

```

// your index number
//200452N Navinda Mansitha Perera (Perera D.N.M)
import java.util.ArrayList;
import java.util.Scanner;
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;
import javax.mail.*;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;
import java.util.Properties;
import java.io.Serializable;
import java.io.*;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.ObjectOutputStream;

class Email_Client {

    public static void main(String[] args) {
        LocalDate today_object = LocalDate.now();
        DateTimeFormatter myFormatObj =
DateTimeFormatter.ofPattern("yyyy/MM/dd");
        String date_now= today_object.format(myFormatObj);
        System.out.println(date_now.substring(0, 10));

        // taking input of the required the required task to perform

        Scanner scanner0 = new Scanner(System.in);
        System.out.println("Enter option type: \n"
            + "1 - Adding a new recipient\n"
            + "2 - Sending an email\n"
            + "3 - Printing out all the recipients who have birthdays\n"
            + "4 - Printing out details of all the emails sent\n"
            + "5 - Printing out the number of recipient objects in the
application");

        int option = scanner0.nextInt();

        ArrayList<String> birthday_list = new ArrayList<String>();
        birthday_list.add("Personal: sunil,sunila,sunil@gmail.com,2022/08/09");
        birthday_list.add("Personal: sunil,sunila,sunil@gmail.com,2022/08/22");
        birthday_list.add("Personal: sunil,sunila,sunil@gmail.com,2022/08/22");
        birthday_list.add("Office_friend:
kamqweal,kamal@gmail.com,clerk,2000/08/13");
        int s = birthday_list.size();
        ArrayList<Personal> pers_bday = new ArrayList<Personal>();
        ArrayList<OfficeFriend> ofp_bday= new ArrayList<OfficeFriend>();
        ArrayList<Official> off_bday = new ArrayList<Official>();

```

```

        ArrayList<PersonalMail> mail_list1 = new ArrayList<PersonalMail>();
        ArrayList<GeneralMail> mail_list2 = new ArrayList<GeneralMail>();
        ArrayList<OfficeFriendMail> mail_list3 = new
ArrayList<OfficeFriendMail>();

```

```

        for(String item : birthday_list){
            String first = item.substring(0,10);
            if( first.equals("Official: ")){
                int l = item.length();
                String[] String_list = item.substring(10,l).split(",");
                Official append_item = new Official(item.substring(0,7),
String_list[0], String_list[1], String_list[2]);
                off_bday.add(append_item);
            }else if(item.substring(0,15).equals("Office_friend: ")){
                int l1 = item.length();
                String[] String_list1 = item.substring(15,l1).split(",");
                OfficeFriend append_item = new
OfficeFriend(item.substring(0,15), String_list1[0], String_list1[1],
String_list1[2], String_list1[3]);
                ofp_bday.add(append_item);
            }else if(item.substring(0,10).equals("Personal: ")){
                int l2 = item.length();
                String[] String_list2 = item.substring(10,l2).split(",");
                Personal append_item = new Personal(item.substring(0,8),
String_list2[0], String_list2[2], String_list2[1], String_list2[3]);
                pers_bday.add(append_item);
            }else{
                System.out.println("error in intialize class logic");
            }
        }
    }

```

```

        System.out.println("Please wait till we send the birthday mails");

```

```

        for(Personal item1 : pers_bday){
            if(item1.birthday.substring(4,10).equals(date_now.substring(4,
10))){
                item1.SendBirthdayMail();
                mail_list1.add(new PersonalMail(item1.email_address,"Birthday
Wish" , "hugs and love on your birthday. Navinda",
date_now,item1.nickname,item1.birthday));
            }
        }

        for(OfficeFriend item2 : ofp_bday){
            if(item2.birthday.substring(4,10).equals(date_now.substring(4,10)))
{
                item2.SendBirthdayMail();
                mail_list3.add(new
OfficeFriendMail(item2.email_address,"Birthday Wish" , "hugs and love on your

```

```

        birthday. Navinda", date_now,item2.position,item2.birthday));
    }
}

