program db;

uses crt;

var id, menu, a\_bd, a\_bm, a\_by : integer;

st, a\_family, a\_name, a\_sex, a\_phone : string;

idf : text;

function ExistFile(FName:string):Boolean;

var

F:File;

begin

Assign(F,FName);

{$I-}

Reset(F,1);

{$I+}

If IoResult=0 then begin

ExistFile:=True;

Close(F);

end

else

ExistFile:=false;

end;

procedure create\_record(family : string; name : string; day : integer; month : integer; year : integer; sex : string; phone : string);

var f, idf : text;

tmp : string;

begin

AssignFile(f,'f.txt');

if (ExistFile('f.txt') = true) then Append(f) else Rewrite(f);

AssignFile(idf,'idf.txt');

reset(idf);

readln(idf,id);

close(idf);

rewrite(idf);

str(id+1,tmp);

writeln(idf,tmp);

close(idf);

str(id+1,tmp);

writeln(f,tmp);

id := id + 1;

writeln(f,family);

writeln(f,name);

str(day,tmp);

writeln(f,tmp);

str(month,tmp);

writeln(f,tmp);

str(year,tmp);

writeln(f,tmp);

writeln(f,sex);

writeln(f,phone);

Close(f);

end;

procedure sort\_records();

var sFamily,sName,sBd,sBm,sBy,sSex,sPhone : array of String;

n, i, j, tmpInt\_1, tmpInt\_2, tmpInt\_3 : integer;

tmp, tF, tN, tBd, tBm, tBy, tS, tP : string;

f, idfl : text;

begin

n := id;

SetLength(sFamily, n);

SetLength(sName, n);

SetLength(sBd, n);

SetLength(sBm, n);

SetLength(sBy, n);

SetLength(sSex, n);

SetLength(sPhone, n);

AssignFile(f,'f.txt');

if (ExistFile('f.txt') = false) then writeln('No file') else begin

reset(f);

for i := 0 to n-1 do begin

readln(f,tmp);

readln(f,sFamily[i]);

readln(f,sName[i]);

readln(f,sBd[i]);

readln(f,sBm[i]);

readln(f,sBy[i]);

readln(f,sSex[i]);

readln(f,sPhone[i]);

end;

close(f);

for j:=0 to n-2 do

for i:=0 to n-2-j do

if (upcase(sFamily[i]) > upcase(sFamily[i+1])) then

begin

tF := sFamily[i];

sFamily[i] := sFamily[i+1];

sFamily[i+1] := tF;

sName[i+1] := tN;

sBd[i+1] := tBd;

sBm[i+1] := tBm;

sBy[i+1] := tBy;

sSex[i+1] := tS;

sPhone[i+1] := tP;

end;

AssignFile(idfl,'idf.txt');

rewrite(idfl);

writeln(idfl,0);

close(idfl);

AssignFile(f,'f.txt');

rewrite(f);

close(f);

for i := 0 to n-1 do begin

val(sBd[i],tmpInt\_1);

val(sBm[i],tmpInt\_2);

val(sBy[i],tmpInt\_3);

create\_record(sFamily[i],sName[i],tmpInt\_1,tmpInt\_2,tmpInt\_3,sSex[i],sPhone[i]);

end;

clrscr;

end;

end;

procedure delete\_record(id\_remove : string);

var sID,sFamily,sName,sBd,sBm,sBy,sSex,sPhone : array of String;

n, i, z, tmpInt\_1, tmpInt\_2, tmpInt\_3 : integer;

tmp : string;

f, idfl : text;

begin

n := id;

SetLength(sID, n);

SetLength(sFamily, n);

SetLength(sName, n);

SetLength(sBd, n);

SetLength(sBm, n);

SetLength(sBy, n);

SetLength(sSex, n);

SetLength(sPhone, n);

AssignFile(f,'f.txt');

if (ExistFile('f.txt') = false) then writeln('No file') else begin

reset(f);

for i := 0 to n-1 do begin

readln(f,tmp);

if (tmp <> id\_remove) then begin

sID[i] := tmp;

readln(f,sFamily[i]);

readln(f,sName[i]);

readln(f,sBd[i]);

readln(f,sBm[i]);

readln(f,sBy[i]);

readln(f,sSex[i]);

readln(f,sPhone[i]);

end else begin

for z := 1 to 7 do

readln(f,tmp);

end;

end;

close(f);

