C++ Hospital Management

```
#include<iostream.h>
#include<conio.h>
#include<process.h>
class all
{
private:
struct address
{
int house;
char street[30];
char city[30];
char state[30];
char country[30];
};
struct age
{
int day;
int month;
int year;
};
struct patient_info
{
char name[50];
address AD1; //nested structure implemented
age A1; //nested structure inplemented
int martial_status;
int reg_no;
int bld_group;
int sex;
}
PI[100];
int task;
protected:
void enter_patient_info();
void show_patient_detail();
public:
void software_detail();
void tasks();
char answer;
char answer1;
char ch;
int serial;
};
class date
{
private:
int date;
int month;
int year;
```

```
public:
void enter_date();
void show_date();
};
class dob
{
private:
struct dob1
{
int date;
int month;
int year;
int rem;
}
DOB11[100];
public:
void enter_date();
void show_date();
};
int i=0;
int rem;
int count;
int regis;
int attempt;
int temp;
int show_count=0;
all A1; //object declared
date D1; //object declared
dob DOB1; //object declared
void main()
{
count=0;
cout<<"Welcome to..."<<"";
cout<<"***HOSPITAL MANAGEMENT SOFTWARE***"<<"";
D1.enter_date();
A1.tasks();
}
void all::tasks()
{
attempt=0;
D1.show_date();
cout<<"***HOSPITAL MANAGEMENT SOFTWARE***"<<"";
cout<<"**Hospital Management Tasks**"<<"";
cout<<" ****************************"<<"";
cout<<"Please select a task to do...."<<"";
cout<<"1. Enter a new patient information ."<<"";
cout<<"2. View detail of existing patient ."<<"";
cout<<"3. View detail about the program ."<<"";
cout<<"4. Exit from the program ."<<"";
//other function remain
cout<<"Enter your task serail :"<<"";
cin>>task;
switch(task)
{
```

```
case 1:
{
A1.enter_patient_info();
break;
}
case 2:
{
A1.show_patient_detail();
break;
}
case 3:
{
A1.software_detail();
break;
}
case 4:{
clrscr();
cout<<"Thank You for trying this program !!!"<<"";
cout<<" This is the end of program...."<<"";
cout<<"Press any key to exit....."<<"";
getch();
exit(0);
break;
}
default:
{
clrscr();
cout<<"Invalid task serial ."<<"";
cout<<"Press any key to continue...."<<"";
getch();
clrscr();
A1.tasks();
}
}
}
void all::enter_patient_info()
{
clrscr();
answer='y';
if(count==0)
{
serial=1;
}
else
{
i=serial;
}
for(i=serial;answer=='y'||answer=='Y';i++)
{
PI[i].reg_no=i;
temp=serial;
cout<<"***ENTERING INFORMATION FOR PATIENT SERIAL NUMBER "
<<i<<"***"<"";
cin.get(ch);
```

```
cout<<"Registration Number : "<<PI[i].reg_no<<"";
cout<<"Enter the name of patient :"<<"";
clreol();
cin.getline(PI[i].name,50);
cout<<"Sex (1-Male 2-Female) :"<<"";
clreol();
cin>>PI[i].sex;
while(PI[i].sex!=1&&PI[i].sex!=2)
{
cout<<"Invalid input for sex of patient!!!"<<"";
cout<<"Sex :"<<"";
clreol();
cin>>PI[i].sex;
}
cout<<"***ENTERING ADDRESS**"<<"";
cout<<"House number :"<<"
";
clreol();
cin>>PI[i].AD1.house;
while(PI[i].AD1.house<=0)
{
cout<<"Invalid input for house number ."<<"";