```

```

ArrayList<GeneralMail> s11 = new ArrayList<GeneralMail>();
ArrayList<PersonalMail> s12 = new ArrayList<PersonalMail>();
ArrayList<OfficeFriendMail> s13 = new ArrayList<OfficeFriendMail>();

```

```

PersonalSerialize.serialize_start(mail_list1);
OfficeFriendSerialize.serialize_start(mail_list3);

```

```

switch(option){
    case 1:
        // input format - Official: nimal,nimal@gmail.com,ceo
        // Use a single input to get all the details of a recipient
        // code to add a new recipient1
        // store details in clientList.txt file
        // Hint: use methods for reading and writing files
        Scanner scanner2 = new Scanner(System.in);
        System.out.println("write your new record \ninput format -
Official: nimal,nimal@gmail.com,ceo");
        String entry = scanner2.nextLine();
        FileHandle.WriteFile(entry);
        scanner2.close();
        break;
    case 2:
        // input format - email, subject, content
        // code to send an email
        Scanner scanner3 = new Scanner(System.in);
        System.out.println("Input the Receptient Email");
        String receipient = scanner3.nextLine();
        System.out.println("Input the Subject");
        String subject = scanner3.nextLine();
        System.out.println("Input the Content");
        String content = scanner3.nextLine();
        SendEmailTLS.sendmail(receipient,subject,content);
        mail_list2.add(new GeneralMail(receipient, subject, content,
date_now));

        General_serialize.serialize_start(mail_list2);
        scanner3.close();
        break;
    case 3:
        // input format - yyyy/MM/dd (ex: 2018/09/17)
        // code to print recipients who have birthdays on the given
date

        Scanner scanner4 = new Scanner(System.in);
        String day = scanner4.nextLine();

        for(int i = 0; i < s;i++){
            int size = birthday_list.get(i).length();
            String compare = birthday_list.get(i).substring(size-10);
            if(compare.equals(day)){
                System.out.println(birthday_list.get(i));
            }
        }
    }
}

```

```

    }

    scanner4.close();
    break;

    case 4:
        // input format - yyyy/MM/dd (ex: 2018/09/17)
        // code to print the details of all the emails sent on the
input date
        s11 = General_serialize.deserialize_start();
        s12 = PersonalSerialize.deserialize_start();
        s13= OfficeFriendSerialize.deserialize_start();
        System.out.println("Input the date you want to check\ninput
format - yyyy/MM/dd");
        Scanner scanner5 = new Scanner(System.in);
        String input_date = scanner5.nextLine();
        for (GeneralMail item : s11){
            if (item.date.equals(input_date)){
                System.out.println(item.receipient+ "\n"+item.subject
+" \n"+ item.content);
            }
        }

        for( PersonalMail item : s12){
            if (item.date.equals(input_date)){
                System.out.println(item.receipient+ "\
n"+item.subject +"\n"+ item.content);
            }
        }

        for (OfficeFriendMail item : s13){
            if (item.date.equals(input_date)){
                System.out.println(item.receipient+ "\
n"+item.subject +"\n"+ item.content);
            }
        }

        scanner5.close();
        break;

    case 5:
        // code to print the number of recipient objects in the
application
        int x =InitializeClass.func();
        System.out.println(x);
        break;

}

scanner0.close();

// start email client
// code to create objects for each recipient in clientList.txt
// use necessary variables, methods and classes

