**Hospital Management System using C++**

**by Dharaneish V C**

**ABSTRACT:**

The purpose of the project entitled as “HOSPITAL MANAGEMENT SYSTEM “is to computerize the Front Office Management of Hospital to develop software which is user friendly, simple, fast, and cost – effective. It deals with the collection of patient’s information, diagnosis details etc. Traditionally, it was done manually. The main function of the system is register and store patient details and retrieve these details as and when required, and also to manipulate these details.

Meaningfully System input contains patient details, diagnosis details, while system output is to get these details on to the screen. The Hospital Management System can be entered using a username and password. It is accessible either by an administrator or receptionist. Only they can add data into the database. The data can be retrieved easily. The data are well protected for personal use and makes the data processing very fast.

**1. INTRODUCTION**

**1.1. About C++**

C++ is an object oriented programming (OOP) language, developed by Bjarne Stroustrup, and is an extension of the C language. It is therefore possible to code C++ in a "C style" or "object-oriented style." In certain scenarios, it can be coded in either way and is thus an effective example of a hybrid language.

C++ is a general purpose object oriented programming language. It is considered to be an intermediate level language, as it encapsulates both high and low level language features. Initially, the language was called 'C with classes’ as it had all properties of C language with an additional concept of 'classes’. However, it was renamed to C++ in 1983.

It is pronounced "C-Plus-Plus". C++ is one of the most popular languages primarily utilized with system/application software, drivers, client-server applications and embedded firmware.

The main highlight of C++ is a collection of pre-defined classes, which are data types that can be instantiated multiple times. The language also facilitates declaration of user defined classes. Classes can further accommodate member functions to implement specific functionality. Multiple objects of a particular class can be defined to implement the functions within the class. Objects can be defined as instances created at run time. These classes can also be inherited by other new classes which take in the public and protected functionalities by default.

C++ includes several operators such as comparison, arithmetic, bit manipulation, logical operators etc. One of the most attractive features of C++ is that it enables the overloading of certain operators such as addition.   
  
A few of the essential concepts within C++ programming language include polymorphism, virtual and friend functions, templates, namespaces and pointers.

**1.2. Overview of Files Concept**

**1.2.1. File handling:**

File handling concept in C++ language is used for store a data permanently in computer. Using file handling we can store our data in Secondary memory (Hard disk).

**1.2.2. Use of file handling:**

* For permanent storage.
* The transfer of input - data or output - data from one computer to another can be easily done by using files.

For read and write from a file you need another standard C++ library called stream which defines three new data types:

|  |  |
| --- | --- |
| **Data type** | **Description** |
| Ofstream | This is used to create a file and write data on files |
| Ifstream | This is used to read data from files |
| Fstream | This is used to both read and write data from/to files |

**1.2.3. How to achieve File Handling**

For achieving file handling in C++ we need follow following steps

* Naming a file
* Opening a file
* Reading data from file
* Writing data into file
* Closing a file

**1.2.4. Functions use in File Handling**

|  |  |
| --- | --- |
| **Function** | **Operation** |
| open() | To create a file |
| close() | To close an existing file |
| get() | Read a single character from a file |
| put() | write a single character in file. |
| read() | Read data from file |
| write() | Write data into file. |

**1.2.5. Defining and Opening a File**

The function open() can be used to open multiple files that use the same stream object.

**1.2.6. Closing a File**

A file must be close after completion of all operation related to file. For closing file we need **close()** function.

**1.2.7. File Opening mode**

|  |  |  |
| --- | --- | --- |
| **Mode** | **Meaning** | **Purpose** |
| ios :: out | Write | Open the file for write only. |
| ios :: in | read | Open the file for read only. |
| ios :: app | Appending | Open the file for appending data to end-of-file. |
| ios :: ate | Appending | take us to the end of the file when it is opened. |

Both ios :: app and ios :: ate take us to the end of the file when it is opened. The difference between the two parameters is that the ios :: app allows us to add data to the end of file only, while ios :: ate mode permits us to add data or to modify

the existing data anywhere in the file. The mode can combine two or more parameters using the bitwise OR operator (symbol |)

**1.2.8. File pointer**

Each file has two associated pointers known as the file pointers. One of them is called the input pointer (or get pointer) and the other is called the output pointer (or put pointer). The input pointer is used for reading the contents of a given file location and the output pointer is used for writing to a given file location.

**1.2.9. Function for manipulation of file pointer**

When we want to move file pointer to desired position then use these function for manage the file pointers.

|  |  |
| --- | --- |
| **Function** | **Operation** |
| seekg() | moves get pointer (input) to a specified location. |
| seekp() | moves put pointer (output) to a specified location. |
| tellg() | gives the current position of the get pointer. |
| tellp() | gives the current position of the put pointer. |
| fout . seekg(0, ios :: beg) | go to start |
| fout . seekg(0, ios :: cur) | stay at current position |
| fout . seekg(0, ios :: end) | go to the end of file |
| fout . seekg(m, ios :: beg) | move to m+1 byte in the file |
| fout . seekg(m, ios :: cur) | go forward by m bytes from the current position |
| fout . seekg(-m, ios :: cur) | go backward by m bytes from the current position |
| fout . seekg(-m, ios :: end) | go backward by m bytes from the end |

**1.2.10. put() and get() function**

The function put() write a single character to the associated stream. Similarly, the function get() reads a single character from the associated stream.

**1.2.11. read() and write() function**

These function take two arguments. The first is the address of the variable V, and the second is the length of that variable in bytes. The address of variable must be cast to type char \* (i.e pointer to character type).

**2. SYSTEM REQUIREMNETS**

* Operating System: Windows XP, Vista, 7 and 8.
* Processor: Pentium III/1.4GHz Processor
* RAM: 512MB to 2GB
* Hard Disk Space: 900MB space required.
* Compiler: BORLAND / TURBO compiler

**3. OBJECTIVE OF THE PROJECT**

As the name suggests, this project is based on hospital management system. In this project one can:

* Create Patient’s record
* Modify Patient’s record
* Remove Patient’s record
* View Patient record
* And other Database functions.

The source code contains different header files, built-in functions, user-defined functions, switch statements, loop statements, etc.

The main objective of the Hospital Management system project is discipline of the planning, organizing and managing the Hospital tasks. Our project aims at making the task of Hospital easy. Hospital Management is entering the records of new patient and retrieving the details of patients available in the Database. We can delete Patient’s details when the patient left the hospital. We can modify patient’s details in case any error occurs. We can view token number of pending patients in various departments. In the project we can get feedback about the project from staff and read it by admin options. This project is secured and only members with UserID and password can access it. Every page is displayed with a Medical Tip or Quote useful to the users. Some options are restricted to Admin only.

**4. HEADER FILES PURPOSES**

**4.1. Header Files Used**

* fstream.h
* iomanip.h
* stdio.h
* conio.h
* string.h
* Process.h
* dos.h
* stdlib.h

**4.2. Header Files Purposes**

* fstream.h : This header file is used for defining several iostreams template classes that manipulate external files. This is a standard C++ library that handles reading from and writing to files either in text or in binary formats
* iomanip.h : This header file declares stream I/O manipulators and contains macros for creating parameterized manipulators.
* stdio.h : This header file defines types and macros needed for the standard I/O package.
* conio.h : This header file in c++ programming contains functions for console input/output. Some of the most commonly used functions of conio.h are clrscr(), getch(), etc.
* string.h : The string.h provides various functions for comparing, coping, concatenating, reversing, many other functions for various operations.
* process. h :process.h is a C header file which contains function declarations and macros used in working with threads and processes
* dos.h :The dos.h header file of c++ language contains functions for handling interrupts, producing sound, date and time functions etc. It is borland specific and works in turbo c++ compiler.
* stdlib.h :This header file in c++ programming contains some standard library functions such as random(),randomize().

