//Zara Luigi

//Classe 3E

/\*

Modificare l'esercizio visto in classe in modo da inserire un "contatto" solamente se non è

già presente all'interno della collezione, controllare il cognome e il nome.

\*/

//importazione librerie

import static tools.utility.\*;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

//dichiarazione scanner

Scanner keyboard = new Scanner(System.*in*);

//dichiarazione e inizializzazione delle variabili

final int nMax=3;

int contrattiVenduti = 0;

Contatto[] gestore = new Contatto[nMax];

boolean fine = true;

boolean Sitel = true;

//menù

String[] operazioni = {"VODAFONE", "[1] Inserimento",

"[2] Visualizzazione",

"[3] Ricerca",

"[4] Modifica contratto",

"[5] Modifica numero telefono",

"[6] Cancella contatto inserendo il n. di telefono",

"[7] Cancella contatto inserendo nome/cognome",

"[8] Fine"

};

//do-while

do {

switch(*menu*(operazioni, keyboard)) {

case 1:

if(contrattiVenduti < nMax){

//firma contratto

gestore[contrattiVenduti]=*leggiPersona*(Sitel,keyboard, gestore, contrattiVenduti);

contrattiVenduti++;

}else{

System.*out*.println("Non ci sono più contratti da vendere");

}

break;

case 2:

*visualizza*(gestore, contrattiVenduti);

break;

case 3:

if(contrattiVenduti !=0){

//Ci sono contratti venduti

//lettura, ricerca, visualizzazione

if(*ricerca*(gestore, *leggiPersona*(Sitel, keyboard, gestore, contrattiVenduti), contrattiVenduti)){

System.*out*.println("Il contatto esiste");

} else {

System.*out*.println("Il contatto non esiste");

}

}else{

System.*out*.println("Non sono ancora presenti contratti venduti");

}

break;

case 4:

*ModificaContratto*(Sitel, keyboard, gestore, contrattiVenduti);

break;

case 5:

if (contrattiVenduti < nMax) {

// firma contratto

*NuovoTelefono*(Sitel, keyboard, gestore, contrattiVenduti);

contrattiVenduti++;

} else {

System.*out*.println("Non ci sono più contratti da vendere");

}

*visualizza1*(gestore, contrattiVenduti);

break;

case 6:

*cancellaContattoPerTelefono*(gestore, contrattiVenduti, keyboard);

break;

case 7:

*cancellaContattoPerNomeCognome*(gestore, contrattiVenduti, keyboard);

break;

default: fine = false;

}

}while(fine);

}

//metodo ricerca

private static boolean ricerca (Contatto[] gestore, Contatto contatto, int contrattiVenduti){

//Controllo se il nome e il cognome del contatto e ugale al nome e cogome del gestore

boolean ricerca = false;

for(int i = 0; i<contrattiVenduti; i++){

if(contatto.nome.equals(gestore[i].nome) && contatto.cognome.equals(gestore[i].cognome)){

ricerca = true;

}

}

return ricerca;

}

//metodo visualizza

private static void visualizza(Contatto [] gestore, int contrattiVenduti){

for(int i=0 ; i<contrattiVenduti; i++){

System.*out*.println(gestore[i].stampa());

}

}

//metodo conta contatti abitazione

private static int contaContattiAbitazione(Contatto [] gestore, int contrattiVenduti){

int contAbitazione=0;

for(int i=0; i<contrattiVenduti; i++){

if(gestore[i].tipo == tipoContratto.*abitazione*) {

contAbitazione++;

}

}

return contAbitazione;

}

//metodo leggiPersona

private static Contatto leggiPersona(boolean Sitel, Scanner keyboard, Contatto[] gestore, int contrattiVenduti){

//istanzio un oggetto di tipo contatto

Contatto persona = new Contatto();

//vettore di stringhe

String[] tipoC = {"telefono", "[1] - abitazione","[2] - cellulare", "[3] -aziendale"};

