using System;

using System.Collections.Generic;

using System.Linq;

class PatientRecordSystem

{

static Dictionary<string, Dictionary<string, string>> departmentShelves = new Dictionary<string, Dictionary<string, string>>();

static void Main()

{

InitializeDepartments();

while (true)

{

Console.WriteLine("Patient Record Keeping System");

Console.WriteLine("1. Add Patient Record");

Console.WriteLine("2. Remove Patient Record");

Console.WriteLine("3. Sort Records in a Department");

Console.WriteLine("4. Search for a Patient");

Console.WriteLine("5. Exit");

Console.Write("\nEnter your choice: ");

int choice;

if (int.TryParse(Console.ReadLine(), out choice))

{

switch (choice)

{

case 1:

AddPatientRecord();

break;

case 2:

RemovePatientRecord();

break;

case 3:

SortRecords();

break;

case 4:

SearchForPatient();

break;

case 5:

Environment.Exit(0);

break;

default:

Console.WriteLine("Invalid choice. Please try again.");

break;

}

}

else

{

Console.WriteLine("Invalid choice. Please enter a number.");

}

}

}

static void InitializeDepartments()

{

departmentShelves["NUERO"] = new Dictionary<string, string>();

departmentShelves["HEART"] = new Dictionary<string, string>();

departmentShelves["BONES"] = new Dictionary<string, string>();

}

static void AddPatientRecord()

{

Console.Write("\nEnter Serial Number (four digits): ");

string serial = Console.ReadLine();

Console.Write("\nEnter Department Initials (NUERO, HEART, BONES): ");

string department = Console.ReadLine();

Console.Write("\nEnter Patient's Name: ");

string name = Console.ReadLine();

string record = $"{serial}-{department}-{name}";

if (departmentShelves.ContainsKey(department))

{

departmentShelves[department][serial] = record;

Console.WriteLine("RECORD ADDED SUCCESSFULLY");

ShowRecords(department);

}

else

{

Console.WriteLine("Invalid department. Record not added.");

}

}

static void RemovePatientRecord()

{

Console.Write("Enter Serial Number of the record to remove: ");

string serial = Console.ReadLine();

Console.Write("Enter Department Initials: ");

string department = Console.ReadLine();

if (departmentShelves.ContainsKey(department) && departmentShelves[department].ContainsKey(serial))

{

departmentShelves[department].Remove(serial);

Console.WriteLine("Record removed successfully.");

ShowRecords(department);

}

else

{

Console.WriteLine("Record not found. Removal failed.");

}

}

static void SortRecords()

{

Console.Write("Enter Department Initials to sort (NUERO, HEART, BONES): ");

string department = Console.ReadLine();

if (departmentShelves.ContainsKey(department))

{

var sortedRecords = departmentShelves[department]

.OrderByDescending(record => record.Key)

.ToDictionary(record => record.Key, record => record.Value);

departmentShelves[department] = sortedRecords;

Console.WriteLine("Records sorted successfully.");

ShowRecords(department);

}

else

{

Console.WriteLine("Invalid department. Sorting failed.");

}

}

static void SearchForPatient()

{

Console.Write("Enter Serial Number to search (optional): ");

string serial = Console.ReadLine();

Console.Write("Enter Department Initials (optional): ");

string department = Console.ReadLine();

Console.Write("Enter Patient's Name (optional): ");

string name = Console.ReadLine();

if (string.IsNullOrEmpty(serial) && string.IsNullOrEmpty(department) && string.IsNullOrEmpty(name))

{

Console.WriteLine("Please provide search criteria.");

return;

}

var results = departmentShelves.SelectMany(departmentShelf => departmentShelf.Value)

.Where(record =>

(string.IsNullOrEmpty(serial) || record.Key.Contains(serial)) &&

(string.IsNullOrEmpty(department) || record.Value.Contains(department)) &&

(string.IsNullOrEmpty(name) || record.Value.Contains(name))

)

.ToList();

if (results.Any())

{

Console.WriteLine("Matching Records:");

foreach (var result in results)

{

Console.WriteLine(result.Value);

}

}

else

{

Console.WriteLine("No matching records found.");

}

}

static void ShowRecords(string department)

{

Console.WriteLine($"{department} Department Records:");

foreach (var record in departmentShelves[department])

{

Console.WriteLine(record.Value);

}

Console.WriteLine($"{department} Records Count: {departmentShelves[department].Count}");

}

}