**5. SOURCE CODE**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdlib.h>**

**#include<stdio.h>**

**#include<process.h>**

**#include<string.h>**

**#include<dos.h>**

**#include<iomanip.h>**

**#include<fstream.h>**

**int cb=4;**

**void admin();**

**struct day { short int date;**

**short int month;**

**int year;**

**};**

**void calc\_age(int date,int month,int year)**

**{**

**cout<<"\nAGE (as of Jan 2020) is "<<(2020-year-1)<<" years"<<(12-month)<<" months.";**

**}**

**void calc\_age(int month,int year) //Function Overloaded**

**{**

**cout<<"\nAGE (as of Jan 2020) is "<<(2020-year-1)<<" years"<<(12-month)<<" months.";**

**}**

**char tips[40][500]=**

**{**

**"A healthy outside starts from the inside.",**

**"Love yourself enough to live a healthy lifestyle.",**

**"To keep the body in good health is a duty, otherwise we shall not be able to keep our mind strong and clear.",**

**"Health is not valued till sickness comes.",**

**"Take care of your body. It's the only place you have to live.",**

**"Health is the greatest gift.",**

**"Health is the crown on the well persons head that only the ill person can see.",**

**"It is health that is real wealth and not pieces of gold and silver.",**

**"Those who think they have no time for healthy eating will sooner or later have to find time for illness.",**

**"You can't enjoy wealth if you're not in good health.",**

**"The mind and body are not separate. what affects one, affects the other.",**

**"Happiness is the highest form of health.",**

**"The first wealth is health.",**

**"Healthy citizens are the greatest asset any country can have.",**

**"There is no diet that will do what eating healthy does. Skip the diet. Just eat healthy.",**

**"You learn good health and family are it, and nothing else matters.",**

**"Your body deserves the best.",**

**"Take care of yourself, be healthy, and always believe you can be successful in anything you truly want.",**

**"If you're too busy to workout, your priorities need to change. I can't think of too many things more important than your health.",**

**"I'm healthy as can be – not an ache or pain. A lot of my prayer is thanking God that I am healthy. I pray for long life and good health.",**

**"The ingredients of health and long life, and great temperance, open air, easy labor, and little care.",**

**"A healthy mind does not speak ill of others.",**

**"What you eat literally becomes you. you have a choice in what you're made of.",**

**"Think positively and exercise daily, eat healthy, work hard, stay strong, build faith, worry less, read more, and be happy.",**

**"A fool does not care about his health.",**

**"We know that food is a medicine, perhaps the most powerful drug on the planet with the power to cause or cure most disease.",**

**"The food you eat can be either the safest and most powerful form of medicine or the slowest form of poison.",**

**"If it came from a plant, eat it. If it was made in a plant, don't.",**

**"If you keep good food in your fridge, you will eat good food.",**

**"When diet is wrong, medicine is of no use. When diet is correct, medicine is of no need.",**

**"You can't expect to look like a million bucks if you eat from the dollar menu.",**

**"Health requires healthy food.",**

**"Sufficient sleep, exercise, healthy food, friendship, and peace of mind are necessities, not luxuries.",**

**};**

**void disp()**

**{**

**randomize();**

**int n;**

**n=random(32);**

**cout<<tips[n];**

**}**

**void line()**

**{**

**cout<<"\n";**

**for(int i=0;i<=78;i++)**

**cout<<char(254);**

**cout<<"\n";**

**}**

**void newpage()**

**{**

**textbackground(cb);**

**textcolor(WHITE);**

**clrscr();**

**cout<<"\n\t\t\t\tABCD HOSPITALS";**

**cout<<"\n\t\t\t\t CHENNAI";**

**line();**

**disp();**

**line();**

**}**

**class feedback**

**{**

**private:**

**char name[30],town[20],mail[30],content[1000];**

**public:**

**void putfb()**

**{**

**cout<<"\nFEEDBACK:";**

**cout<<"\nName:"<<name;**

**cout<<"\nTown/City:"<<town;**

**cout<<"\nMail:"<<mail;**

**cout<<"\nContent:"<<content;**

**}**

**void getfb()**

**{**

**newpage();**

**cout<<"\nWe are glad to know your Feedback\n";**

**cout<<"\nEnter the following details:\nName:";**

**cin>>name;**

**cout<<"\nTown/City:";**

**cin>>town;**

**cout<<"\nMail:";**

**cin>>mail;**

**cout<<"\nContent(max 1000 characters):";**

**gets(content);**

**cout<<"\nYour Feedback has been send succesfully.THANK YOU!!!";**

**}**

**};**

**void exitpro()**

**{**

**for(int i=0,j=8;i<=8;i++,j--)**

**{**

**clrscr();**

**textbackground(i);**

**textcolor(j);**

**line();cout<<"\n\n";**

**cout<<"\n\n\t\tA PROJECT BY:\n\t\t\tASWATHKRISHNA.G\n\t\t\tDHARANEISH.V.C\n\t\t\tNANDAPRABHU.B\n\n\n\t\t\tXII-SCIENCE(2019-20)\n\n\t\tSAGAR INTERNATIONAL SENIOR SECONDARY SCHOOL\n\t\t\t\tPERUNDURAI";**

