CODE

25 June, 2021

```cpp
//                                  Hospital Management System

#include<iostream>
#include<fstream>
#include<iomanip>
#include<string>
#include<stdlib.h>
#include<stdio.h>

using namespace std;

//function declarations
// -------------------------------------------------------------
void clear_screen();
void menu();
int chooseMain();
void getDoctorDatabase();
void getPatientDatabase();
void getBedDatabase();
void getPatientReport();


//the password class
// -------------------------------------------------------------

class password {
private:
  string pass;
public:

  void getPassword() {
    cout << "\n\t\t\t\t\t\tChoose a password for your system: " ;
    cin >> pass;
  }

  int verifyPassword()
  {
    string x;
    cout << "\n\t\t\t\t\t\tEnter the password: ";
    cin >> x;
    if (x == pass)
      return 1;
    else
      return 0;
  }

  void passwordProtect() {
    int veri = verifyPassword();

    if (veri == 1) {
      cout << "\n\t\t\t\t\t\tAccess Granted!" << endl;
    }
    else
    {
      cout << "\n\t\t\t\t\tAccess Denied!\n\n\t\t\t\t\tSystem will shut down
            \n\n\t\t\t\t\tPlease come back again." << endl;
      exit(0);
    }
  }
};

// password class ends
// -------------------------------------------------------------




//the doctor class
```

```cpp
70  // ---------------------------------------------------------------
71
72  class doctor {
73  private:
74    string d_fname, d_lname;
75    int d_ID, d_age, experience, d_totalno;
76    string qual, d_city;
77    string getSpec();
78  public:
79    friend void addDoc();
80    friend void editDoc(doctor &docobj);
81    friend void displayDoc();
82    friend void deleteDoc();
83    friend void dispDatabase();
84    friend void numberofDocs();
85    friend void getDoctorDatabase();
86  };
87  // doctor class ends here
88  // ---------------------------------------------------------------
89
90
91
92
93  // Doctor Speciality
94  //---------------------------------------------------------------
95  string doctor::getSpec() {
96    string sp;
97    cout << "\t\t\t\t\t\t\t1.GENERAL_MEDICINE" << endl;
98    cout << "\t\t\t\t\t\t\t2.NEUROLOGY" << endl;
99    cout << "\t\t\t\t\t\t\t3.ONCOLOGY" << endl;
100   cout << "\t\t\t\t\t\t\t4.HAEMATOLOGY" << endl;
101   cout << "\t\t\t\t\t\t\t5.GASTROENTEROLOGY" << endl;
102   cout << "\t\t\t\t\t\t\t6.GYNAECOLOGY " << endl;
103   cout << endl;
104   int s;
105   cout << "\t\t\t\t\t\t";
106   cin >> s;
107   switch (s) {
108
109   case 1: {
110     sp = "GENERAL_MEDICINE";
111     break;
112   }
113   case 2: {
114     sp = "NEUROLOGY";
115     break;
116   }
117   case 3: {
118     sp = "ONCOLOGY";
119     break;
120   }
121   case 4: {
122     sp = "HAEMATOLOGY";
123     break;
124   }
125   case 5: {
126     sp = "GASTROENTEROLOGY";
127     break;
128   }
129   case 6: {
130     sp = "GYNAECOLOGY ";
131     break;
132   }
133   default: {
134     cout << "\n\t\t\t\t\tCome on!Enter a valid option." << endl;
135
136   }
137
138   }
139   return sp;
140
141 }
142 // Doctor Speciality Ends
```

```cpp
143  //-----------------------------------------------------------------
144
145
146
147
148
149  //the Adding New Doctor function
150  //-----------------------------------------------------------------
151
152  void addDoc() {
153      doctor d;
154      cout << "\n\t\t\t\t\tFill up the following details" << endl;
155      cout << "\t\t-----------------------------------------------------------------" << endl;
156      cout << "\n\t\t\t\t\t\tID: ";
157      cin >> d.d_ID;
158      cout << "\n\t\t\t\t\t\tFirst Name: ";
159      cin >> d.d_fname;
160      cout << "\n\t\t\t\t\t\tLast Name: " ;
161      cin >> d.d_lname;
162      cout << "\n\t\t\t\t\t\tAge: " ;
163      cin >> d.d_age;
164      cout << "\n\t\t\t\t\t\tChoose Qualification among the following:
165          \n\t\t\t\t\t\t\t1.MBBS\n\t\t\t\t\t\t\t2.MD" << endl;
166      int q;
167      cout << "\t\t\t\t\t\t";
168      cin >> q;
169      if (q == 1)
170          d.qual = "MBBS";
171      else d.qual = "MD";
172      cout << "\n\t\t\t\t\t\tChoose Specialization among the following: " << endl;;
173      string spec = d.getSpec();
174      cout << "\n\t\t\t\t\t\tExperience (in Years): ";
175      cin >> d.experience;
176      cout << "\n\t\t\t\t\t\tCity: ";
177      cin >> d.d_city;
178
179
180      clear_screen();
181      fstream fileObj;
182      fileObj.open("doctor.txt", ios::app );
183      fileObj.seekp (0, ios::end);
184      fileObj << d.d_ID << endl << d.d_fname << endl << d.d_lname <<
185              endl << d.d_age << endl << d.qual << endl << spec << endl <<
186              d.experience << endl << d.d_city << endl;
187      fileObj.close();
188
189
190
191      cout << "\n\t\t\t\t\tNew Doctor profile created successfully!" << endl << endl << endl
          ;
192
193
194      cout << "\n\t\t\t\t\tEnter 1 to return to Doctor Database\n\t\t\t\t\tEnter 2 to Exit."