AssignFile(idfl,'idf.txt');

rewrite(idfl);

writeln(idfl,0);

close(idfl);

AssignFile(f,'f.txt');

rewrite(f);

close(f);

for i := 0 to n-1 do begin

if (sFamily[i] <> '') then begin

val(sBd[i],tmpInt\_1);

val(sBm[i],tmpInt\_2);

val(sBy[i],tmpInt\_3);

create\_record(sFamily[i],sName[i],tmpInt\_1,tmpInt\_2,tmpInt\_3,sSex[i],sPhone[i]);

end;

end;

end;

end;

procedure show\_records(); forward;

procedure search(p : string);

var sID, sFamily,sName,sBd,sBm,sBy,sSex,sPhone : array of String;

n, i, menu : integer;

tmp, tmpID : string;

f : text;

flag : boolean;

begin

n := id;

SetLength(sID, n);

SetLength(sFamily, n);

SetLength(sName, n);

SetLength(sBd, n);

SetLength(sBm, n);

SetLength(sBy, n);

SetLength(sSex, n);

SetLength(sPhone, n);

AssignFile(f,'f.txt');

if (ExistFile('f.txt') = false) then writeln('No file') else begin

reset(f);

for i := 0 to n-1 do begin

readln(f,sID[i]);

readln(f,sFamily[i]);

readln(f,sName[i]);

readln(f,sBd[i]);

readln(f,sBm[i]);

readln(f,sBy[i]);

readln(f,sSex[i]);

readln(f,sPhone[i]);

end;

close(f);

case (p) of

'id' : begin

flag := false;

writeln('Enter ID: ');

readln(tmp);

for i := 0 to n-1 do begin

if (sID[i] = tmp) then begin

flag := true;

writeln('ID: ' + sID[i]);

writeln('Name: ' + sFamily[i] + ' ' + sName[i]);

writeln('Date of Birth: ' + sBd[i] + '.' + sBm[i] + '.' + sBy[i]);

writeln('Sex: ' + sSex[i]);

writeln('Phone: ' + sPhone[i]);

tmpID := sID[i];

writeln();

end else if ((i = n-1) and (flag = false)) then writeln('No such records');

end;

writeln('1. Back');

writeln('2. Delete record');

readln(menu);

case (menu) of

1 : begin

clrscr;

show\_records();

end;

2 : begin

delete\_record(tmpID);

clrscr;

show\_records();

end;

end;

end;

'family' : begin

flag := false;

writeln('Enter Second Name: ');

readln(tmp);

for i := 0 to n-1 do begin

if (upcase(sFamily[i]) = upcase(tmp)) then begin

flag := true;

writeln('ID: ' + sID[i]);

writeln('Name: ' + sFamily[i] + ' ' + sName[i]);

writeln('Date of Birth: ' + sBd[i] + '.' + sBm[i] + '.' + sBy[i]);

writeln('Sex: ' + sSex[i]);

writeln('Phone: ' + sPhone[i]);

writeln();

end else if ((i = n-1) and (flag = false)) then writeln('No such records');

end;

writeln('Press Enter to Back');

readln();

clrscr;

show\_records();

end;

'name' : begin

flag := false;

writeln('Enter Name: ');

readln(tmp);

for i := 0 to n-1 do begin

if (upcase(sName[i]) = upcase(tmp)) then begin

flag := true;

writeln('ID: ' + sID[i]);

writeln('Name: ' + sFamily[i] + ' ' + sName[i]);

writeln('Date of Birth: ' + sBd[i] + '.' + sBm[i] + '.' + sBy[i]);

writeln('Sex: ' + sSex[i]);

writeln('Phone: ' + sPhone[i]);

writeln();

end else if ((i = n-1) and (flag = false)) then writeln('No such records');

end;

writeln('Press Enter to Back');

readln();

clrscr;

show\_records();

end;

'by' : begin

flag := false;

writeln('Enter Birth Year: ');

readln(tmp);

for i := 0 to n-1 do begin

if (sBy[i] = tmp) then begin

flag := true;

writeln('ID: ' + sID[i]);

writeln('Name: ' + sFamily[i] + ' ' + sName[i]);

writeln('Date of Birth: ' + sBd[i] + '.' + sBm[i] + '.' + sBy[i]);

writeln('Sex: ' + sSex[i]);

writeln('Phone: ' + sPhone[i]);

writeln();

end else if ((i = n-1) and (flag = false)) then writeln('No such records');

end;

writeln('Press Enter to Back');

readln();

clrscr;

show\_records();

end;