**cout<<"\n\n\n";line();cout<<"\t\tTHANK YOU...VISIT AGAIN.";**

**delay(500);**

**}**

**getch();**

**exit(0);**

**}**

**class log**

**{**

**char id[11],password[9];**

**public:**

**log()**

**{**

**newpage();**

**}**

**void signin();**

**}i1;**

**void log::signin()**

**{**

**cout<<"\n\nEnter UserName:";**

**gets(id);**

**char test[11],vf='n';**

**ifstream fin7("userid.dat",ios::in);**

**while((fin7))**

**{**

**fin7.read((char\*)&test,sizeof(test));**

**if((strcmp(test,id)==0))**

**{**

**vf='y';**

**}**

**}**

**fin7.close();**

**if(vf!='y')**

**{**

**cout<<"\nINVALID USERID!!!\nPress any key to exit";**

**getch();**

**exitpro();**

**}**

**cout<<"\nEnter Password(case sensitive) : ";**

**for(int i=0;i<8;i++)**

**{**

**password[i]=getch();**

**cout<<"\*";**

**}**

**if(!((strcmp(password,"computer")==0)))**

**{cout<<"\nINCORRECT PASSWORD!!!\nPress any key to exit";**

**getch();**

**exitpro();**

**}**

**cout<<"\nLogin Successful.Press any key.";**

**getch();**

**}**

**class personal**

**{**

**protected:**

**char name[30],attender[30],dept[40],town[20];**

**day dob; unsigned long long int mobile;**

**};**

**class patient:private personal**

**{**

**private:**

**void choose\_dept();**

**public:**

**patient()**

**{**

**token=0;**

**}**

**int token;**

**void new\_patient();**

**void find\_patient();**

**};**

**void patient::choose\_dept()**

**{**

**int ch=0;**

**char department[20][40]=**

**{**

**"Accident and emergency (A&E)\n",**

**"Cardiology\n",**

**"Coronary Care Unit (CCU)\n",**

**"Critical Care\n",**

**"Elderly services\n",**

**"Gastroenterology\n",**

**"General Services\n",**

**"Gynecology\n",**

**"Haematology\n",**

**"Intensive Care Unit (ICU)\n",**

**"Maternity\n",**

**"Nephrology\n",**

**"Neurology\n",**

**"Obstetrics/Gynecology\n",**

**"Oncology\n",**

**"Ophthalmology\n",**

**"Otolaryngology\n",**

**"Psycholopgy\n",**

**"Radiology\n",**

**"Urology\n"**

**};**

**for(int i=0;i<20;i++)**

**{**

**cout<<i+1<<"."<<department[i];**

**}**

**cout<<"\nCHOOSE YOUR DEPT:";cin>>ch;**

**if((ch>20) ||(ch<1))**

**{cout<<"\nINVALID CHOICE";**

**choose\_dept(); //Recursive Function**

**}**

**cout<<"\n\n"<<department[ch-1][40];**

**strcpy(dept,department[ch-1]);**

**}**

**void patient::new\_patient()**

**{**

**newpage();**

**cout<<"\nEnter the Patient's Details:";**

**tk:**

**cout<<"\nToken no(no duplicate entries allowed):";cin>>token;**

**patient p7;**

**int tkt=token;**

**ifstream fin6("dataentry.dat",ios::in);**

**fin6.read((char\*)&p7,sizeof(p7));**

**{**

**if(p7.token==tkt)**

**{**

**cout<<"\nToken Number already exists. Enter a different one";**

**goto tk;**

**}**

**}**

**cout<<"\nName:";gets(name);**

**cout<<"Enter DATE OF BIRTH:(day):";cin>>dob.date;**

**cout<<"(month):";cin>>dob.month;**

**cout<<"(Year):";cin>>dob.year;**

**cout<<"Attender's name:";gets(attender);**

**cout<<"\nChoose Department:";**

**choose\_dept();**

**newpage();**

**cout<<"\nName:"<<name;**

**calc\_age(dob.month,dob.year);**

**cout<<"\nAttender's name:"<<attender;**

**cout<<"\nDepartment:"<<dept;**

**cout<<"\nCity:";gets(town);**

**cout<<"\nMobile:";cin>>mobile;**

**}**

**void patient::find\_patient()**

**{**

**newpage();**

**cout<<"\nThe patient details are as follows:";**

**cout<<"\nToken no:"<<token;**

**cout<<"\nName:"<<name<<"\nAttender name:"<<attender<<"\nDepartment:"<<dept<<"\nCity:"<<town<<"\nMobile:"<<mobile;**

**calc\_age(dob.date,dob.month,dob.year);**

**}**

**void intro()**

**{**

**textcolor(GREEN);**

**textbackground(WHITE);**

**clrscr();**

**cout<<"\n\t\t\tABCD HOSPITALS WELCOMES YOU...";**

**cout<<"\n\n\n\n\t\t\t\tLOADING...";**

**cout<<"\n\t\t\t Please wait..\n\n\n";**

**for(int k=0;k<=78;k++)**

**{**

**cout<<char(219);**

**delay(50);**

**}**

**textcolor(WHITE);**

**textbackground(RED);**

**clrscr();**

**for(int j=0;j<10;j++)**

**{**

**clrscr();**

**gotoxy(j,3); cputs (" ABCD HOSPITALS");**

**gotoxy(j,5); cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,6); cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,7); cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,8); cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,9);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,10);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,11);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,12);cputs (" ÛÛÛÛÛ ÛÛÛÛÛ");**

**gotoxy(j,13);cputs (" MEDICAL ÛÛÛÛÛ ÛÛÛÛÛ CENTRE");**

**gotoxy(j,14);cputs (" ÛÛÛÛÛ ÛÛÛÛÛ");**

**gotoxy(j,15);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,16);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,17);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,18);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,19);cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**gotoxy(j,20); cputs (" ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛ");**

**delay(90);**

**}**

**gotoxy(24,23);**

**cout<<" Press any key to continue...";**

**getch();**

**}**

**void main()**

**{**

**intro();**

**a1:**

**textcolor(cb);**

**textbackground(WHITE);**

**clrscr();**

**newpage();int cho2;**

**cout<<"\n\n1.ADMIN\n2.STAFF\n3.CHANGE BACKGROUND COLOUR\nEnter your choice:";**

**cin>>cho2;**

**if(cho2==1)**

**{admin();exit(1);}**

**if(cho2==3)**

**{**

**int chc;**

**cout<<"\nChoose your color:\n1.Black\n2.Blue\n3.Light Green\n4.Pale Blue\n5.Red\n6.Violet\n7.Yellow";**

**cin>>chc;**

**cb=chc-1;**

**cout<<"\n\nChanged Successfully.";**

**goto a1;**

**}**

**i1.signin();**

**char c,cho='y';**

**start:**

**newpage();**

**cout<<"\n1.Add new patient\n2.Remove a patient\n3.Edit Patient details\n4.view Patient Details\n5.More Options\n6.Send feedback\n7.Logout\n8.exit\n\nEnter your choice:";**

**cin>>c;**

**switch(c)**

**{**

**case '1' :patient p1; cho='y';**

**ofstream fout1("dataentry.dat",ios::app);**

**while((cho=='y')||(cho=='Y'))**

**{cout<<"\nENTER NEW PATIENT DETAILS";**

**p1.new\_patient();**

**fout1.write((char\*)&p1,sizeof(p1));**

**cout<<"\nPatient Details saved successfully.";**

**cout<<"\nDo you want to continue:(y/n)";cin>>cho;**

**}**

**fout1.close();**

**goto start;**

**case '2' :patient p2;cho='y';**

**int tk;char confirm='n',found='n';**

**ofstream fout2("temp.dat",ios::out);**

**ifstream fin1("dataentry.dat",ios::in);**

**while((cho=='y')||(cho=='Y'))**

**{**

**found='n';**

**cout<<"\nEnter the Token no to be deleted :";**

**cin>>tk;**

**while(!(fin1.eof()))**

**{**

**fin1.read((char\*)&p2,sizeof(p2));**

**if(tk==p2.token)**

**{**

**found='y';**

**p2.find\_patient();**

**cout<<"\nAre you sure want to delete the above patient(y/n):";**

**cin>>confirm;**

**if((confirm=='n')||(confirm=='N'))**

**{**

**cout<<"\nPatient Details not deleted.";**

**fout2.write((char\*)&p2,sizeof(p2));**

**}**

**}**

**else**

**{**

**fout2.write((char\*)&p2,sizeof(p2));**

**}**

**}**

**if(found=='n')**

**{**

**cout<<"\nRecord not Found in the file.";**

**}**

**cout<<"\nDo you want to continue:(y/n)";**

**cin>>cho;**

**}**

**fin1.close();**

**fout2.close();**

**remove("dataentry.dat");**

**rename("temp.dat","dataentry.dat");**

**goto start;**

**case '3' :**

**{**

**patient p3,p4;**

**fstream fio3("dataentry.dat",ios::in|ios::out);**

**tk=0;long pos;cho='y';found='n';confirm='n';**

**while((cho=='y')||(cho=='Y'))**

**{**

**fio3.seekg(0);**

**found='n';confirm='n';**

**cout<<"\nEnter the Current Token no to be modified :";**

**cin>>tk;**

**while(!(fio3.eof()))**

**{**

**pos = fio3.tellg();**

**fio3.read((char\*)&p3,sizeof(p3));**

**if(tk==p3.token)**

**{**

**found='y';**

**p3.find\_patient();**

**cout<<"\nAre you sure want to modify the above patient details(y/n):";**

**cin>>confirm;**

**if((confirm=='n')||(confirm=='N'))**

**{**

**cout<<"\nPatient Details saved as it is.";**

**fio3.write((char\*)&p3,sizeof(p3));**

**}**

**if((confirm=='y')||(confirm=='Y'))**

**{**

**fio3.seekg(pos);**

**p4.new\_patient();**

**fio3.write((char\*)&p4,sizeof(p4));**

**cout<<"\nPatient Details saved successfully.";**

**break;**

**}**

**}**

**else**

**{**

**fio3.write((char\*)&p3,sizeof(p3));**

**}**

**}**

**if(found=='n')**

**{cout<<"\nRecord not Found in the file.";}**

**fio3.seekg(0);**

**cout<<"\nDo you want to continue:(y/n)";cin>>cho;**

**}**

**if(!((cho=='y')||(cho=='Y')))**

**{**

**goto start;**

**}**

**}**

**case '4' : patient p5;**

**ifstream fin4("dataentry.dat",ios::in|ios::nocreate);**

**cho='y';**

**while((cho=='y')||(cho=='Y'))**

**{**

**found='n';**

**cout<<"\nEnter the Current Token no to be viewed :";cin>>tk;**

**while(!(fin4.eof()))**

**{**

**fin4.read((char\*)&p5,sizeof(p5));**

**if(tk==p5.token)**

**{**

**found='y';**

**p5.find\_patient();**

**}**

**}**

**if(found=='n')**

**{cout<<"\nRecord not Found in the file.";}**

**cout<<"\nDo you want to continue:(y/n)";cin>>cho;**

**}**

**goto start;**

**case '5' :{**

**newpage();int opt;**

**cout<<"\n1.Delete the entire Database\n2.View all Patients one by one\n3.View All Pending Token Numbers\n Enter your choice:";cin>>opt;**