              << endl;
195      cout << "\t\t\t\t\t";
196      int c;
197      cin >> c;
198      if (c == 1) {
199          clear_screen();
200          getDoctorDatabase();
201      }
202
203      else
204          exit(0);
205
206  }
207
208  // Add Doctor Functions End Here
209  //-----------------------------------------------------------------
210
211
212
213
```

3

```cpp
214
215
216  // Edit Doctor Function Starts Here
217  //-------------------------------------------------------------
218
219  void editDoc(doctor &docobj)
220  {
221
222      string x;
223      cout << "\n\n\t\t\t\t\tEnter the ID of the Doctor whose profile you want to Edit: " ;
224      cin >> x;
225      fstream fileObj("doctor.txt");
226      fileObj.seekg(0);
227
228      int age, exp, count = 0;
229      string ID, fname, lname, ci, qua, spe;
230      while (fileObj >> ID >> fname >> lname >> age >> qua >> spe >> exp >> ci)
231      {
232          if (ID == x) {
233              cout << "\n\n\t\t\t\t\t1.ID:" << ID << endl
234                   << "\t\t\t\t\t2.First Name:" << fname << endl
235                   << "\t\t\t\t\t3.Last Name:" << lname << endl
236                   << "\t\t\t\t\t4.Age:" << age << endl
237                   << "\t\t\t\t\t5.Qualification:" << qua << endl
238                   << "\t\t\t\t\t6.Specialization:" << spe << endl
239                   << "\t\t\t\t\t7.Experience:" << exp << endl
240                   << "\t\t\t\t\t8.City:" << ci << endl << endl;
241              count++;
242              break;
243          }
244      }
245
246      fileObj.close();
247
248
249
250      if (count == 0) {
251          cout << "\n\t\t\t\t\tNo matching records found!" << endl;
252          cout << "\n\n\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\tEnter 2
                 to Exit." << endl;
253          cout << "\t\t\t\t\t";
254          int c;
255          cin >> c;
256          if (c == 1) {
257              clear_screen();
258              getDoctorDatabase();
259          }
260
261          else
262              exit(0);
263      }
264
265
266
267      fstream file("doctor.txt");
268      file.seekg (0, ios::beg);
269      file.seekg(0);
270      string line;
271      ofstream tempObj;
272      tempObj.open("copy.txt");
273      tempObj.seekp(0);
274      int skip = 0;
275      int age1, exp1;
276      string fname1, lname1, ci1, qua1, spe1;
277      file.seekg(0); file.seekg(0, ios::cur);
278
279
280      while (file >> line >> fname1 >> lname1 >> age1 >> qua1 >> spe1 >> exp1 >> ci1)
281      {
282          if (line != x) {
283              tempObj << line << endl << fname1 << endl
284                      << lname1 << endl << age1 << endl << qua1 << endl
285                      << spe1 << endl << exp1 << endl << ci1 << endl;
```

```cpp
      }
  }

  file.close();
  tempObj.close();
  remove("doctor.txt");
  int result;
  char oldname[] = "copy.txt";
  char newname[] = "doctor.txt";
  result = rename( oldname , newname );
  int edit;
  cout << "\t\t\t\t\tEnter number corresponding to the Field you want to Edit: ";
  cin >> edit;
  cout << endl;
  switch (edit)
  {
  case 1: {
    cout << "\t\t\t\t\tID :" ;
    cin >> ID;
    break;
  }
  case 2: {
    cout << "\t\t\t\t\tFirst Name: " ;
    cin >> fname;

    break;
  }
  case 3: {
    cout << "\t\t\t\t\tLast Name: " ;
    cin >> lname;

    break;
  }
  case 4: {
    cout << "\t\t\t\t\tAge: " ;
    cin >> age;
    break;
  }
  case 5: {
    cout << "\t\t\t\t\tChoose Qualification among the following:\n\t\t\t\t\t\t
         1.MBBS\n\t\t\t\t\t\t2.MD" << endl;
    cout << "\t\t\t\t\t";
    int q;
    cin >> q;
    if (q == 1) {
      qua = "MBBS";
    }

    else {
      qua = "MD";
    }

    break;
  }
  case 6: {
    clear_screen();
    spe = docobj.getSpec();
    break;
  }
  case 7: {
    cout << "\t\t\t\t\tExperience: " ;
    cin >> exp;
    break;
  }
  case 8: {
    cout << "\t\t\t\t\tCity: " ;
    cin >> ci;
    break;
  }
  default: {
    cout << "\t\t\t\t\t\tInvalid Input!" << endl;
    cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\t\tEnter 2
       to Exit." << endl;
```

```cpp
            cout << "\t\t\t\t\t\t";
            int c;
            cin >> c;
            if (c == 1) {
                clear_screen();
                getDoctorDatabase();
            }

            else
                exit(0);
    }
    }

    clear_screen();

    cout << "\t\t\t\t-------------Updated Profile----------------" << endl << endl;
    cout << "\n\n\t\t\t\t\t1.ID:" << ID << endl << "\t\t\t\t\t2.First Name:" << fname <<
        endl << "\t\t\t\t\t3.Last Name:" << lname << endl << "\t\t\t\t\t4.Age:" << age <<
        endl << "\t\t\t\t\t5.Qualification:" << qua << endl << "\t\t\t\t\t6.Specialization:"
         << spe << endl << "\t\t\t\t\t7.Experience:" << exp << endl << "\t\t\t\t\t8.City:"
        << ci << endl << endl;

    fstream newObj;
    newObj.open("doctor.txt", ios::app);
    newObj.seekp (0, ios::end);
    newObj << ID << endl << fname << endl << lname
           << endl << age << endl << qua << endl << spe
           << endl << exp << endl << ci << endl;
    newObj.close();



    cout << "\t\t\t\t" << "----------------------------------------------------" << endl;

    cout << "\n\n\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\tEnter 2 to
        Exit." << endl;
    cout << "\n\t\t\t\t\t";
    int c;
    cin >> c;
    if (c == 1) {
        clear_screen();
        getDoctorDatabase();
    }

    else
        exit(0);
}


// Edit Doctor Functions Ends Here
//-------------------------------------------------------------



//the Display Doctor Function
//-------------------------------------------------------------
void displayDoc()
{
    int x;
    cout << "\n\n\t\t\t\t\tEnter the ID of the Doctor whose profile you want to Edit: " ;
    cin >> x;
    ifstream file; file.open("doctor.txt", ios::out | ios::in);
    file.seekg(0);

    int ID, age, exp, count = 0;
    string fname, lname, ci, qua, spe;

    while (file >> ID >> fname >> lname >> age >> qua >> spe >> exp >> ci)
    {

        if (ID == x)
        {
```

```cpp
          cout << "\n\n\t\t\t\t\t1.ID:" << ID << endl
                << "\t\t\t\t\t2.First Name:" << fname << endl
                << "\t\t\t\t\t3.Last Name:" << lname << endl
                << "\t\t\t\t\t4.Age:" << age << endl
                << "\t\t\t\t\t5.Qualification:" << qua << endl
                << "\t\t\t\t\t6.Specialization:" << spe << endl
                << "\t\t\t\t\t7.Experience:" << exp << endl
                << "\t\t\t\t\t8.City:" << ci << endl << endl;
          count++;
      }
  }
  file.close();
  if (count == 0)
     cout << "\n\t\t\t\t\t\tNo matching records found!" << endl;

  cout << "\t\t\t\t\t" << "----------------------------------------------" << endl;

  cout << "\n\n\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\tEnter 2 to
     Exit." << endl;
  cout << "\n\t\t\t\t\t";
  int c;
  cin >> c;
  if (c == 1) {
     clear_screen();
     getDoctorDatabase();
  }

  else
     exit(0);

}

// Display Doctor Functions End
// --------------------------------------------------------------



//the Delete Doctor function
// --------------------------------------------------------------
void deleteDoc() {
  string x;
  cout << "\n\n\t\t\t\t\tEnter the ID of the Doctor whose profile you want to Edit: " ;

  cin >> x;
  fstream fileObj("doctor.txt");
  fileObj.seekg(0);

  int age, exp, count = 0;
  string ID, fname, lname, ci, qua, spe;
  while (fileObj >> ID >> fname >> lname >> age >> qua >> spe >> exp >> ci)
  {
     if (ID == x) {
        cout << "\n\n\t\t\t\t\t1.ID:" << ID << endl
                << "\t\t\t\t\t2.First Name:" << fname << endl
                << "\t\t\t\t\t3.Last Name:" << lname << end
                << "\t\t\t\t\t4.Age:" << age << endl
                << "\t\t\t\t\t5.Qualification:" << qua << endl
                << "\t\t\t\t\t6.Specialization:" << spe << endl
                << "\t\t\t\t\t7.Experience:" << exp << endl
                << "\t\t\t\t\t8.City:" << ci << endl << endl;
        count++;
        break;
     }
  }

  fileObj.close();
  if (count == 0) {
     cout << "\n\t\t\t\t\t\tNo matching records found!" << endl;
     cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\t\tEnter 2
      to Exit." << endl;
     cout << "\t\t\t\t\t\t";
     int c;
     cin >> c;
```

```cpp
    if (c == 1) {
       clear_screen();
       getDoctorDatabase();
    }

    else
       exit(0);
  }


  cout << "\t\t\t\t\tAre you sure you want to delete this profile?
