(array);

            }
            myReader.close();

            //Making birthday list array.
            for (Recipient object: BirthdayList.getObjectList()){
                if (object instanceof Personal per){
                    if (per.getBirthDay().equals(input.substring(5,10))){
}
                        birthdayObjectList.add(object);
                        existFile=true;
                    }
                }
                if (object instanceof OfficialFriend offFri){
                    if (offFri.getBirthDay().equals(input.substring(5,10))){
                        birthdayObjectList.add(object);
                        existFile=true;
                    }
                }
            }

        }else{
            existFile=false;
        }
    }


    //Printing the details that every recipient who has birthday on given date.
    public static void printDetails(String input){
        if (existFile){
            for (Recipient recipient : birthdayObjectList) {
                //Printing out their names.
                System.out.println("Recipient Name: "+recipient.getName()+
                        ", Recipient Email Address: "+recipient.getEmailAddress()+
                        ", Birthday Date: "+input.substring(5,10));
            }
        }else{
            System.out.println("There is no client file to read or there is no
recipient who has birthday on"+input);
        }

    }

    //Returning the existence of the file

    public static boolean isExistFile() {
        return existFile;
    }

    //Returning the array list of reciepients who have birthday on given date.
    public static ArrayList<Recipient> getBirthdayObjectList() {
        return birthdayObjectList;
    }

    public static void clearBirthdayObject(){
        birthdayObjectList.clear();
    }
}
```

```
//===========================================================================

package EmailClient;

class WriteFile {
    public static void write(String fileName,String input) throws IOException {
        //Create file if there is no file in the folder or appending clients list if
there is the file in the folder.
        FileWriter file = new FileWriter(fileName, true);
        BufferedWriter bufferedFile = new BufferedWriter(file);
        bufferedFile.write(input);
        bufferedFile.newLine();
        //Close the file.
        bufferedFile.close();
        file.close();
    }
}

//===========================================================================
package EmailClient;

class ReadObjectFile {
    private static boolean existFile;
    public static ArrayList<SendEmail> sendEmailsObjects;

    //Read the object file
    public static void read(String fileName) throws IOException {
        File myFile=new File(fileName);
        if (myFile.exists()){
            existFile=true;
            DeserializationObject DeseObject=new DeserializationObject();
            DeseObject.setSendEmailObjects();
            sendEmailsObjects = DeseObject.getSendEmailObjects();

        }else{
            existFile=false;
        }
    }


    //Printing the details of mails that has been sent on given date
    public static void printMail(String input){
        //Printing the mail details
        if (existFile){
            for (SendEmail mails : sendEmailsObjects) {
                if (input.equals(mails.getDate())) {
                    System.out.println("Date: "+mails.getDate()+
                            ", To: "+mails.getEmail()+
                            ", Email Subject: "+mails.getSubject());
                }
            }
        }
    }

    //Return the existence of object file
    public static boolean isExistFile() {
        return existFile;
    }

    //Return the send mails as an Array list
    public static ArrayList<SendEmail> getSendEmailsObjects() {
        return sendEmailsObjects;
    }
}

//===========================================================================

package EmailClient;

class SetMailClient {
    public static void setMailClient(){
        ReadFile.clearBirthdayObject();
        BirthdayList.clearBirthdayList();
        Recipient.setNumberOfRecipients();
    }
}
```