**if(opt==1)**

**{**

**unsigned long int access=0;**

**cout<<"Enter Access Code:";cin>>access;**

**if(access==2000)**

**{**

**remove("dataentry.dat");**

**cout<<"\nData Base Cleared Succesfully";**

**}**

**else**

**{**

**cout<<"\nWrong Password. Try Again!!!\nPress any key to go to Main Menu.";**

**getch();**

**goto start;**

**}**

**}**

**if(opt==2)**

**{**

**int i=0;patient p6;**

**ifstream fin5("dataentry.dat",ios::in|ios::nocreate);**

**fin5.seekg(0);**

**while(!(fin5.eof()))**

**{**

**fin5.read((char\*)&p6,sizeof(p6));**

**p6.find\_patient();**

**i++;cout<<"\nPress Enter to next.";getch();**

**}**

**cout<<"\nEnd of the File Reached\nTotal No of Patients:"<<i;**

**}**

**if(opt==3)**

**{**

**int arr[100],i=0; patient p8;**

**ifstream fin6("dataentry.dat",ios::in|ios::nocreate);**

**fin6.seekg(0);**

**int size=0;**

**while(!fin6.eof())**

**{**

**fin6.read((char\*)&p8,sizeof(p8));**

**arr[i]=p8.token;**

**i++;**

**}**

**size=i-1;**

**{**

**int small=0,pos,tmp;**

**for(int i=0;i<size-1;i++)**

**{**

**small=arr[i];**

**pos=i;**

**for(int j=i+1;j<size;j++)**

**{**

**if(arr[j]<small)**

**{small=arr[j];pos=j;}**

**}**

**tmp=arr[i];**

**arr[i]=arr[pos];**

**arr[pos]=tmp;**

**}**

**cout<<"\nThe Token Numbers in database are (in ascending order): ";**

**for(int j=0;j<size;j++)**

**cout<<arr[j]<<",";**

**}**

**}**

**cout<<"\nPress any key to goto main menu";**

**getch();goto start;**

**}**

**case '6' : feedback f1;**

**f1.getfb();**

**ofstream foutfb("feedback.dat",ios::app);**

**foutfb.write((char\*)&f1,sizeof(f1));**

**goto start;**

**case '7' : {**

**char cho3='n';**

**cout<<"\nDo you want to Logout??(y/n)";**

**cin>>cho3;**

**if((cho3=='y')||(cho3=='Y'))**

**main();**

**else**

**goto start;**

**break;**

**}**

**case '8' : exitpro();**

**break;**

**default : {**

**cout<<"\nPlease enter a correct choice!!!"; getch(); goto start;**

**}**

**}**

**getch();**

**}**

**void admin()**

**{**

**int code,i=3;**

**newpage();**

**cout<<"\n\nADMIN OPTIONS(Authorised person only)";**

**p1:**

**cout<<"\n\nEnter Code to continue:";cin>>code;**

**if(code!=2000)**

**{**

**if(i>1)**

**{**

**cout<<"\nWRONG CODE!!!!\nPress any key to Try again."<<i<<" tries left.";**

**getch();i--;**

**goto p1;**

**}**

**if(i==1)**

**{**

**cout<<"\nWRONG CODE!!!\nYOu have crossed the maximun tries.Press any key to goto Main Menu.";**

**exitpro();**

**}**

**}**

**p2:**

**int c;**

**cout<<"\nVerify as HUMAN";int cp1,cp2;**

**for(i=0;i<3;i++)**

**{**

**cout<<'\n'<<3-i<<" tries left";**

**randomize();**

**cp1=1000+random(999);**

**cout<<"\nCapcha:"<<cp1;**

**cout<<"\nEnter The Number above:";cin>>cp2;**

**if(cp1==cp2)**

**{**

**cout<<"\nVerified Successfully.";**

**goto p3;**

**}**

**else**

**{**

**cout<<"\nWrong code!!Try again.";**

**}**

**}**

**if(i==3)**

**{**

**cout<<"\nYou could not be Verified.";**

**getch();goto p1;**

**}**

**p3:**

**newpage();**

**cout<<"\n1.Sign up\n2.View all User IDs\n3.Feedback reading\n4.Delete Feedback file\n5.Delete entire Database\n6.Logout\n7.Exit\nEnter your choice:";cin>>c;**