//dichiarazione variabile per la verifica del contatto

boolean contattoGiaEsistente;

do {

contattoGiaEsistente=true;

//chiedo all'utente di fornire in ingresso le seguenti informazioni:

System.*out*.println("Inserisci il tuo nome: ");

persona.nome = keyboard.nextLine();

System.*out*.println("Inserisci il tuo cognome: ");

persona.cognome = keyboard.nextLine();

//controllo che l'utente non sia già registrato

for (int i = 0; i < contrattiVenduti; i++) {

if (persona.nome.equals(gestore[i].nome) && persona.cognome.equals(gestore[i].cognome)) {

System.*out*.println("L'utente inserito è già in possesso di un account. Registrane uno nuovo:");

contattoGiaEsistente = false;

break;

}

}

} while (contattoGiaEsistente==false);

//chiedo all'utente di insereire il suo numero di telefono

System.*out*.println("Inserisci il tuo numero di telefono: ");

if(Sitel){//se Sitel è true:

//scanner numero telefono

persona.telefono = keyboard.nextLine();

//switch-case

switch(*menu*(tipoC, keyboard)){

case 1 -> persona.tipo = tipoContratto.*abitazione*;

case 2 -> persona.tipo = tipoContratto.*cellulare*;

default -> persona.tipo = tipoContratto.*aziendale*;

}

}

return persona;

}

//metodo che cancella il contatto inserendo il numero di telefono

private static void cancellaContattoPerTelefono(Contatto[] gestore, int contrattiVenduti, Scanner keyboard) {

//chiedo all'utente di inserire il numero di telefono del contatto da cancellare

System.*out*.println("Inserisci il numero di telefono del contatto da cancellare: ");

String telefonoDaCancellare = keyboard.nextLine();

boolean contattoTrovato = false;

for (int i = 0; i < contrattiVenduti; i++) {

if (gestore[i].telefono.equals(telefonoDaCancellare)) {

// Se il numero di telefono corrisponde, cancella il contatto

for (int j = i; j < contrattiVenduti - 1; j++) {

gestore[j] = gestore[j + 1];

}

contrattiVenduti--;

System.*out*.println("Contatto cancellato con successo.");

contattoTrovato = true;

break;

}

}

if (!contattoTrovato) {

System.*out*.println("Contatto non trovato.");

}

}

//metodo che cancella il contatto inserendo il nome e il cognome

private static void cancellaContattoPerNomeCognome(Contatto[] gestore, int contrattiVenduti, Scanner keyboard) {

//chiedo all'utente di inserire il nome del contatto da cancellare

System.*out*.println("Inserisci il nome del contatto da cancellare: ");

String nomeDaCancellare = keyboard.nextLine();

//chiedo all'utente di inserire il cognome del contatto da cancellare

System.*out*.println("Inserisci il cognome del contatto da cancellare: ");

String cognomeDaCancellare = keyboard.nextLine();

boolean contattoTrovato = false;

for (int i = 0; i < contrattiVenduti; i++) {

if (gestore[i].nome.equalsIgnoreCase(nomeDaCancellare) && gestore[i].cognome.equalsIgnoreCase(cognomeDaCancellare)) {

// Se il nome e cognome corrispondono, cancella il contatto

for (int j = i; j < contrattiVenduti - 1; j++) {

gestore[j] = gestore[j + 1];

}

contrattiVenduti--;

System.*out*.println("Contatto cancellato con successo.");

contattoTrovato = true;

break;

}

}

if (!contattoTrovato) {

System.*out*.println("Contatto non trovato.");

}

}

//metodo cambia numero di telefono

private static void NuovoTelefono(boolean Sitel, Scanner keyboard, Contatto[] gestore, int contrattiVenduti){

//dichiarazione e inizializzazione delle variabili

boolean contattoTrovato = false;

//chiedo all'utente di inserire il cognome del contatto di cui vuole modificarne il numero di telefono