       \n\t\t\t\t\t 1.YES
       \n\t\t\t\t\t2.NO"
       << endl;
  cout << "\n\t\t\t\t\t";
  int n; cin >> n;
  if (n == 1) {
    fstream file("doctor.txt");
    file.seekg (0, ios::beg);
    file.seekg(0);
    string line;
    ofstream tempObj;
    tempObj.open("copy.txt");
    tempObj.seekp(0);
    int skip = 0;
    int age1, exp1;
    string fname1, lname1, ci1, qua1, spe1;
    file.seekg(0); file.seekg(0, ios::cur);


    while (file >> line >> fname1 >> lname1 >> age1 >> qua1 >> spe1 >> exp1 >> ci1)
    {
      if (line != x) {
        tempObj << line << endl << fname1 << endl
                << lname1 << endl << age1 << endl << qua1
                << endl << spe1 << endl << exp1 << endl
                << ci1 << endl;
      }
    }

    file.close();
    tempObj.close();
    remove("doctor.txt");
    int result;
    char oldname[] = "copy.txt";
    char newname[] = "doctor.txt";
    result = rename( oldname , newname );
    clear_screen();
    cout << "\n\t\t\t\t\tFile successfully deleted" << endl;

  }
  clear_screen();

  cout << "\t\t\t\t\t" << "----------------------------------------------" << endl;

  cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\t\tEnter 2
    to Exit." << endl;
  cout << "\n\t\t\t\t\t\t";
  int c;
  cin >> c;
  if (c == 1) {
    clear_screen();
    getDoctorDatabase();
  }

  else
     exit(0);

}

// Delete Doctor Functions Ends Here
// ------------------------------------------------------------

```

```cpp
//the Display DataBase Function
//---------------------------------------------------------------
void dispDatabase() {
  fstream fileObj("doctor.txt");
  fileObj.seekg(0);
  cout << "\t\t\t\t\tDOCTOR DATABASE" << endl << endl;
  int age, exp, count = 0;
  string ID, fname, lname, ci, qua, spe;
  while (fileObj >> ID >> fname >> lname >> age >> qua >> spe >> exp >> ci)
  { count++;
    cout << "\t\t\t\t\tDoctor Profile:" << count << endl;
    cout << "\n\t\t\t\t\t1.ID:" << ID << endl
         << "\t\t\t\t\t2.First Name:" << fname << endl
         << "\t\t\t\t\t3.Last Name:" << lname << endl
         << "\t\t\t\t\t4.Age:" << age << endl
         << "\t\t\t\t\t5.Qualification:" << qua << endl
         << "\t\t\t\t\t6.Specialization:" << spe << endl
         << "\t\t\t\t\t7.Experience:" << exp << endl
         << "\t\t\t\t\t8.City:" << ci << endl << endl;

  }

  if (count == 0) {
    cout << "\n\t\t\t\t\t\tNo matching records found!" << endl;
  }

  cout << "\t\t\t\t\t" << "--------------------------------------------------" << endl;


  cout << "\n\n\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\tEnter 2 to
    Exit." << endl;
  cout << "\n\t\t\t\t\t";
  int c;
  cin >> c;
  if (c == 1) {
    clear_screen();
    getDoctorDatabase();
  }

  else
    exit(0);
}

// Display Database Function Ends Here
//---------------------------------------------------------------




//Total NUmber Of Doctors function
// --------------------------------------------------------------
void numberofDocs() {

  fstream fileObj("doctor.txt");
  fileObj.seekg(0);
  int age, exp, count = 0;
  string ID, fname, lname, ci, qua, spe;
  while (fileObj >> ID >> fname >> lname >> age >> qua >> spe >> exp >> ci)
  { count++;
  }
  cout << "\n\n\t\t\t\t\tThe Total number of doctors in the hospital=" << count << endl
    << endl;

  cout << "\t\t\t\t\t" << "--------------------------------------------------" << endl;

  cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\t\tEnter 2
    to Exit." << endl;
  cout << "\n\t\t\t\t\t\t";
```

```cpp
    int c;
    cin >> c;
    if (c == 1) {
        clear_screen();
        getDoctorDatabase();
    }

    else
        exit(0);
}
//Total NUmber Of Doctors function Ends Here
// -------------------------------------------------------------




//Patient Class
// -------------------------------------------------------------
class patient {
private:
    string fname, lname, occupation, p_city, dis, addDate, relDate;
    int p_age, p_ID;
    long double phnumber, bill;
    string getDept();
public:
    friend void getDoctor();
    friend void addPat();
    friend void editPat(patient &pat);
    friend void displayPat();
    friend void deletePat();
    friend void dispPatDatabase();
    friend void numberofPats();
    friend void getPatientDatabase();

};

//-------------------------------------------------------------



// Department Function
//-------------------------------------------------------------
string patient::getDept() {
    string sp;
    cout << "\t\t\t\t\t\t\t1.GENERAL_MEDICINE" << endl;
    cout << "\t\t\t\t\t\t\t2.NEUROLOGY" << endl;
    cout << "\t\t\t\t\t\t\t3.ONCOLOGY" << endl;
    cout << "\t\t\t\t\t\t\t4.HAEMATOLOGY" << endl;
    cout << "\t\t\t\t\t\t\t5.GASTROENTEROLOGY" << endl;
    cout << "\t\t\t\t\t\t\t6.GYNAECOLOGY " << endl << endl;
    int s;
    cout << "\t\t\t\t\t\t";
    cin >> s;
    switch (s) {

    case 1: {
        sp = "GENERAL_MEDICINE";
        break;
    }
    case 2: {
        sp = "NEUROLOGY";
        break;
    }
    case 3: {
        sp = "ONCOLOGY";
        break;
    }
    case 4: {
        sp = "HAEMATOLOGY";
        break;
```

```cpp
712    }
713    case 5: {
714      sp = "GASTROENTEROLOGY";
715      break;
716    }
717    case 6: {
718      sp = "GYNAECOLOGY ";
719      break;
720    }
721    default: {
722      cout << "\nCome on!Enter a valid option." << endl;
723
724    }
725