**switch(c)**

**{**

**case 1:**

**{ char cho='y';**

**while((cho=='y')||(cho=='Y'))**

**{**

**char id1[11];**

**cout<<"\nEnter User ID to be added: ";**

**gets(id1);**

**ofstream fout("userid.dat",ios::app);**

**fout.write((char\*)&id1,sizeof(id1));**

**cout<<"\nUser ID saved successfully\n\nDo you want to Continue(y/n):";**

**cin>>cho;**

**}**

**cout<<"\nPress any key to go to Menu.";**

**getch();goto p3;**

**break;**

**}**

**case 2:**

**{**

**char id2[11];**

**ifstream fin("userid.dat",ios::in);**

**cout<<"\nUser IDs:\n";**

**while(!(fin.eof()))**

**{**

**fin.read((char\*)&id2,sizeof(id2));**

**puts(id2);**

**}**

**cout<<"\nPress any key to go to Menu.";**

**getch();goto p3;**

**}**

**break;**

**case 3:**

**{**

**feedback f2;**

**ifstream finfb("feedback.dat",ios::in);**

**while(!(finfb.eof()))**

**{**

**finfb.read((char\*)&f2,sizeof(f2));**

**f2.putfb();**

**cout<<"\nPress Enter for Next. ";getch();**

**}**

**cout<<"\nPress any key to go to Menu.";**

**getch();goto p3;**

**break;**

**}**

**case 4:**

**{**

**char cho;**

**cout<<"\nDo you want to Delete Entire Feedback Data(y/n).";**

**cin>>cho;**

**if((cho=='y')||(cho=='Y'))**

**{**

**remove("feedback.dat");**

**cout<<"\nFeedback DataBase Cleared Successfully.";**

**cout<<"\nPress any key to go to Menu.";**

**getch();goto p3;**

**}**

**else**

**{**

**goto p3;**

**}**