'sex' : begin

flag := false;

writeln('Enter Sex (M/F): ');

readln(tmp);

for i := 0 to n-1 do begin

if (upcase(sSex[i]) = upcase(tmp)) then begin

flag := true;

writeln('ID: ' + sID[i]);

writeln('Name: ' + sFamily[i] + ' ' + sName[i]);

writeln('Date of Birth: ' + sBd[i] + '.' + sBm[i] + '.' + sBy[i]);

writeln('Sex: ' + sSex[i]);

writeln('Phone: ' + sPhone[i]);

writeln();

end else if ((i = n-1) and (flag = false)) then writeln('No such records');

end;

writeln('Press Enter to Back');

readln();

clrscr;

show\_records();

end;

end;

end;

end;

procedure show\_records();

var f : text;

tmp, tmp2 : string;

begin

AssignFile(f,'f.txt');

if (ExistFile('f.txt') = false) then writeln('No file') else begin

reset(f);

while not eof(f) do begin

readln(f,tmp);

writeln('ID: ' + tmp);

readln(f,tmp);

readln(f,tmp2);

writeln('Name: ' + tmp + ' ' + tmp2);

readln(f,tmp);

write('Date of Birth: ' + tmp + '.');

readln(f,tmp);

write(tmp + '.');

readln(f,tmp);

writeln(tmp);

readln(f,tmp);

writeln('Sex: ' + tmp);

readln(f,tmp);

writeln('Phone: ' + tmp);

writeln();

end;

close(f);

writeln();

writeln('1. Back');

writeln('2. Sort');

writeln('3. Search');

end;

readln(menu);

case (menu) of

1 : begin

clrscr;

writeln('1. Add Person');

writeln('2. Show data base');

readln(menu);

case (menu) of

1 : begin

clrscr;

writeln('Enter the second-name: ');

readln(a\_family);

writeln('Enter the name: ');

readln(a\_name);

writeln('Enter birthday: ');

readln(a\_bd);

writeln('Enter birthmonth: ');

readln(a\_bm);

writeln('Enter birthyear: ');

readln(a\_by);

writeln('Enter the sex: ');

readln(a\_sex);

writeln('Enter the phone number: ');

readln(a\_phone);

create\_record(a\_family,a\_name,a\_bd,a\_bm,a\_by,a\_sex,a\_phone);

writeln('The entry is created successfully. Press any key to watch it.');

readln();

clrscr;

show\_records();

end;

2 : begin

clrscr;

show\_records();

end;

else writeln('Try again.');

end;

end;

2: begin

clrscr;

sort\_records();

show\_records();

end;

3 : begin

clrscr;

writeln('Search by...');

writeln('1. ID');

writeln('2. Second Name');

writeln('3. Name');

writeln('4. Birth year');

writeln('5. Sex');

readln(menu);

clrscr;

case (menu) of

1 : search('id');

2 : search('family');

3 : search('name');

4 : search('by');

5 : search('sex')

else writeln('Try again');

end;

end else

writeln('Try again');

end;

end;

begin

AssignFile(idf,'idf.txt');

if (ExistFile('idf.txt') = true) then begin

reset(idf);

readln(idf,id);

end else begin

rewrite(idf);

writeln(idf,0);

end;

close(idf);

Window(1,1,80,50);

TextBackGround(0);

clrscr;

writeln('1. Add Person');

writeln('2. Show data base');

readln(menu);

case (menu) of

1 : begin

clrscr;

writeln('Enter the second-name: ');

readln(a\_family);

writeln('Enter the name: ');

readln(a\_name);

writeln('Enter birthday: ');

readln(a\_bd);

writeln('Enter birthmonth: ');

readln(a\_bm);

writeln('Enter birthyear: ');

readln(a\_by);

writeln('Enter the sex: ');

readln(a\_sex);

writeln('Enter the phone number: ');

readln(a\_phone);

create\_record(a\_family,a\_name,a\_bd,a\_bm,a\_by,a\_sex,a\_phone);

writeln('The entry is created successfully. Press any key to watch it.');

readln();

clrscr;

show\_records();

end;

2 : begin

clrscr;

show\_records();

end

else writeln('Try again.');

end;

readln();

end.