package Impact;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.time.LocalDate;

import java.time.Period;

import java.util.ArrayList;

import java.util.Calendar;

import java.util.Date;

import java.util.List;

import java.util.Scanner;

/\*\*

\* Impact main source: create a SNS called Impact to connect individual, organization and volunteer.

\* @author Aimi Ross

\* @version 1.0.0

\* @since 10/09/2019

\* \*\*/

public class Impact {

public static void main(String[] args) throws ParseException {

Scanner in = new Scanner(System.in);

String pickedAction,pickedOrg,donationAmount, pickedVLT,from,till;

int indx = -1;

List<User> Accounts = new ArrayList<>();

List<Post> Posts = new ArrayList<>();

Accounts.add(new Individual("Aimi", "touyamaaaa@geee", "eye-me", "ughCOMPSCI!", "Hello world!", "09/27/2017", " "));

Accounts.add(new Individual("Vanessa", "VANNNNNNEEEESSA!@geee", "ImSoFunny", "VAN999!", "Love Java!", "09/07/1996", "Union Bank"));

Accounts.add(new Individual("Tristen", "teeen@geee", "Tt", "Kendama!", "Do you want to play Kendama!?", "02/14/1996", "Union Bank"));

Accounts.add(new Individual("Ruben", "rrrru@geee", "TightDUDE", "YO!", "I surf, dance and code.. Tight!", "01/01/2004", "Union Bank"));

Accounts.add(new Organization("SNUG", "touyamaaaa@geee", "wahahaha", "omgCOMPSCI!", "Hello world!", "Orizaba Ave", "9086425", 1234235, "Unnnion bank"));

Accounts.add(new Organization("NIGHT", "VANNNNNNEEEESSA!@geee", "HHHEELLLYA", "VAN999!", "Love Java!", "OKINAWA sdgga", "987632", 43532462, "westerrrrrrn bank"));

Accounts.add(new Volunteer("gasdg", "teeen@geee", "KENKENpo!", "Kendannnnnma!", "Do you want to play Kendama!?", "10/20/2019", "10/31/2019", "OOOOkinawa Yomitan", "7842524"));

Accounts.add(new Volunteer("RUUUUU", "rrrru@geee", "surf?", "YO maaan!", "I surf, dance and code.. Tight!", "08/20/2019", "08/31/2019", "OOOOkinawa Iramina", "237658"));

MainMenu(); // ++++++++++++++++++++++++++++++++Main menu++++++++++++++++++++++++++++++++

pickedAction = in.nextLine();

while (!pickedAction.equalsIgnoreCase("A") && !pickedAction.equalsIgnoreCase("C") && !pickedAction.equalsIgnoreCase("D")) {

System.out.println("Invalid input, please pick from below..\n");

MainMenu();

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("A")) {

UserMenu(); // ++++++++++++++++++++++++++++++++User menu++++++++++++++++++++++++++++++++

String userType = in.nextLine();

while (!userType.equals("1") && !userType.equals("2") && !userType.equals("3") ) {

System.out.println("Invalid input, please pick 1-3.");

UserMenu();

userType = in.nextLine();

}

if(userType.equals("1")) { //1111111111111111111111entered Individual Login page111111111111111111111111111111111

System.out.println("Individual Menu=========================================");

indx = findUser(Accounts); // \*\*\*\*\*\*\*\*\*\* ACTIVE USER'S INDEX NUMBER \*\*\*\*\*\*\*\*\*\*

Accounts.get(indx).displayMenu();

pickedAction = in.nextLine();

while(!pickedAction.equalsIgnoreCase("P") && !pickedAction.equalsIgnoreCase("D") && !pickedAction.equalsIgnoreCase("R")) {

System.out.println("Invalid input, please enter P(post), D(donate), or R(register)..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("P")) {

System.out.println("Do you want to add an image? Y/N: ");

pickedAction = in.next();

while(!pickedAction.equalsIgnoreCase("Y") && !pickedAction.equalsIgnoreCase("N")) {

System.out.println("Invalid input, please enter Y(yes)or N(no)..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("N"))