cout<<"Again enter the house number ."<<"";
clreol();
cin>>PI[i].AD1.house;
}
cin.get(ch);
cout<<"Street :"<<"";
clreol();
cin.getline(PI[i].AD1.street,30);
cout<<"City :"<<"";
clreol();
cin.getline(PI[i].AD1.city,30);
cout<<"State :"<<"";
clreol();
cin.getline(PI[i].AD1.state,30);
cout<<"Country :"<<"";
clreol();
cin.getline(PI[i].AD1.country,30);
DOB1.enter_date();
//to calculate age
cin.get(ch);
cout<<"Martial status(1-Married,2-Not Married ):"<<"";
if(count!=0)
{
clreol();
}
cin>>PI[i].martial_status;
while(PI[i].martial_status<1||PI[i].martial_status>2)
{
cout<<"Invalid input for martial status ."<<"";
cout<<"Enter a valid martial status :"<<"";
clreol();
cin>>PI[i].martial_status;
```

```
}
cin.get(ch);
if(count!=0)
{
clreol();
}
clreol();
cout<<"Blood group :"<<"";
clreol();
cout<<"1. A+ "<<"";
clreol();
cout<<"2. A- "<<"";
clreol();
cout<<"3. B+ "<<"";
clreol();
cout<<"4. B- "<<"";
clreol();
cout<<"5. AB+ "<<"";
clreol();
cout<<"6. AB- "<<"";
clreol();
cout<<"7. O+ "<<"";
clreol();
cout<<"8. O- "<<"";
clreol();
cout<<"Enter :"<<"";
clreol();
cin>>PI[i].bld_group;
switch(PI[i].bld_group)
{
case 1:
case 2:
case 3:
case 4:
case 5:
case 6:
case 7:
case 8:
{
break;
}
default:
{
while(PI[i].bld_group!=1&&PI[i].bld_group!=2&&PI[i].bld_group!=3&&
PI[i].bld_group!=4&&PI[i].bld_group!=5&&PI[i].bld_group!=6&&
PI[i].bld_group!=7&&PI[i].bld_group!=8)
{
clreol();
cout<<"Invalid input !"<<"";
cout<<"Blood Group :"<<"";
clreol();
cin>>PI[i].bld_group;
}
break;
```

```
}
}
cin.get(ch);
cout<<"Want to enter information for another patient ? "<<"";
clreol();
cin>>answer;
count++;
serial++;
}
clrscr();
A1.tasks();
}
void dob::enter_date()
{
clreol();
cout<<"Date of birth"<<"";
clreol();
cout<<"Year :";
clreol();
clreol();
cin>>DOB11[temp].year;
if(DOB11[temp].year<=0||DOB11[temp].year>10000)
{
do
{
clreol();
cout<<"Invalid input for year !"<<"";
cout<<"Please enter the year correctly :"<<"";
cin>>DOB11[temp].year;
}
while(DOB11[temp].year<0||DOB11[temp].year>10000);
}
clreol();
cout<<"Month :";
clreol();
cin>>DOB11[temp].month;
if(DOB11[temp].month<=0||DOB11[temp].month>12)
{
do
{
clreol();
cout<<"Invalid input for month !"<<"";
cout<<"Again enter the month :"<<"";
clreol();
if(count!=0)
{
clreol();
}
cin>>DOB11[temp].month;
}
while(DOB11[temp].month<0||DOB11[temp].month>12);
}
cout<<"Date :";
clreol();
```

```
switch(DOB11[temp].month)
{
case 1:
case 3:
case 5:
case 7:
case 8:
case 10:
case 12:
{
cin>>DOB11[temp].date;
while(DOB11[temp].date<1||DOB11[temp].date>31)
{
clreol();
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
clreol();
cin>>DOB11[temp].date;
}
break;
}
case 2:
{
cin>>DOB11[temp].date;
if(DOB11[temp].year%4==0)
{
while(DOB11[temp].date<0||DOB11[temp].date>29)
//for leap year
{
clreol();
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
clreol();
cin>>DOB11[temp].date;
}
}
else
{
while(DOB11[temp].date<0||DOB11[temp].date>28)
//for non-leap year
{