```

```
    }  
}
```

// create more classes needed for the implementation (remove the public access modifier from classes when you submit your code)

```
//SendEmailTLS
```

```
//fzwykuuekmjolcmz app password
```

```
class SendEmailTLS {  
    public static void sendmail(String receipient, String subject, String content) {  
        final String username = "navindaoop@gmail.com";  
        final String password = "fzwykuuekmjolcmz";  
  
        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 {  
            Message message = new MimeMessage(session);  
            message.setFrom(new InternetAddress("navindaoop@gmail.com"));
```

```

        message.setRecipients(
            Message.RecipientType.TO,
            InternetAddress.parse(receipient)
        );
        message.setSubject(subject);
        message.setText(content);

        Transport.send(message);

        System.out.println("Done");
    } catch (MessagingException e) {
        e.printStackTrace();
    }
}

//FileHandle

class FileHandle {
    static ArrayList<String> file_objects = new ArrayList<String>();
    public static void WriteFile(String entry) {
        try {
            FileWriter writer = new FileWriter("clientList.txt",true);

            writer.write(entry);
            writer.write("\n");
            writer.close();
        } catch (IOException e) {
            e.printStackTrace();
            System.out.println("Error in FileHandle Class");
        }
    }
    public static ArrayList<String> ReadFile() {
        try {
            File myObj = new File("clientList.txt");
            Scanner myReader = new Scanner(myObj);
            while (myReader.hasNextLine()) {
                String data = myReader.nextLine();
                file_objects.add(data);
            }
            myReader.close();

        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
        return file_objects;
    }
}

//ReceipientSuperClass

abstract class ReceipientSuperClass implements Serializable{
    protected String status;

```

```

        protected String name_of_receipient;
        protected String email_address;

    }

    //Birthday_interface

    interface Birthday_Interface {
        public void SendBirthdayMail();
    }

    //

    class Personal extends ReceipientSuperClass implements Birthday_Interface {
        public String nickname;
        public String birthday;

        public Personal(String status,String name,String mail,String nickname, String
        birthday){
            this.status = status;
            this.nickname = nickname;
            this.name_of_receipient = name;
            this.email_address = mail;
            this.birthday = birthday;

        }

        public void SendBirthdayMail(){
            SendEmailTLS.sendmail(this.email_address,"Birthday Wish","hugs and love on
            your birthday. Navinda");
        }
    }

    class Official extends ReceipientSuperClass{
        public String position;

        public Official(String status,String name,String mail,String position){
            this.position = position;
            this.name_of_receipient = name;
            this.status = status;
            this.email_address = mail;

        }

    }

    class OfficeFriend extends ReceipientSuperClass implements Birthday_Interface {
        public String position;
        public String birthday;

        public OfficeFriend(String status,String name,String mail,String position,
        String birthday){
            this.position = position;
            this.name_of_receipient = name;

```

```

        this.status = status;
        this.email_address = mail;
        this.birthday = birthday;
    }

    public void SendBirthdayMail(){
        SendEmailTLS.sendmail(this.email_address,"Birthday Wish","Wish you a Happy
        Birthday. Navinda");
    }
}

```

```

class InitializeClass {
    static ArrayList<ReceipientSuperClass> int_objects = new
    ArrayList<ReceipientSuperClass>();
    static ArrayList<Personal> pers_objects = new ArrayList<Personal>();
    static ArrayList<OfficeFriend> ofp_objects = new ArrayList<OfficeFriend>();
    static ArrayList<Official> off_objects = new ArrayList<Official>();
    static ArrayList<String> pers_strings = new ArrayList<String>();
    public static int func(){
        ArrayList<String> st0 = FileHandle.ReadFile();
        for(String item : st0){
            String first = item.substring(0,10);
            if( first.equals("Official: ")){
                int l = item.length();
                String[] String_list = item.substring(9,l).split(",");
                Official append_item = new Official(item.substring(0,7),
                String_list[0], String_list[1], String_list[2]);
                int_objects.add(append_item);
                off_objects.add(append_item);
            }else if(item.substring(0,15).equals("Office_friend: ")){
                int l1 = item.length();
                String[] String_list1 = item.substring(14,l1).split(",");
                OfficeFriend append_item = new OfficeFriend(item.substring(0,12),
                String_list1[0], String_list1[1], String_list1[2], String_list1[3]);
                int_objects.add(append_item);
                ofp_objects.add(append_item);
            }else if(item.substring(0,10).equals("Personal: ")){
                int l2 = item.length();
                String[] String_list2 = item.substring(10,l2).split(",");
                Personal append_item = new Personal(item.substring(0,8),
                String_list2[0], String_list2[2], String_list2[1], String_list2[3]);
                int_objects.add(append_item);
                pers_objects.add(append_item);
                pers_strings.add(item);
            }else{
                System.out.println("error in intialize class logic");
            }
        }
        return int_objects.size();
    }
}