Posts.add(posting(indx));

else

Posts.add(postingWITHimage(indx));

}

else if(pickedAction.equalsIgnoreCase("D")) {

if(getAge(((Individual) Accounts.get(indx)).getDOB())<18)

System.out.println("We're sorry, donation is allowed over 18 years old. \nBut Thank you so much for your consideration!");

else {

for (int i = 0; i < Accounts.size(); i++)

if(Accounts.get(i).getClassName().equals("Organization")) {

System.out.println(Accounts.get(i).getUserName());

}

System.out.println("Thank you for your consideration! \nLet's pick which organization to donate from above: ");

pickedOrg = in.nextLine();

System.out.println("Please enter your desiered donation amount: ");

donationAmount = in.nextLine();

Posts.add(new Donation(indx, pickedOrg, donationAmount));

}

}else if(pickedAction.equalsIgnoreCase("R")){

System.out.println("Okay! let's register to an event! The dates are showing as mm.dd.yyyy");

for (int i = 0; i < Accounts.size(); i++)

if(Accounts.get(i).getClassName().equals("Volunteer")) {

System.out.println(((Volunteer) Accounts.get(i)).displayDetail());

}

System.out.print("Which one would you like to join? ");

pickedVLT = in.nextLine();

for (int i = 0; i < Accounts.size(); i++)

if(Accounts.get(i).getUserName().equals(pickedVLT)) {

Posts.add(new Events(indx, pickedVLT));

}

}

// Posts.add(new Post(3, "Yahoo!"));

// Posts.add(new PostWITHimage(1, "Night Night!", "Flower", "desktoooop"));

// Posts.add(new Events(2, "surf?"));

// for(int i = 0; i < Posts.size(); i++) {

// System.out.println(Accounts.get(Posts.get(i).getIndex()).username + Posts.get(i).toString());

// }

else {

// create an exit function

}

}else if(userType.equals("2")) { //222222222222222222222222Organization222222222222222222222222

System.out.println("Organization Menu=========================================");

indx = findUser(Accounts); // \*\*\*\*\*\*\*\*\*\* ACTIVE USER'S INDEX NUMBER \*\*\*\*\*\*\*\*\*\*

Accounts.get(indx).displayMenu();

System.out.print("Enter your choice: ");

pickedAction = in.nextLine();

while(!pickedAction.equalsIgnoreCase("P") && !pickedAction.equalsIgnoreCase("E")) {

System.out.println("Invalid input, please enter P(post) or E(exit)..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("P")) {

System.out.println("Do you want to add an image? Y/N: ");

pickedAction = in.nextLine();

while(!pickedAction.equalsIgnoreCase("Y") && !pickedAction.equalsIgnoreCase("N")) {

System.out.println("Invalid input, please enter Y(yes)or N(no)..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("N"))

Posts.add(posting(indx));

else

Posts.add(postingWITHimage(indx));

}

else {

// create an exit function

}

}else {// 333333333333333333333333333Volunteer3333333333333333333333333333

System.out.println("Volunteer Menu=========================================");

indx = findUser(Accounts); // \*\*\*\*\*\*\*\*\*\* ACTIVE USER'S INDEX NUMBER \*\*\*\*\*\*\*\*\*\*

Accounts.get(indx).displayMenu();

System.out.print("Enter your choice: ");

pickedAction = in.nextLine();

while(!pickedAction.equalsIgnoreCase("P") && !pickedAction.equalsIgnoreCase("M") && !pickedAction.equalsIgnoreCase("E")) {

System.out.println("Invalid input, please enter P(post), M(modify Date/Time/Location)or E(exit)..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("P")) {

System.out.println("Do you want to add an image? Y/N: ");

pickedAction = in.nextLine();

while(!pickedAction.equalsIgnoreCase("Y") && !pickedAction.equalsIgnoreCase("N")) {

System.out.println("Invalid input, please enter Y(yes)or N(no)..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("N"))

Posts.add(posting(indx));

else

Posts.add(postingWITHimage(indx));

}else if(pickedAction.equalsIgnoreCase("M")) {

System.out.print("From? ");

from = in.nextLine();