726    }
727    return sp;
728 }
729 //--------------------------------------------------------------
730
731
732
733 // Get Doctor Function
734 //--------------------------------------------------------------
735 void getDoctor() {
736    patient p;
737    string dept;
738    cout << "\n\t\t\t\t\tEnter the kind of specialization you are looking for:" << endl <<
         endl;
739    dept = p.getDept();
740    fstream fileObj("doctor.txt");
741    fileObj.seekg(0);
742
743    int age, exp, count = 0;
744    string ID, fname, lname, ci, qua, spe;
745    while (fileObj >> ID >> fname >> lname >> age >> qua >> spe >> exp >> ci)
746    {
747      if (spe == dept) {
748        cout << "\n\t\t\t\tThe doctors available with  " << dept << " specialization are
       :" << endl << endl;
749        cout <<    "\n\t\t\t\t\t\t1.ID:" << ID << endl
750              << "\t\t\t\t\t\t2.First Name:" << fname << endl
751              << "\t\t\t\t\t\t3.Last Name:" << lname << endl
752              << "\t\t\t\t\t\t4.Age:" << age << endl
753              << "\t\t\t\t\t\t5.Qualification:" << qua << endl
754              << "\t\t\t\t\t\t6.Specialization:" << spe << endl
755              << "\t\t\t\t\t\t7.Experience:" << exp << endl
756              << "\t\t\t\t\t\t8.City:" << ci << endl << endl;
757        count++;
758      }
759    }
760
761    fileObj.close();
762    if (count == 0) {
763      cout << "\t\t\t\t\tNo doctors with this specialization found!" << endl;
764    }
765    cout << "\t\t\t\t\t" << "-----------------------------------------------" << endl;
766
767
768    cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Patient Database\n\t\t\t\t\tEnter 2 to
          Exit." << endl;
769    cout << "\n\t\t\t\t\t\t";
770    int c;
771    cin >> c;
772    if (c == 1) {
773      clear_screen();
774      getPatientDatabase();
775    }
776
777    else
778      exit(0);
779 }
780 // Get Doctor Functions End Here
781 //
```

```cpp
782
783
784
785   // Add patient Function
786   //----------------------------------------------------------------
787
788
789
790   void addPat() {
791
792     patient p;
793     cout << "\n\t\t\t\t\tFill up the following details" << endl;
794     cout << "\t\t----------------------------------------------------------------" << endl;
795
796     cout << "\n\t\t\t\t\t\tID: ";
797     cin >> p.p_ID;
798     cout << "\n\t\t\t\t\t\tFirst Name: ";
799     cin >> p.fname;
800     cout << "\n\t\t\t\t\t\tLast Name: " ;
801     cin >> p.lname;
802     cout << "\n\t\t\t\t\t\tAge: " ;
803     cin >> p.p_age;
804     cout << "\n\t\t\t\t\t\tPhone No: " ;
805     cin >> p.phnumber;
806     cout << "\n\t\t\t\t\t\tOccupation: " ;
807     cin >> p.occupation;
808     cout << "\n\t\t\t\t\t\tCity: ";
809     cin >> p.p_city;
810     cout << "\n\t\t\t\t\t\tDisease/Symptoms: ";
811     cin >> p.dis;
812     cout << "\n\t\t\t\t\t\tAdmission date  (DD/MM/YYYY): " ;
813     cin >> p.addDate;
814     cout << "\n\t\t\t\t\t\tRelease date  (DD/MM/YYYY): " ;
815     cin >> p.relDate;
816     cout << "\n\t\t\t\t\t\tTotal Bill generated: " ;
817     cin >> p.bill;
818     cout << "\n\t\t\t\t\t\tKind of Specialization required: " << endl;
819     string dept = p.getDept();
820     clear_screen();
821     fstream patObj;
822     patObj.open("patient.txt", ios::app );
823     patObj.seekp (0, ios::end);
824     patObj << p.p_ID << endl << p.fname << endl << p.lname << endl
825            << p.p_age << endl << p.phnumber << endl << p.occupation << endl
826            << p.p_city << endl << p.dis << endl << p.addDate << endl
827            << p.relDate << endl << p.bill << endl << dept << endl;
828     patObj.close();
829
830
831     cout << "\n\t\t\t\t\tNew Patient profile created successfully!" << endl << endl;
832
833     cout << "\n\t\t\t\t\t" << "----------------------------------------------" << endl;
834
835     cout << "\n\t\t\t\t\tEnter 1 to return to Patient Database\n\t\t\t\t\tEnter 2 to Exit. " << endl;
836     cout << "\t\t\t\t\t";
837     int c;
838     cin >> c;
839     if (c == 1) {
840       clear_screen();
841       getPatientDatabase();
842     }
843
844     else
845       exit(0);
846   }
847
848   // Add Patient Ends Here
849   //----------------------------------------------------------------
850
851
852
853
```

```cpp
// Edit Patient Detail
//----------------------------------------------------------------

void editPat(patient &pat)
{
    int x;
    cout << "\n\n\t\t\t\t\tEnter the ID of the Patient whose profile you want to Edit: " ;
    cin >> x;
    fstream patObj("patient.txt");
    patObj.seekg(0);
    string fname, lname, occupation, p_city, dis, addDate, relDate, dept;
    int p_age, p_ID, count = 0;
    long double phnumber, bill;
    while (patObj >> p_ID >> fname >> lname >> p_age >> phnumber
               >> occupation >> p_city >> dis >> addDate >> relDate >> bill >> dept)
    {
        if (p_ID == x) {
            cout << "\n\n\t\t\t\t\t1.ID:" << p_ID << endl
                 << "\t\t\t\t\t2.First Name:" << fname << endl
                 << "\t\t\t\t\t3.Last Name:" << lname << endl
                 << "\t\t\t\t\t4.Age" << p_age << endl
                 << "\t\t\t\t\t5.Phone No.:" << phnumber << endl
                 << "\t\t\t\t\t6.Occupation:" << occupation << endl
                 << "\t\t\t\t\t7.City:" << p_city << endl
                 << "\t\t\t\t\t8.Disease/Symptoms:" << dis << endl