```

```

class MailClass implements Serializable {

```



```

    protected String receipient;
    protected String content;
    protected String subject;
    protected String date;
}

class GeneralMail extends MailClass implements Serializable{

    public GeneralMail(String receipient, String subject, String content, String
date){
        this.receipient = receipient;
        this.content = content;
        this.subject = subject;
        this.date = date;
    }
}

class OfficeFriendMail extends MailClass {

    public String position;
    public String birthday;

    public OfficeFriendMail(String receipient, String subject, String content,String
date,String position, String birthday){
        this.receipient = receipient;
        this.content = content;
        this.subject = subject;
        this.date = date;
        this.position = position;
        this.birthday = birthday;
    }
}

class PersonalMail extends MailClass implements Serializable {

    public String nickname;
    public String birthday;

    public PersonalMail(String receipient, String subject, String content,String
date,String nickname, String birthday){
        this.receipient = receipient;
        this.content = content;
        this.subject = subject;
        this.date = date;
    }
}

```

```

        this.nickname = nickname;
        this.birthday = birthday;
    }
}

```

```

class General_serialize{

```

```

    public static void serialize_start( ArrayList<GeneralMail> obj0) {
        try {
            FileOutputStream fileOut =new FileOutputStream("GeneralMail.ser");
            ObjectOutputStream out = new ObjectOutputStream(fileOut);
            out.writeObject(obj0);
            out.close();
            fileOut.close();
            System.out.printf("Serialized data is saved in GeneralEmail.ser\n");
        } catch (IOException i) {
            i.printStackTrace();
        }
    }
}

```

```

    public static ArrayList<GeneralMail> deserialize_start() {
        ArrayList<GeneralMail> obj0 = new ArrayList<GeneralMail>();
        try {
            FileInputStream fileStream = new FileInputStream("GeneralMail.ser");
            ObjectInputStream os1 = new ObjectInputStream(fileStream);
            obj0 = (ArrayList) os1.readObject();
            os1.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return obj0;
    }
}

```

```

}

```

```

class OfficeFriendSerialize{

```

```

    public static void serialize_start( ArrayList<OfficeFriendMail> obj0) {
        try {
            FileOutputStream fileOut =new FileOutputStream("OfficeFriendMail.ser");
            ObjectOutputStream out = new ObjectOutputStream(fileOut);
            out.writeObject(obj0);
            out.close();
            fileOut.close();
            System.out.printf("Serialized data is saved in OfficeEmails.ser\n");
        } catch (IOException i) {
            i.printStackTrace();
        }
    }
}

```

```

    public static ArrayList<OfficeFriendMail> deserialize_start() {
        ArrayList<OfficeFriendMail> obj0 = new ArrayList<OfficeFriendMail>();
    }
}

```

```

        try {
            FileInputStream fileStream = new
FileInputStream("OfficeFriendMail.ser");
            ObjectInputStream os1 = new ObjectInputStream(fileStream);
            obj0 = (ArrayList) os1.readObject();
            os1.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return obj0;
    }}

class PersonalSerialize{

    public static void serialize_start( ArrayList<PersonalMail> obj0) {
        try {
            FileOutputStream fileOut =new FileOutputStream("PersonalMail.ser");
            ObjectOutputStream out = new ObjectOutputStream(fileOut);
            out.writeObject(obj0);
            out.close();
            fileOut.close();
            System.out.printf("Serialized data is saved in PersonalMail.ser\n");
        } catch (IOException i) {
            i.printStackTrace();
        }
    }

    public static ArrayList<PersonalMail> deserialize_start() {
        ArrayList<PersonalMail> obj0 = new ArrayList<PersonalMail>();
        try {
            FileInputStream fileStream = new FileInputStream("PersonalMail.ser");
            ObjectInputStream os1 = new ObjectInputStream(fileStream);
            obj0 = (ArrayList) os1.readObject();
            os1.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return obj0;
    }

}

```