System.out.print("Till? ");

till=in.nextLine();

((Volunteer) Accounts.get(indx)).setEventDate(from, till);

System.out.println("Now, " + Accounts.get(indx).getUserName() + "'s date has changed as " + from + " - " + till);

}else {

// create an exit function

}

}

}else if(pickedAction.equalsIgnoreCase("C")) {

System.out.println("Okay, let's get ready to become a member of Impact!"); // ======create an account=========

System.out.print("Please select your choice of account from I (individual), O (organization), V (vlunteer)");

pickedAction = in.nextLine();

while(!pickedAction.equalsIgnoreCase("I") && !pickedAction.equalsIgnoreCase("O") && !pickedAction.equalsIgnoreCase("V")) {

System.out.println("Invalid input, please enter I, O, or V..\n");

pickedAction = in.nextLine();

}

if(pickedAction.equalsIgnoreCase("I")) {

Accounts.add(createNewAccountIND());

System.out.println("Okay, your all set, " + Accounts.get(Accounts.size()-1).getUserName() + "!");

// we'll seeeeeeeeeeeeeeeeeeeee!!!! add all actions after creating donation and registers

}else if (pickedAction.equalsIgnoreCase("O")) {

Accounts.add(createNewAccountORG());

System.out.println("Okay, your all set, " + Accounts.get(Accounts.size()-1).getUserName() + "!");

// add all actions after creating modify an event.

}else {

Accounts.add(createNewAccountVLN());

System.out.println("Okay, your all set, " + Accounts.get(Accounts.size()-1).getUserName() + "!");

// add all actions after creating exist.

}

}else { // D)isplay

System.out.println("Suuuuure, I'll show you...");

for(int i = 0; i < Posts.size(); i++)

System.out.println(Accounts.get(Posts.get(i).getIndex()).username + " posted "+ Posts.get(i).toString());

}

in.close();

}

public static void MainMenu() {

System.out.println("Main Menu=========================================");

System.out.println("C)reate a new user");

System.out.println("A)ct as user");

System.out.println("D)isplay Feed");

}

public static void UserMenu() {

System.out.println("\nWelcome to the Impact!!!");

System.out.print("Please pick your account type from below.\n");

System.out.println("1: Individual");

System.out.println("2: Organization");

System.out.println("3: Volunteer");

}

public static int nameMatchinguuu(List a, String userName) {

int indx = 0;

while (indx != a.size())

if(!userName.equals(((User) a.get(indx)).getUserName()))

indx++;

else if (userName.equals(((User) a.get(indx)).getUserName()))

return indx;

return -1;

}

public static int pwMatchinguuu(List a, String pw) {

int indx = 0;

while (indx != a.size())

if(!pw.equals(((User) a.get(indx)).getPassword()))

indx++;

else if (pw.equals(((User) a.get(indx)).getPassword()))

return indx;

return -1;

}

public static int findUser(List a) {

Scanner in = new Scanner(System.in);

String userName;

String password;

System.out.print("Enter your user name: ");

userName = in.next();

nameMatchinguuu(a, userName);

while(nameMatchinguuu(a, userName) == -1) {

System.out.print("Data didn't match, please enter your user name: ");

userName = in.next();

nameMatchinguuu(a, userName);

}

System.out.print("Enter your password: ");

password = in.next();

pwMatchinguuu(a, password);

while(pwMatchinguuu(a, password) == -1) {

System.out.print("Data didn't match, please enter your password: ");

password = in.next();

pwMatchinguuu(a, password);

}

while (nameMatchinguuu(a, userName) != pwMatchinguuu(a, password)) {

System.out.print("Data didn't match, please enter your user name: ");

userName = in.next();

nameMatchinguuu(a, userName);

while(nameMatchinguuu(a, userName) == -1) {

System.out.print("Data didn't match, please enter your user name: ");

userName = in.next();

nameMatchinguuu(a, userName);

}

System.out.print("Data didn't match, please enter your password: ");

password = in.next();

pwMatchinguuu(a, password);

while(pwMatchinguuu(a, password) == -1) {

System.out.print("Data didn't match, please enter your password: ");

password = in.next();

pwMatchinguuu(a, password);