                 << "\t\t\t\t\t9.Admission date(DD/MM/YYYY):" << addDate << endl
                 << "\t\t\t\t\t10.Release date(DD/MM/YYYY):" << relDate << endl
                 << "\t\t\t\t\t11.Total Bill generated:" << bill << endl
                 << "\t\t\t\t\t12.Kind of Specialization required:" << dept << endl << endl;
            break;
        }
    }

    patObj.close();
    if (count == 0)
    {
        cout << "\n\t\t\t\t\t\tNo matching records found!" << endl;
        cout << "\n\t\t\t\t\t" << "----------------------------------------------" << endl;

        cout << "\n\n\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\tEnter 2 to Exit." << endl;
        cout << "\t\t\t\t\t\t";
        int c;
        cin >> c;
        if (c == 1) {
            clear_screen();
            getPatientDatabase();
        }
        else
            exit(0);
    }


    fstream file("patient.txt");
    file.seekg (0, ios::beg);
    file.seekg(0);
    int line;
    ofstream tempObj;
    tempObj.open("copy2.txt");
    tempObj.seekp(0);
    int skip = 0;
    string fname1, lname1, occupation1, p_city1, dis1, addDate1, relDate1, dept1;
    int p_age1;
    long double phnumber1, bill1;
    file.seekg(0); file.seekg(0, ios::cur);

    while (file >> line >> fname1 >> lname1 >> p_age1 >> phnumber1 >> occupation1 >>
        p_city1 >> dis1 >> addDate1 >> relDate1 >> bill1 >> dept1)
```

```cpp
924    {
925      if (line != x) {
926        tempObj << line << endl << fname1 << endl << lname1
927                << endl << p_age1 << endl << phnumber1 << endl << occupation1
928                << endl << p_city1 << endl << dis1 << endl << addDate1 << endl
929                << relDate1 << endl << bill1 << endl << dept1 << endl;
930      }
931    }
932
933    file.close();
934    tempObj.close();
935    remove("patient.txt");
936    int result;
937    char oldname[] = "copy2.txt";
938    char newname[] = "patient.txt";
939    result = rename( oldname , newname );
940    int edit;
941    cout << "\t\t\t\tEnter number corresponding to the Field you want to Edit: ";
942    cin >> edit;
943    switch (edit)
944    {
945    case 1: {
946      cout << "\n\t\t\t\t\tID :" ;
947      cin >> p_ID;
948      break;
949    }
950    case 2: {
951      cout << "\n\t\t\t\t\tFirst Name:" ;
952      cin >> fname;
953
954      break;
955    }
956    case 3: {
957      cout << "\n\t\t\t\t\tLast Name:" ;
958      cin >> lname;
959
960      break;
961    }
962    case 4: {
963      cout << "\n\t\t\t\t\tAge:" ;
964      cin >> p_age;
965      break;
966    }
967    case 5: {
968      cout << "\n\t\t\t\t\tPhone No.:" ;
969      cin >> phnumber;
970      break;
971    }
972    case 6: {
973      cout << "\n\t\t\t\t\tOccupation:";
974      cin >> occupation;
975      break;
976    }
977    case 7: {
978      cout << "\n\t\t\t\t\tCity:" ;
979      cin >> p_city;
980      break;
981    }
982
983    case 8: {
984      cout << "\n\t\t\t\t\tDisease/Symptoms:" ;
985      cin >> dis;
986      break;
987    }
988    case 9: {
989      cout << "\n\t\t\t\t\tAdmission date  (DD/MM/YYYY):" ;
990      cin >> addDate;
991      break;
992    }
993    case 10:
994    {
995      cout << "\n\t\t\t\t\tRelease date  (DD/MM/YYYY):" ;
996      cin >> relDate;
```

```cpp
      }
      case 11: {
        cout << "\n\t\t\t\t\tTotal Bill generated:" ;
        cin >> bill;
        break;
      }
      case 12: {
        cout << "\n\t\t\t\t\tKind of Specialization required:" ;
        dept = pat.getDept();
        break;
      }
      default: {
        cout << "\n\t\t\t\t\t\tInvalid Input!" << endl;
        cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Doctor Database\n\n\t\t\t\t\t\tEnter 2
          to Exit." << endl;
        cout << "\t\t\t\t\t\t";
        int c;
        cin >> c;
        if (c == 1) {
          clear_screen();
          // getDoctorDatabase();
          getPatientDatabase();

        }

        else
          exit(0);
      }
    }



    clear_screen();
    cout << "\t\t\t\t--------------Updated Profile-----------------" << endl << endl;
    cout << "\n\n\t\t\t\t\t\t1.ID:" << p_ID << endl
         << "\t\t\t\t\t2.First Name:" << fname << endl
         << "\t\t\t\t\t3.Last Name:" << lname << endl
         << "\t\t\t\t\t4.Age" << p_age << endl
         << "\t\t\t\t\t5.Phone No.:" << phnumber << endl
         << "\t\t\t\t\t6.Occupation:" << occupation << endl
         << "\t\t\t\t\t7.City:" << p_city << endl
         << "\t\t\t\t\t8.Disease/Symptoms:" << dis << endl
         << "\t\t\t\t\t9.Admission date(DD/MM/YYYY):" << addDate << endl
         << "\t\t\t\t\t10.Release date(DD/MM/YYYY):" << relDate << endl
         << "\t\t\t\t\t11.Total Bill generated:" << bill << endl
         << "\t\t\t\t\t12.Kind of Specialization required:" << dept << endl << endl;
    fstream newObj;
    newObj.open("patient.txt", ios::app);
    newObj.seekp (0, ios::end);
    newObj << p_ID << endl << fname << endl << lname << endl
           << p_age << endl << phnumber << endl << occupation << endl
           << p_city << endl << dis << endl << addDate << endl
           << relDate << endl << bill << endl << dept << endl;
    newObj.close();

    //after completion
    cout << "\n\t\t\t\t\t" << "--------------------------------------------------" << endl;


    cout << "\n\n\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\tEnter 2 to
      Exit." << endl;