clreol();
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
clreol();
cin>>DOB11[temp].date;
}
}
break;
}
default:
{
cin>>DOB11[temp].date;
while(DOB11[temp].date<1||DOB11[temp].date>30)
```

```
{
clreol();
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
clreol();
cin>>DOB11[temp].date;
}
break;
}
} //end of switch
clreol();
}
void date::enter_date()
{
cout<<"First of all I need the current date ..."<<"";
cout<<"Year :";
cin>>year;
if(year<=0||year>10000)
{
do
{
cout<<"Invalid input for year !"<<"";
cout<<"Please enter the year correctly :"<<"";
cin>>year;
}
while(year<0||year>10000);
}
cout<<"Month :";
cin>>month;
if(month<=0||month>12)
{
do
{
cout<<"Invalid input for month !"<<"";
cout<<"Again enter the month :"<<"";
cin>>month;
}
while(month<0||month>12);
}
cout<<"Date :";
switch(month)
{
case 1:
case 3:
case 5:
case 7:
case 8:
case 10:
case 12:
{
cin>>date;
while(date<1||date>31)
{
cout<<"Invalid date !"<<"";
```

```
cout<<"Again enter the date :"<<"";
cin>>date;
}
break;
}
case 2:
{
cin>>date;
if(year%4==0)
{
while(date<0||date>29) //for leap year
{
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
cin>>date;
}
}
else
{
while(date<0||date>28) //for non-leap year
{
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
cin>>date;
}
}
break;
}
default:
{
cin>>date;
while(date<1||date>30)
{
cout<<"Invalid date !"<<"";
cout<<"Again enter the date :"<<"";
cin>>date;
}
break;
}
} //end of switch
}
void date::show_date() //remove the goto ststements in this function
{
clrscr();
cout<<"Hello....It's ";
cout<<date;
rem=date%10;
switch(date)
{
case 11:
case 12:
case 13:
case 14:
case 15:
```

```
case 16:
case 17:
case 18:
case 19:
case 20:
{
cout<<"th ";
goto over;
}
}
switch(rem)
{
case 1:
{
cout<<"st ";
break;
}
case 2:
{
cout<<"nd ";
break;
}
case 3:
{
cout<<"rd ";
break;
}
default:
{
cout<<"th ";
break;
}
}
over:
switch(month)
{
case 1:
{
cout<<"January , ";
break;
}
case 2:
{
cout<<"February , ";
break;
}
case 3:
{
cout<<"March , ";
break;
}
case 4:
{
cout<<"April , ";
```

```
break;
}
case 5:
{
cout<<"May , ";
break;
}
case 6:
{
cout<<"June , ";
break;
}
case 7:
{
cout<<"July , ";
break;
}
case 8:
{
cout<<"August , ";
break;
}
case 9:
{
cout<<"September , ";
break;
}
case 10:
{
cout<<"October , ";
break;
}
case 11:
{
cout<<"November , ";
break;
}
case 12:
{
cout<<"December , ";
break;
}
}
cout<<year<<" ";
}
void all::show_patient_detail()
{
do
{
clrscr();
cout<<"Enter registration number :"<<"";
clreol();
cin>>regis;
cin.get(ch);
```

```
show_count++;
if(regis>0&®is<serial)
{
clreol();
cout<<"***INFORMATION FOR PATIENT REGISTRATION NUMBER"
<<regis<<"***";
clreol();
cout<<"Name : "<<PI[regis].name<<"";
clreol();
cout<<"Sex : ";
clreol();
if(PI[regis].sex==1)
{
cout<<"Male "<<"";
clreol();
}
if(PI[regis].sex==2)
{
cout<<"Female "<<"";
clreol();
}
cout<<"Blood Group : ";
clreol();
switch(PI[regis].bld_group)
{
case 1:
{
clreol();