**break;**

**}**

**case 5:**

**{**

**{**

**char cho4;**

**cout<<"\nAre you sure you want to Delete the Entire DataBase?(y/n):";**

**cin>>cho4;**

**if((cho4=='y')||(cho4=='Y'))**

**{**

**remove("database.dat");**

**cout<<"\nData base cleared successfully.";**

**cout<<"\nPress any key to go to Menu.";**

**getch();goto p3;**

**}**

**else**

**{**

**goto p3;**

**}**

**}**

**}**

**case 6:**

**{**

**main();**

**}**

**case 7:**

**{**

**exitpro();**

**}**

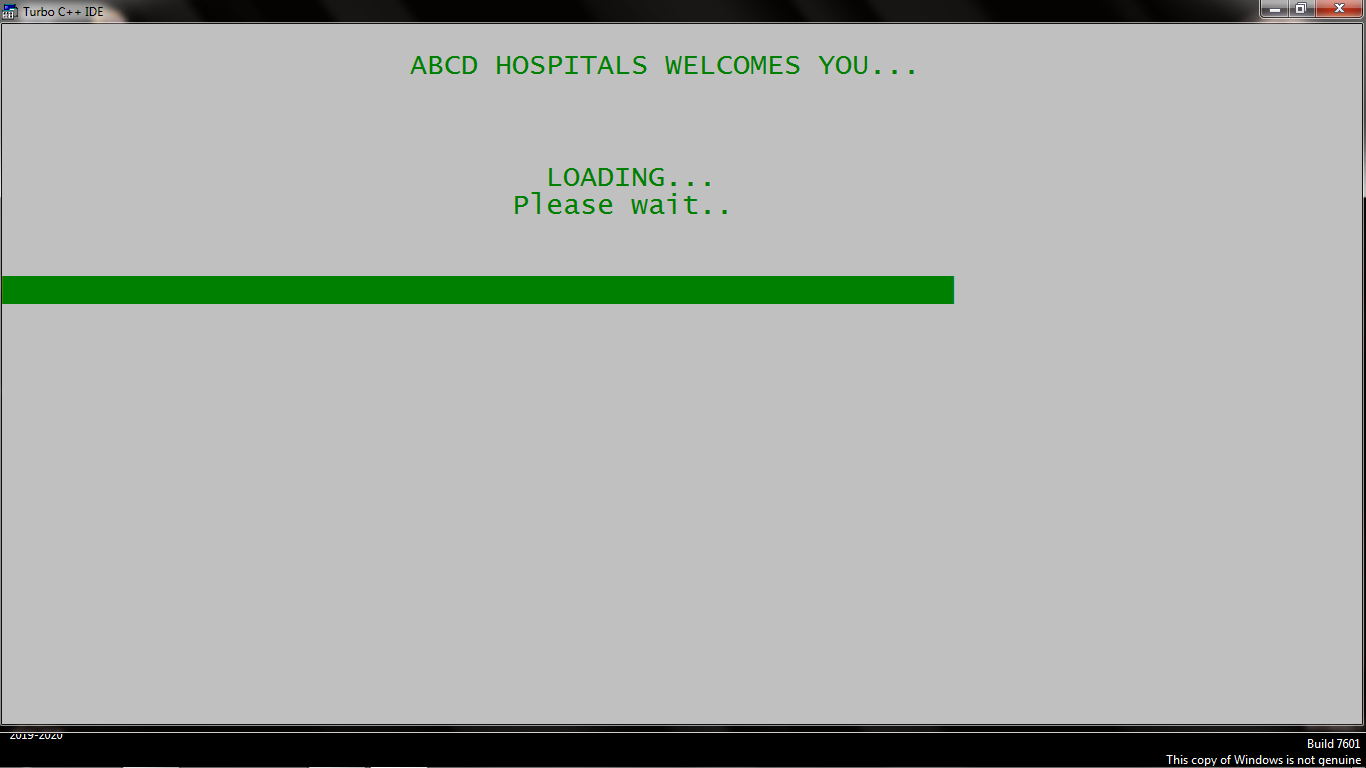
**break;**

**}**

**}**

**6. RESULTS AND DISCUSSION**

**6.1.Results**

**Welcome 1: **

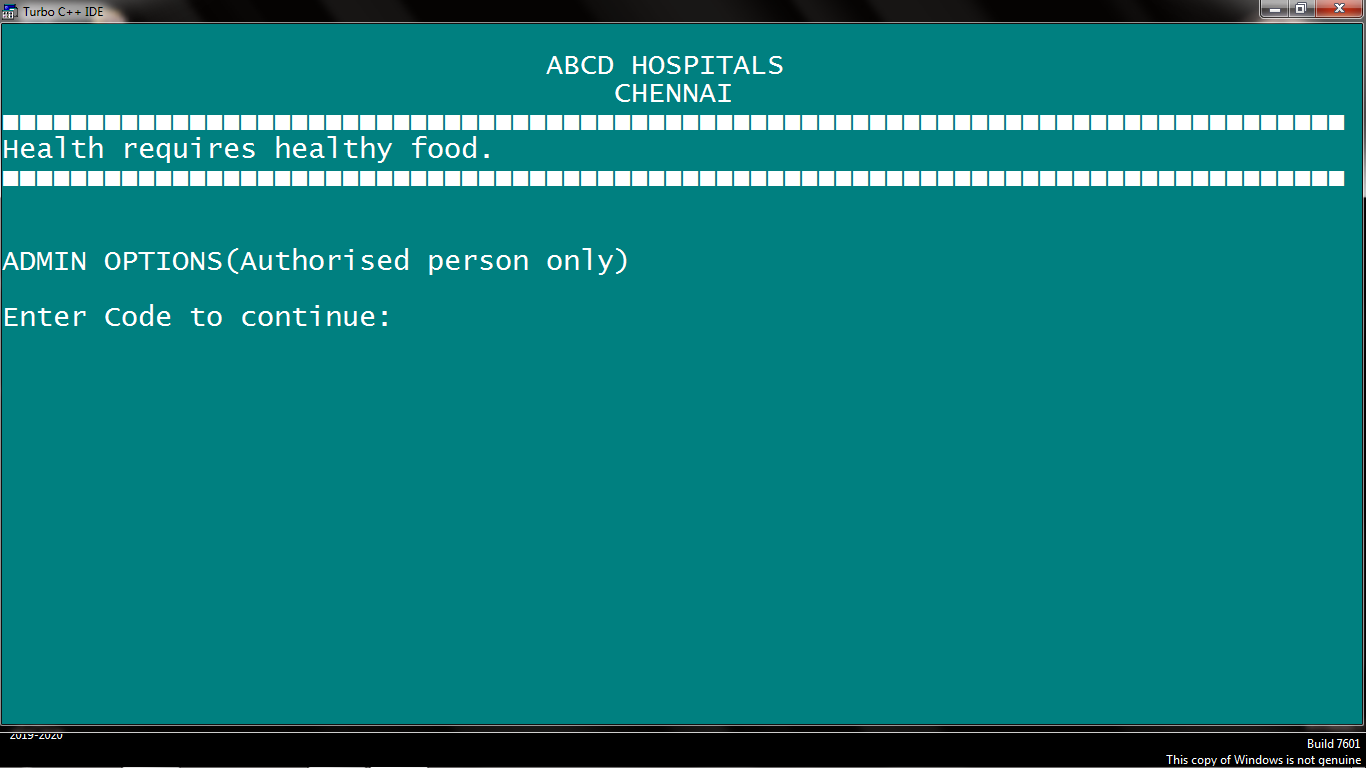
**Welcome 2:**

**Menu :1**

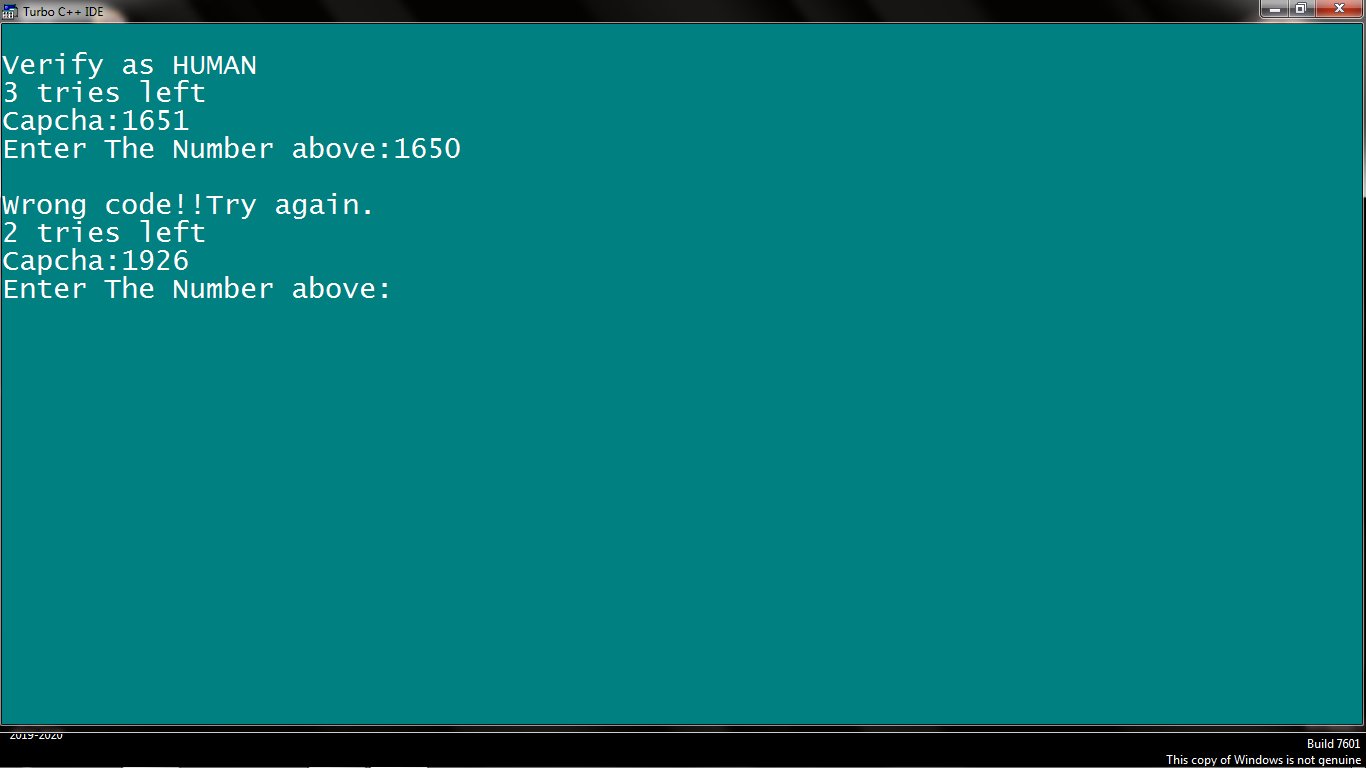