System.*out*.println("Inserisci il cognome del contatto di cui vuoi modificarne il numero di telefono: ");

String cognomeDaModificare = keyboard.nextLine();

//chiedo all'utente di inserire il nome del contatto di cui vuole modificarne il numero di telefono

System.*out*.println("Inserisci il nome del contatto di cui vuoi modificarne il numero di telefono: ");

String nomeDaModificare = keyboard.nextLine();

//Cerca il contatto all'interno della collezione del file contatto

for (int i = 0; i < contrattiVenduti; i++) {

if (nomeDaModificare.equalsIgnoreCase(gestore[i].nome) && cognomeDaModificare.equalsIgnoreCase(gestore[i].cognome)) {

System.*out*.println("Contatto trovato.");

System.*out*.println("Inserisci il nuovo numero di telefono: ");

String nuovoNumero = keyboard.nextLine();

gestore[i].telefono = nuovoNumero;

System.*out*.println("Numero di telefono aggiornato con successo per il contatto: " + nomeDaModificare + " " + cognomeDaModificare);

contattoTrovato = true;

break;

}

}

//controllo che il contatto di cui si vuole modificare il numero esista

if (!contattoTrovato) {

do{

System.*out*.println("Nome o cognome errati!");

System.*out*.println("Reinserisci il cognome: ");

cognomeDaModificare = keyboard.nextLine();

System.*out*.println("Reinserisci il nome: ");

nomeDaModificare = keyboard.nextLine();

}while(contattoTrovato);

}

}

//metodo visualizza nuovo numero di telefono

private static void visualizza1(Contatto[] gestore, int contrattiVenduti){

for(int i=0 ; i<=contrattiVenduti; i++){

System.*out*.println(gestore[i].stampa());

}

}

//metodo per la modifica del ontratto

private static void ModificaContratto(boolean Sitel, Scanner keyboard, Contatto[] gestore, int contrattiVenduti) {

//dichiarazione e inizializzazione delle variabili

boolean contattoTrovato = false;

//istanzio un oggetto di tipo contatto

Contatto persona = new Contatto();

//vettore di stringhe

String[] tipoC = {"telefono", "[1] - abitazione","[2] - cellulare", "[3] -aziendale"};

//chiedo all'utente di inserire il cognome del contatto di cui vuole modificarne il numero di telefono

System.*out*.println("Inserisci il cognome del contatto di cui vuoi modificare il contratto: ");

String cognomeDaModificare = keyboard.nextLine();

//chiedo all'utente di inserire il nome del contatto di cui vuole modificarne il numero di telefono

System.*out*.println("Inserisci il nome del contatto di cui vuoi modificare il contratto: ");

String nomeDaModificare = keyboard.nextLine();

//Cerca il contatto all'interno della collezione del file contatto

for (int i = 0; i < contrattiVenduti; i++) {

if (nomeDaModificare.equalsIgnoreCase(gestore[i].nome) && cognomeDaModificare.equalsIgnoreCase(gestore[i].cognome)) {

System.*out*.println("Contatto trovato.");

if(Sitel){//se Sitel è true:

//switch-case

switch(*menu*(tipoC, keyboard)){

case 1 -> persona.tipo = tipoContratto.*abitazione*;

case 2 -> persona.tipo = tipoContratto.*cellulare*;

default -> persona.tipo = tipoContratto.*aziendale*;

}

}

System.*out*.println("Contratto aggiornato con successo per il contatto: " + nomeDaModificare + " " + cognomeDaModificare);

contattoTrovato = true;

break;

}

}

//controllo che il contatto di cui si vuole modificare il tipo di contratto esista

if (!contattoTrovato) {

do{

System.*out*.println("Nome o cognome errati!");

System.*out*.println("Reinserisci il cognome: ");

cognomeDaModificare = keyboard.nextLine();

System.*out*.println("Reinserisci il nome: ");

nomeDaModificare = keyboard.nextLine();

}while(contattoTrovato);

}

}

}