    cout << "\n\t\t\t\t\t";
    int c;
    cin >> c;
    if (c == 1) {
      clear_screen();
      getPatientDatabase();

    }

    else
      exit(0);
```

```cpp
1068  }
1069
1070  // Edit Patient Detail Function Ends Here
1071  //-----------------------------------------------------------------
1072
1073
1074
1075
1076
1077
1078
1079
1080  //the Display Particular Patient Function
1081  //-----------------------------------------------------------------
1082
1083
1084  void displayPat()
1085  {
1086    int x;
1087    cout << "\n\n\t\t\t\t\tEnter the ID of the Patient whose profile you want to Edit: " ;
1088    cin >> x;
1089    fstream patObj("patient.txt");
1090    patObj.seekg(0);
1091    string fname, lname, occupation, p_city, dis, addDate, relDate, dept;
1092    int p_age, p_ID, count = 0;
1093    long double phnumber, bill;
1094    while (patObj >> p_ID >> fname >> lname >> p_age >> phnumber >> occupation >> p_city
1095      >> dis >> addDate >> relDate >> bill >> dept)
1096    {
1096      if (p_ID == x) {
1097        cout << "\n\n\t\t\t\t\t1.ID:" << p_ID << endl
1098             << "\t\t\t\t\t2.First Name:" << fname << endl
1099             << "\t\t\t\t\t3.Last Name:" << lname << endl
1100             << "\t\t\t\t\t4.Age" << p_age << endl
1101             << "\t\t\t\t\t5.Phone No.:" << phnumber << endl
1102             << "\t\t\t\t\t6.Occupation:" << occupation << endl
1103             << "\t\t\t\t\t7.City:" << p_city << endl
1104             << "\t\t\t\t\t8.Disease/Symptoms:" << dis << endl
1105             << "\t\t\t\t\t9.Admission date(DD/MM/YYYY):" << addDate << endl
1106             << "\t\t\t\t\t10.Release date(DD/MM/YYYY):" << relDate << endl
1107             << "\t\t\t\t\t11.Total Bill generated:" << bill << endl
1108             << "\t\t\t\t\t12.Kind of Specialization required:" << dept << endl << endl;
1109        count++;
1110        break;
1111      }
1112    }
1113
1114    patObj.close();
1115    if (count == 0)
1116      cout << "\n\t\t\t\t\tNo matching records found!" << endl;
1117
1118    cout << "\t\t\t\t\t" << "------------------------------------------------" << endl;
1119
1120    cout << "\n\n\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\tEnter 2 to
1121      Exit." << endl;
1121    cout << "\n\t\t\t\t\t";
1122    int c;
1123    cin >> c;
1124    if (c == 1) {
1125      clear_screen();
1126      getPatientDatabase();
1127    }
1128
1129    else
1130      exit(0);
1131  }
1132  //the Display Particular Patient  Function Ends Here
1133  //-----------------------------------------------------------------
1134
1135
1136
1137
1138
```

```cpp
//Delete Patient function
//------------------------------------------------------------



void deletePat()
{
  int x;
  cout << "\n\n\t\t\t\t\tEnter the ID of the Patient whose profile you want to Edit: " ;
  cin >> x;
  fstream patObj("patient.txt");
  patObj.seekg(0);
  string fname, lname, occupation, p_city, dis, addDate, relDate, dept;
  int p_age, p_ID, count = 0;
  long double phnumber, bill;
  while (patObj >> p_ID >> fname >> lname >> p_age >> phnumber >> occupation >> p_city
    >> dis >> addDate >> relDate >> bill >> dept)
  {
    if (p_ID == x) {
      cout << "\n\n\t\t\t\t\t1.ID:" << p_ID << endl
          << "\t\t\t\t\t2.First Name:" << fname << endl
          << "\t\t\t\t\t3.Last Name:" << lname << endl
          << "\t\t\t\t\t4.Age" << p_age << endl
          << "\t\t\t\t\t5.Phone No.:" << phnumber << endl
          << "\t\t\t\t\t6.Occupation:" << occupation << endl
          << "\t\t\t\t\t7.City:" << p_city << endl
          << "\t\t\t\t\t8.Disease/Symptoms:" << dis << endl
          << "\t\t\t\t\t9.Admission date(DD/MM/YYYY):" << addDate << endl
          << "\t\t\t\t\t10.Release date(DD/MM/YYYY):" << relDate << endl
          << "\t\t\t\t\t11.Total Bill generated:" << bill << endl
          << "\t\t\t\t\t12.Kind of Specialization required:" << dept << endl << endl;
      count++;
      break;
    }
  }

  patObj.close();
  if (count == 0) {
    cout << "\n\t\t\t\t\t\tNo matching records found!" << endl;
    cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\t\tEnter
    2 to Exit." << endl;
    cout << "\t\t\t\t\t\t";
    int c;
    cin >> c;
    if (c == 1) {
      clear_screen();
      getPatientDatabase();
    }

    else
      exit(0);
  }

  cout << "\t\t\t\t\tAre you sure you want to delete this profile?\n\t\t\t\t\t1.YES\n\t\
    t\t\t2.NO" << endl;
  cout << "\n\t\t\t\t\t";
  int n; cin >> n;
  if (n == 1)
  {
    fstream file("patient.txt");
    file.seekg (0, ios::beg);
    file.seekg(0);
    int line;
    ofstream tempObj;
    tempObj.open("copy2.txt");
    tempObj.seekp(0);
    int skip = 0;
    string fname1, lname1, occupation1, p_city1, dis1, addDate1, relDate1, dept1;
    int p_age1;
    long double phnumber1, bill1;
    file.seekg(0); file.seekg(0, ios::cur);


```

```cpp
1209        while (file >> line >> fname1 >> lname1 >> p_age1 >> phnumber1 >> occupation1 >>
           p_city1 >> dis1 >> addDate1 >> relDate1 >> bill1 >> dept1)
1210        {
1211          if (line != x) {
1212            tempObj << line << endl << fname1 << endl << lname1
1213                    << endl << p_age1 << endl << phnumber1 << endl << occupation1
1214                    << endl << p_city1 << endl << dis1 << endl << addDate1 << endl
1215                    << relDate1 << endl << bill1 << endl << dept1 << endl;