cout<<"A+";
break;
}
case 2:
{
clreol();
cout<<"A-";
break;
}
case 3:
{
clreol();
cout<<"B+";
break;
}
case 4:
{
clreol();
cout<<"B-";
break;
}
case 5:
{
clreol();
cout<<"AB+";
break;
```

```
}
case 6:
{
clreol();
cout<<"AB-";
break;
}
case 7:
{
clreol();
cout<<"O+";
break;
}
case 8:
{
clreol();
cout<<"O-";
break;
}
}
clreol();
cout<<"Date of birth : ";
clreol();
DOB1.show_date();
cout<<"Martial Status : ";
clreol();
if(PI[i].martial_status==1)
{
cout<<"Married "<<"";
clreol();
}
else
{
cout<<"Not married "<<"";
clreol();
}
clreol();
cout<<"**ADDRESS**"<<"";
clreol();
cout<<"House no. : "<<PI[regis].AD1.house;
clreol();
cout<<"Street : "<<PI[regis].AD1.street;
clreol();
cout<<"City : "<<PI[regis].AD1.city;
clreol();
cout<<"State : "<<PI[regis].AD1.state;
clreol();
cout<<"Country : "<<PI[regis].AD1.country;
clreol();
}
else
{
if(regis==1)
{
```

```
cout<<"Database is empty !!!"<<"";
cout<<"Press any key to exit to main task menu..."<<"";
getch();
clrscr();
A1.tasks();
}
attempt++;
if(attempt==3)
{
cout<<"You have entered wrong registration number 3 times ."<<"";
cout<<"Access Denied!!! "<<"";
cout<<"Please try again later. "<<"";
cout<<"Press any key to exit to main task menu..."<<"";
getch();
clrscr();
A1.tasks();
}
clreol();
cout<<"Sorry, the registration number is invalid ."<<"";
cout<<"Press any key to continue...."<<"";
getch();
clreol();
A1.show_patient_detail();
}
clreol();
cout<<"Want to see information of another patient :"<<"";
clreol();
cin>>answer1;
}
while(answer1=='y'||answer1=='Y');
clreol();
clrscr();
A1.tasks();
}
void dob::show_date()
{
cout<<DOB11[regis].date;
rem=DOB11[regis].date%10;
switch(DOB11[regis].date)
{
case 11:
case 12:
case 13:
case 14:
case 15:
case 16:
case 17:
case 18:
case 19:
case 20:
{
cout<<"th ";
goto over;
}
```

```
}
switch(rem)
{
case 1:
{
cout<<"st ";
break;
}
case 2:
{
cout<<"nd ";
break;
}
case 3:
{
cout<<"rd ";
break;
}
default:
{
cout<<"th ";
break;
}
}
over:
switch(DOB11[regis].month)
{
case 1:
{
cout<<"January , ";
break;
}
case 2:
{
cout<<"February , ";
break;
}
case 3:
{
cout<<"March , ";
break;
}
case 4:
{
cout<<"April , ";
break;
}
case 5:
{
cout<<"May , ";
break;
}
case 6:
{
```

```
cout<<"June , ";
break;
}
case 7:
{
cout<<"July , ";
break;
}
case 8:
{
cout<<"August , ";
break;
}
case 9:
{
cout<<"September , ";
break;
}
case 10:
{
cout<<"October , ";
break;
}
case 11:
{
cout<<"November , ";
break;

}
case 12:
{
cout<<"December , ";
break;
}
}
cout<<DOB11[regis].year<<" ";
}
void all::software_detail()
{
clrscr();
cout<<"***SOFTWARE DETAILS***";
cout<<" Programming Language : C++ "<<"";
cout<<" Aim : Simulation of the software used in Hospital"<<"";
cout<<"Hope you like it..."<<"";
cout<<"Send your comments to : zubairsaif700@gmail.com ."<<"";
cout<<"Thank You for trying this program. "<<"";
cout<<"Press any key to return to the main task menu......."<<" ";
getch();
A1.tasks();
}
```