}

}

System.out.println("Hello " + userName + ", welcome back to Impact!");

return pwMatchinguuu(a, password);

}

public static Post posting(int a) {

String msg;

Scanner in = new Scanner(System.in);

System.out.print("Please enter your message: ");

msg = in.nextLine();

return new Post(a, msg);

}

public static Post postingWITHimage(int a) {

String msg, image, location, action;

Scanner in = new Scanner(System.in);

System.out.print("Please enter your message: ");

msg = in.nextLine();

System.out.println("Please enter the image name: ");

image = in.nextLine();

System.out.println("Please enter the location of the image: ");

location = in.nextLine();

return new PostWITHimage(a, msg, image, location);

}

public static User defaultInfo() {

String name, email, userName, password, bio;

Scanner in = new Scanner(System.in);

System.out.print("Please enter your name: ");

name = in.nextLine();

System.out.print("Please enter your email address: ");

email = in.nextLine();

System.out.print("Please set your user name: ");

userName = in.nextLine();

System.out.print("Please set your password: ");

password = in.nextLine();

System.out.print("Please set your bio: ");

bio = in.nextLine();

return new User(name, email, userName, password, bio);

}

public static User createNewAccountIND() throws ParseException {

String bio, bankInfo, dob;

Scanner in = new Scanner(System.in);

User ind = defaultInfo();

System.out.print("Please enter your birthday in MM/dd/yyy format: ");

dob = in.nextLine();

//System.out.println(getAge(dob));

if(getAge(dob)<18)

bankInfo = " ";

else {

System.out.print("Please enter your bank name: ");

bankInfo = in.nextLine();

}

return new Individual(ind.getName(), ind.getEmail(), ind.getUserName(), ind.getPassword(), ind.getBio(), dob, bankInfo);

}

public static User createNewAccountORG() {

String address, bankInfo, phone;

int taxID;

Scanner in = new Scanner(System.in);

User ind = defaultInfo();

System.out.print("Please enter your address: ");

address = in.nextLine();

System.out.print("Please enter your bank name: ");

bankInfo = in.nextLine();

System.out.print("Please enter your phone number: ");

phone = in.nextLine();

System.out.print("Please enter your tax ID: ");

taxID = in.nextInt();

return new Organization(ind.getName(), ind.getEmail(), ind.getUserName(), ind.getPassword(), ind.getBio(), address, phone, taxID, bankInfo);

}

public static User createNewAccountVLN() {

String address,phone, from, till;

Scanner in = new Scanner(System.in);

User ind = defaultInfo();

System.out.print("Please enter your address: ");

address = in.nextLine();

System.out.print("Please enter your phone number: ");

phone = in.nextLine();

System.out.print("Please enter the event starting date in mm/dd/yyy form: ");

from = in.nextLine();

System.out.print("Please enter the event ending date in mm/dd/yyy form: ");

till = in.nextLine();

return new Volunteer(ind.getName(), ind.getEmail(), ind.getUserName(), ind.getPassword(), ind.getBio(), from,till, address, phone);

}

public static int getAge(String dateOfBirth) throws ParseException {

SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");

Date d = sdf.parse(dateOfBirth);

Calendar c = Calendar.getInstance();

c.setTime(d);

int year = c.get(Calendar.YEAR);

int month = c.get(Calendar.MONTH) + 1;

int date = c.get(Calendar.DATE);

LocalDate l1 = LocalDate.of(year, month, date);

LocalDate now1 = LocalDate.now();

Period diff1 = Period.between(l1, now1);

return diff1.getYears();

}

public static int getDays(String day) throws ParseException {

SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");

Date d = sdf.parse(day);

Calendar c = Calendar.getInstance();

c.setTime(d);

int year = c.get(Calendar.YEAR);

int month = c.get(Calendar.MONTH) + 1;

int date = c.get(Calendar.DATE);

LocalDate l1 = LocalDate.of(year, month, date);

LocalDate now1 = LocalDate.now();

Period diff1 = Period.between(l1, now1);

return diff1.getDays();

}

public static void wholeIMPACT() {

}

}