1216          }
1217        }
1218
1219        file.close();
1220        tempObj.close();
1221        remove("patient.txt");
1222        int result;
1223        char oldname[] = "copy2.txt";
1224        char newname[] = "patient.txt";
1225        result = rename( oldname , newname );
1226        clear_screen();
1227        cout << "\n\t\t\t\t\tFile successfully deleted" << endl;
1228
1229      }
1230      clear_screen();
1231
1232      cout << "\t\t\t\t\t" << "-------------------------------------------------" << endl;
1233
1234      cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\t\tEnter 2
           to Exit." << endl;
1235      cout << "\t\t\t\t\t\t";
1236      int c;
1237      cin >> c;
1238      if (c == 1) {
1239        clear_screen();
1240        getPatientDatabase();
1241
1242      }
1243
1244      else
1245        exit(0);
1246  }
1247  // Delete Patient Function Ends Here
1248  //------------------------------------------------------------------
1249
1250
1251
1252
1253  //the Display Patient Database function
1254  //------------------------------------------------------------------
1255
1256
1257  void dispPatDatabase()
1258  {
1259      fstream Obj("patient.txt");
1260      Obj.seekg(0);
1261      cout << "\t\t\t\t\t\tPATIENT DATABASE" << endl << endl;
1262
1263      string fname, lname, occupation, p_city, dis, addDate, relDate, dept;
1264      int p_age, p_ID, count = 0;
1265      long double phnumber, bill;
1266      while (Obj >> p_ID >> fname >> lname >> p_age >> phnumber >> occupation >> p_city >>
           dis >> addDate >> relDate >> bill >> dept)
1267      { count++;
1268        cout << "\n\t\t\t\t\t\tPatient Profile:" << count << endl;
1269        cout << "\n\n\t\t\t\t\t\t1.ID:" << p_ID << endl
1270             << "\t\t\t\t\t2.First Name:" << fname << endl
1271             << "\t\t\t\t\t3.Last Name:" << lname << endl
1272             << "\t\t\t\t\t4.Age" << p_age << endl
1273             << "\t\t\t\t\t5.Phone No.:" << phnumber << endl
1274             << "\t\t\t\t\t6.Occupation:" << occupation << endl
1275             << "\t\t\t\t\t7.City:" << p_city << endl
1276             << "\t\t\t\t\t8.Disease/Symptoms:" << dis << endl
1277             << "\t\t\t\t\t9.Admission date(DD/MM/YYYY):" << addDate << endl
1278             << "\t\t\t\t\t10.Release date(DD/MM/YYYY):" << relDate << endl
```

```cpp
                << "\t\t\t\t\t11.Total Bill generated:" << bill << endl
                << "\t\t\t\t\t12.Kind of Specialization required:" << dept << endl << endl;


    }

    Obj.close();

    if (count == 0) {
      cout << "\n\t\t\t\t\t\tNo matching records found!" << endl;
    }

    cout << "\t\t\t\t\t" << "-------------------------------------------------" << endl;

    cout << "\n\n\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\tEnter 2 to
      Exit." << endl;
    cout << "\n\t\t\t\t\t";
    int c;
    cin >> c;
    if (c == 1) {
      clear_screen();
      getPatientDatabase();
    }

    else
      exit(0);

}

//the Display Patient Database Ends Here
//----------------------------------------------------------------


// ----------------------------------------------------------------

//Number of Patient function
//----------------------------------------------------------------
void numberofPats()
{

    fstream Obj("patient.txt");
    Obj.seekg(0);
    string fname, lname, occupation, p_city, dis, addDate, relDate, dept;
    int p_age, p_ID, count = 0;
    long double phnumber, bill;
    while (Obj >> p_ID >> fname >> lname >> p_age >> phnumber >>
            occupation >> p_city >> dis >> addDate >> relDate >> bill >> dept)
    {
      count++;
    }

    Obj.close();
    cout << "\n\n\t\t\t\t\tThe Total number of Patients in the hospital=" << count << endl
       << endl;

    cout << "\t\t\t\t\t" << "-------------------------------------------------" << endl;

    cout << "\n\n\t\t\t\t\t\tEnter 1 to return to Patient Database\n\n\t\t\t\t\t\tEnter 2
      to Exit." << endl;
    cout << "\n\t\t\t\t\t\t";
    int c;
    cin >> c;
    if (c == 1) {
      clear_screen();
      getPatientDatabase();
    }

    else
      exit(0);

}
// Number Of Patient Functions Ends Here
//----------------------------------------------------------------
```

```cpp
// Bed Class
//----------------------------------------------------------------

class bed {
public:
  int total = 1500;
  friend void getStatus();
};
//----------------------------------------------------------------




// Get Status of Bed Function
// ----------------------------------------------------------------

void getStatus()
{ bed b;
  cout << "\n\t\t\t\t\t\tTotal Number of Beds in the hospital:" << b.total << endl;
  fstream file("patient.txt");
  file.seekg(0);
  string fname, lname, occupation, p_city, dis, addDate, relDate, dept;
  int p_age, p_ID, count = 0;
  long double phnumber, bill;
  while (file >> p_ID >> fname >> lname >> p_age >> phnumber
             >> occupation >> p_city >> dis >> addDate >> relDate >> bill >> dept)
  {
     count++;
  }

  file.close();
  cout << "\n\n\t\t\t\t\t\tThe number of Occupied Beds=" << count << endl;
  cout << "\n\t\t\t\t\t\tThe number of Empty Beds=" << b.total - count << endl;

  cout << "\t\t\t\t\t" << "-----------------------------------------------" << endl;

  cout << "\n\t\t\t\t\t\tEnter 1 to return to Bed Database\n\t\t\t\t\t\tEnter 2 to Exit.
    " << endl;
  cout << "\t\t\t\t\t\t";
  int c;
  cin >> c;
  if (c == 1) {
     clear_screen();
     getBedDatabase();
  }

  else
     exit(0);

}
//----------------------------------------------------------------




//the main function

password A;

int main()
{
  A.getPassword();
  clear_screen();

  cout << "\t\t\t\t\tWELCOME TO MEDANTA HOSPITAL MANAGEMENT SYSTEM " << endl;
  cout << endl << endl;
  cout << "\t\t\t\t\t" << "-----------------------------------------------" << endl;

  cout << "\t\t\t\t\t\t\t1.Main Menu" << endl;
  cout << "\t\t\t\t\t\t\t2.Exit" << endl;
```

```cpp
1421      cout << "\t\t\t\t\t---------------------------------------------" << endl << endl;
1422      cout << "\t\t\t\t\t\tEnter your choice:";
1423      int choice1;
1424      cin >> choice1;
1425
1426      if (choice1 == 1) {
1427         menu();
1428      }
1429      else if (choice1 == 2) {
1430         return 0;
1431      }
1432      else
1433         cout << "\nCome on!Enter a valid option." << endl;
1434
1435
1436
1437  }
1438
1439  //----------------------------------------------------------------
1440
1441
1442
1443
1444  void menu()
1445  {
1446      clear_screen();
1447      int var = chooseMain();
1448      clear_screen();
1449      switch (var) {
1450      case 1: {
1451         A.passwordProtect();
1452         clear_screen();
1453         getDoctorDatabase();
1454         break;
1455      }
1456
1457      case 2: {
1458         A.passwordProtect();
1459         clear_screen();
1460         getPatientDatabase();
1461         break;
1462
1463      }
1464
1465      case 3: {
1466         A.passwordProtect();
1467         clear_screen();
1468         getBedDatabase();
1469         break;
1470      }
1471
1472      case 4:
1473      {
1474         exit(0);
1475         break;
1476      }
1477      default:
1478         cout << "\nCome on!Enter a valid option." << endl;
1479      }
1480
1481  }
1482
1483
1484  //the chooseMain function
1485  //----------------------------------------------------------------
1486
1487
1488
1489  int chooseMain() {
1490
1491      int choice2;
1492      cout << "\n\t\t\t\t\t\tMAIN MENU" << endl << endl << endl;
1493      cout << "\t\t----------------------------------------------------------------------
```

21

```cpp
                    " << endl;
  cout << endl;
  cout << "\t\t\t\t\t1.Enter Into Doctors' Database" << endl;
  cout <<  "\t\t\t\t\t2.Enter Into Patients' Database" << endl;
  cout <<  "\t\t\t\t\t3.Check for Hospital Bed Availability" << endl;
  cout << "\t\t\t\t\t4.Exit" << endl << endl;


  cout << "\t\t----------------------------------------------------------------
                    " << endl;


  cout << "\n\t\t\tEnter the number corresponding to your choice: " ;
  cin >> choice2;
  return choice2;

}


//---------------------------------------------------------------


// getDoctorDatabase
// ---------------------------------------------------------------
void getDoctorDatabase()
{

  int choice_doc;
  cout << "\t\t\t\t\tWELCOME TO DOCTOR'S DATABASE" << endl << endl;
  cout << "\t\t----------------------------------------------------------------
                    " << endl;

  cout << "\t\t\t\t\t1.Add New Doctor's Information" << endl;
  cout << "\t\t\t\t\t2.Edit a Doctor's Information" << endl;
  cout << "\t\t\t\t\t3.Display a Doctor's Information" << endl;
  cout << "\t\t\t\t\t4.Delete a Doctor's Information" << endl;
  cout << "\t\t\t\t\t5.Display Entire Doctor Database" << endl;
  cout << "\t\t\t\t\t6.Total Number of Doctors" << endl;
  cout << "\t\t\t\t\t7.Go back to Main Menu" << endl << endl;

  cout << "\t\t----------------------------------------------------------------
                    " << endl;


  cout << "\n\t\t\tEnter the number corresponding to your choice:  ";
  cin >> choice_doc;
  clear_screen();
  switch (choice_doc) {
  case 1: {
    addDoc();
    break;
  }
  case 2: {
    doctor k;
    editDoc(k);
    break;
  }
  case 3: {
    displayDoc();
    break;
  }
  case 4: {
    deleteDoc();
    break;
  }
  case 5: {
    dispDatabase();
    break;
  }
  case 6: {

    numberofDocs();
    break;
```

```cpp
    }

  case 7: {

    menu();
    break;
  }

  }
}



//the getPatientDatabase function
//-----------------------------------------------------------------
void  getPatientDatabase() {
  int choice_pat;
  cout << "\n\t\t\t\t\tWELCOME TO PATIENT'S DATABASE" << endl << endl;
  cout << "\t\t-----------------------------------------------------------------------
    " << endl;
  cout << endl;
  cout << "\t\t\t\t\t1.Add New Patient's Information" << endl;
  cout << "\t\t\t\t\t2.Edit a Patient's Information" << endl;
  cout << "\t\t\t\t\t3.Display a Patient's Information" << endl;
  cout << "\t\t\t\t\t4.Delete a Patient's Information" << endl;
  cout << "\t\t\t\t\t5.Display Entire Patient Database" << endl;
  cout << "\t\t\t\t\t6.Total Number of Patients" << endl;
  cout << "\t\t\t\t\t7.Check for doctor availability" << endl;
  cout << "\t\t\t\t\t8.Go back to Main Menu" << endl << endl;

  cout << "\t\t-----------------------------------------------------------------------
    " << endl;


  cout << "\n\t\t\tEnter the number corresponding to your choice: ";
  cin >> choice_pat;
  clear_screen();
  switch (choice_pat) {
  case 1: {
    addPat();
    break;
  }
  case 2: {
    patient p;
    editPat(p);
    break;
  }
  case 3: {
    displayPat();
    break;
  }
  case 4: {
    deletePat();
    break;
  }
  case 5: {
    dispPatDatabase();
    break;
  }
  case 6: {
    numberofPats();
    break;
  }
  case 7: {
    getDoctor();
    break;
  }


  case 8: {
    menu();
    break;
  }
```

```cpp
  }
}


//the getBedDatabase function
//---------------------------------------------------------------
void getBedDatabase() {
  int choice_bed;
  cout << "\n\t\t\t\t\t\tWELCOME TO BED DATABASE" << endl << endl;
  cout << "\t\t----------------------------------------------------------------------
    " << endl;
  cout << endl;
  cout << "\t\t\t\t\t\t1.Check Bed Status" << endl;
  cout << "\t\t\t\t\t\t2.Go back to Main Menu" << endl << endl;
  cout << "\t\t----------------------------------------------------------------------
    " << endl;

  cout << "\n\t\t\tEnter the number corresponding to your choice:  ";
  cin >> choice_bed;
  clear_screen();
  switch (choice_bed)
  {
  case 1: {
    getStatus();
    break;
  }
  case 2: {
    menu();
    break;
  }
  }
}


void clear_screen()
{
#ifdef _WIN32
  system("cls");
#else
  system("clear");
#endif
}
```