****

**Admin:**

**Security code:**

****

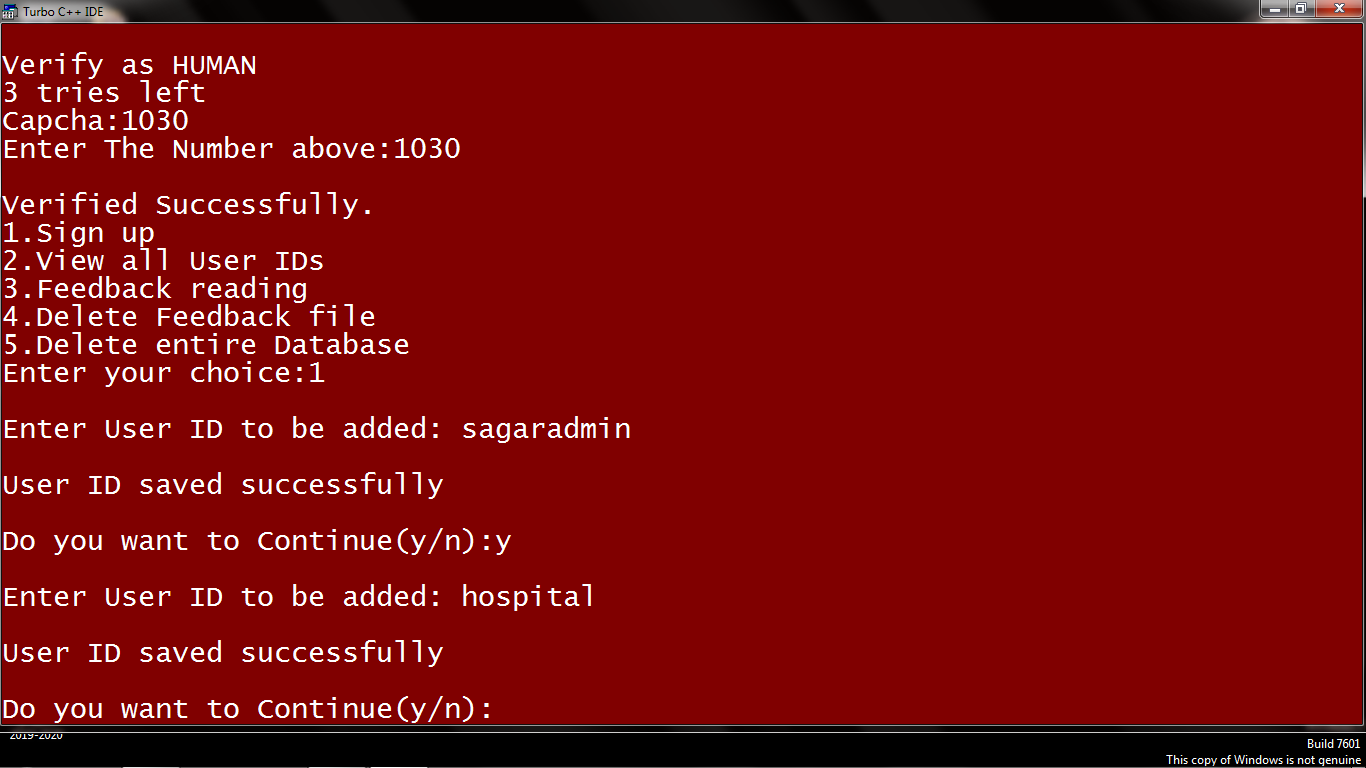
**Captcha verification:**

****

**Admin options:**

****

**Sign up:**

****

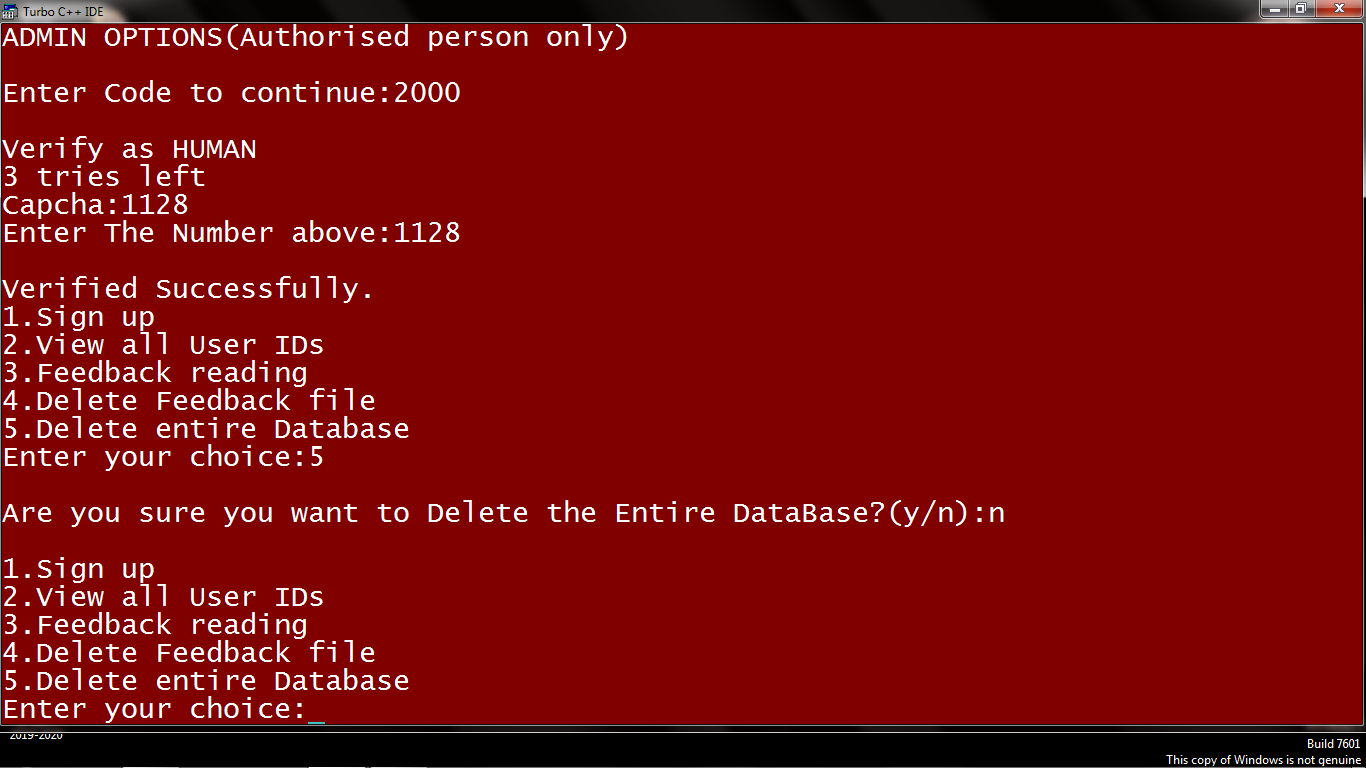
**View all users ID’s:**

****

**Feedback reading:**

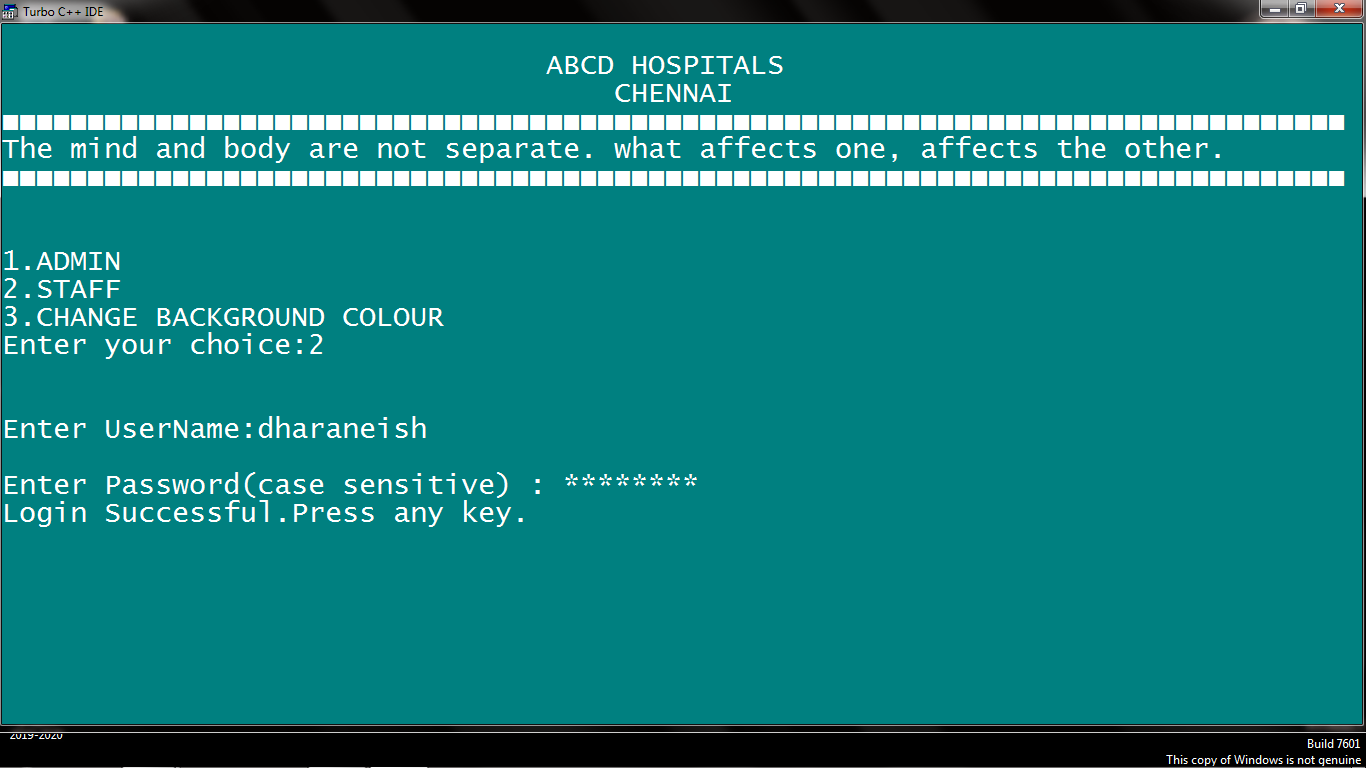
****

**Deleting files:**

****

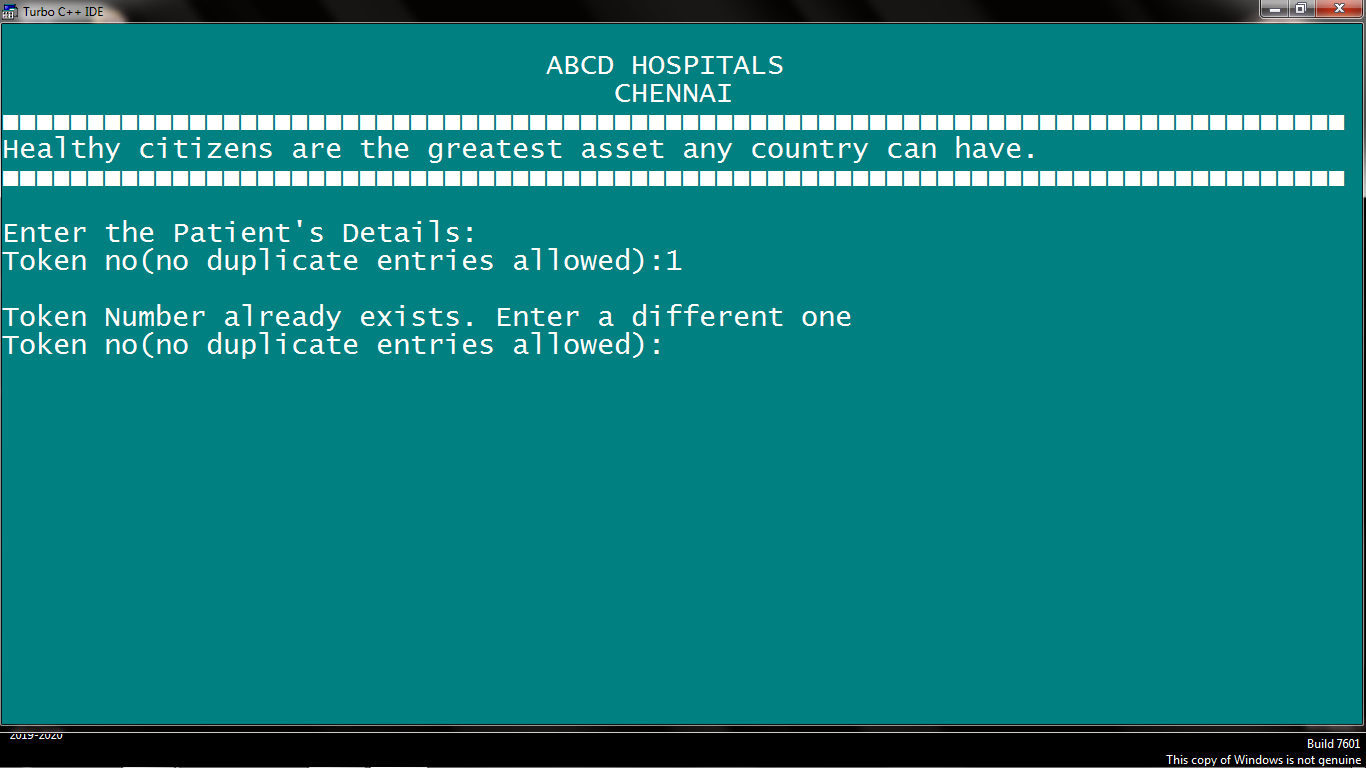
**Staff**

**Security system**

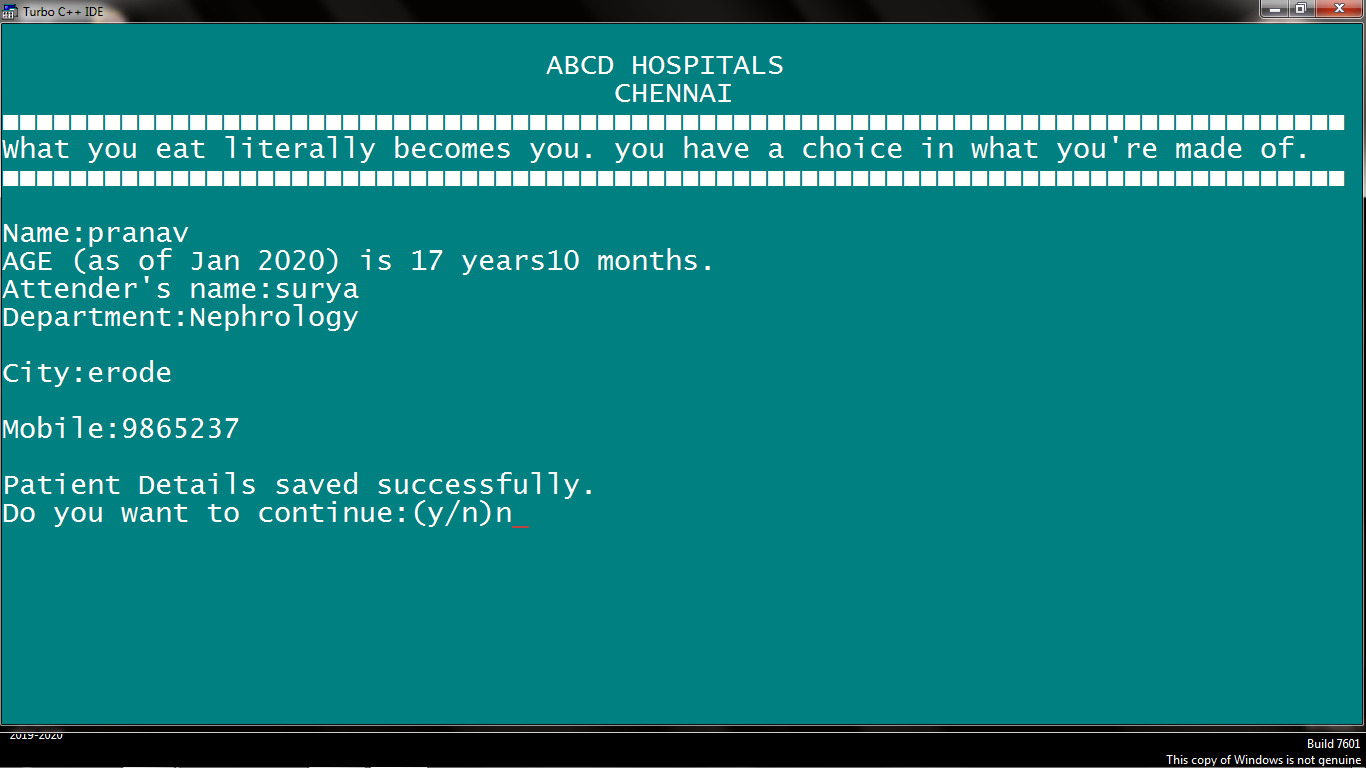
****

**Staff options**

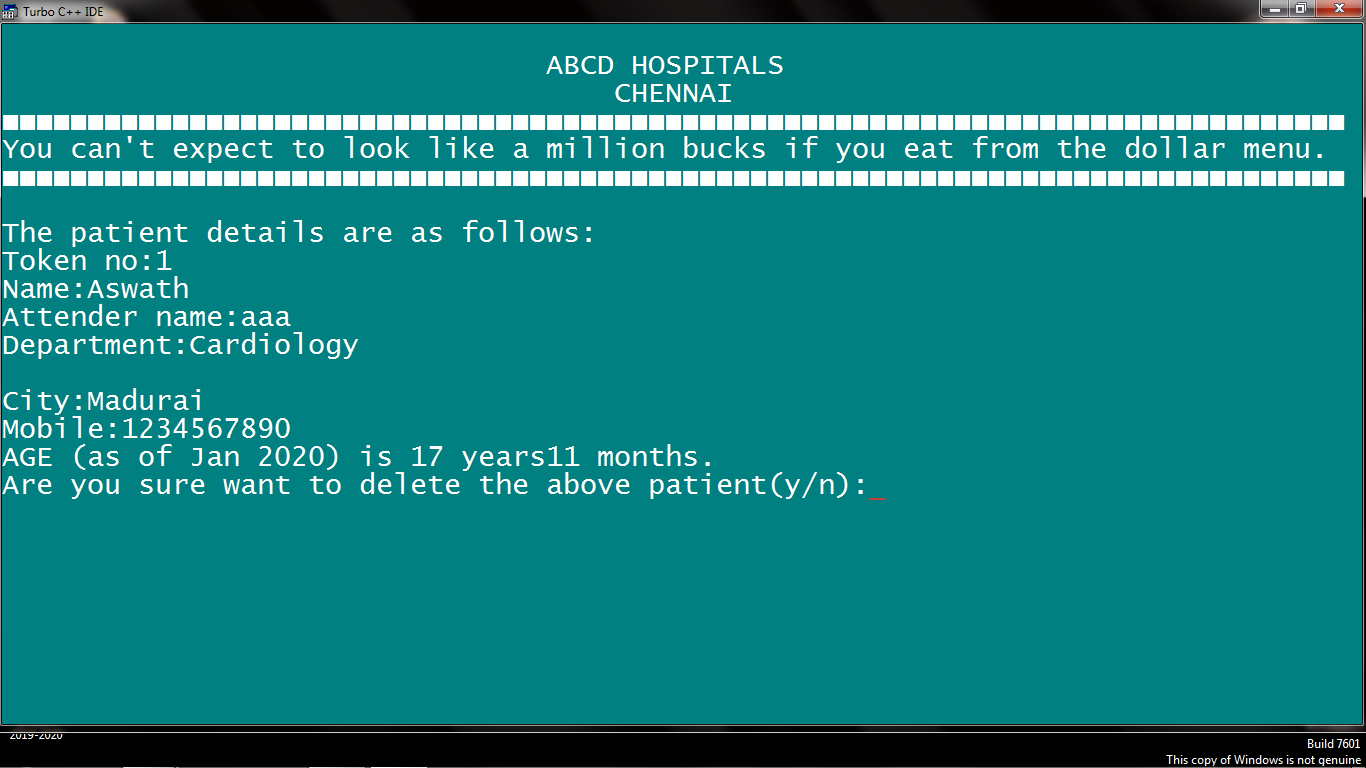
**No duplication of token number**

****

**Adding patient**

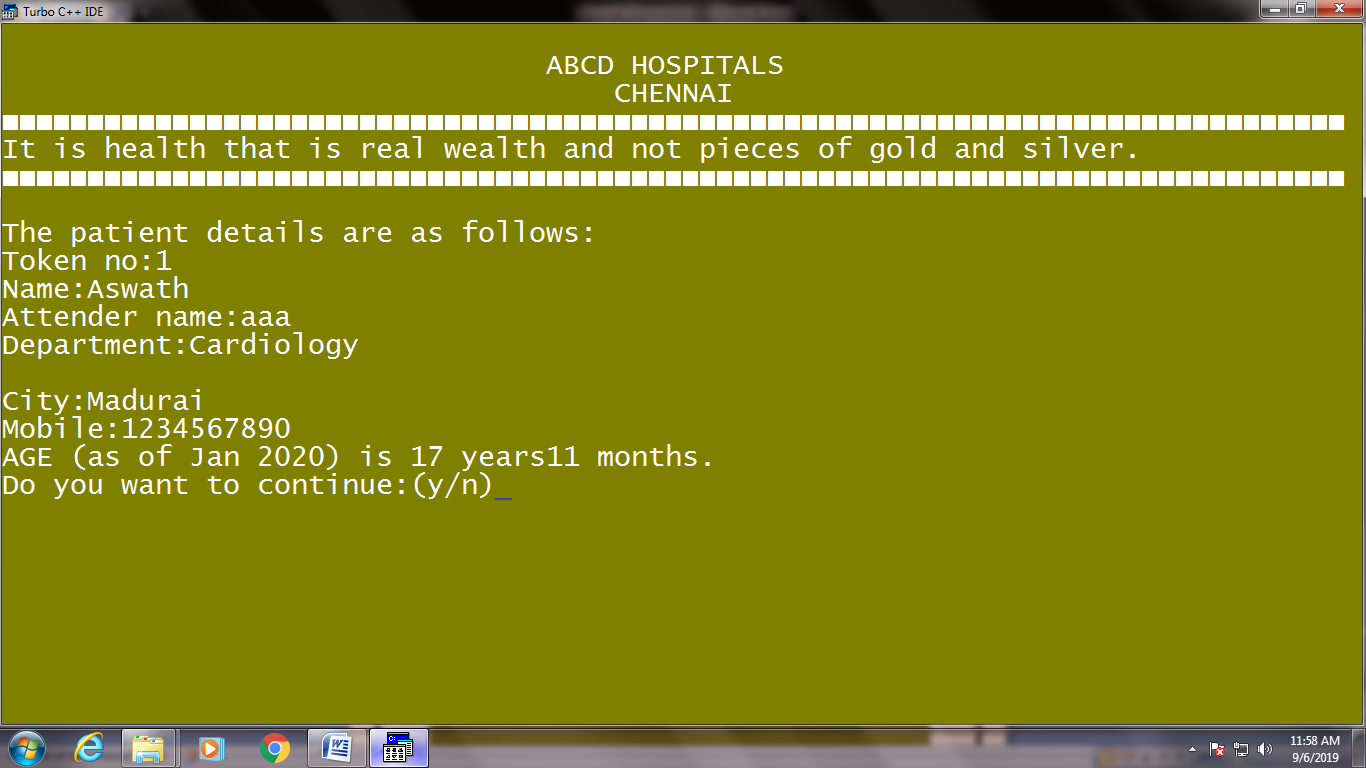
****

**Deleting patient**

****

**Modifying patient details**

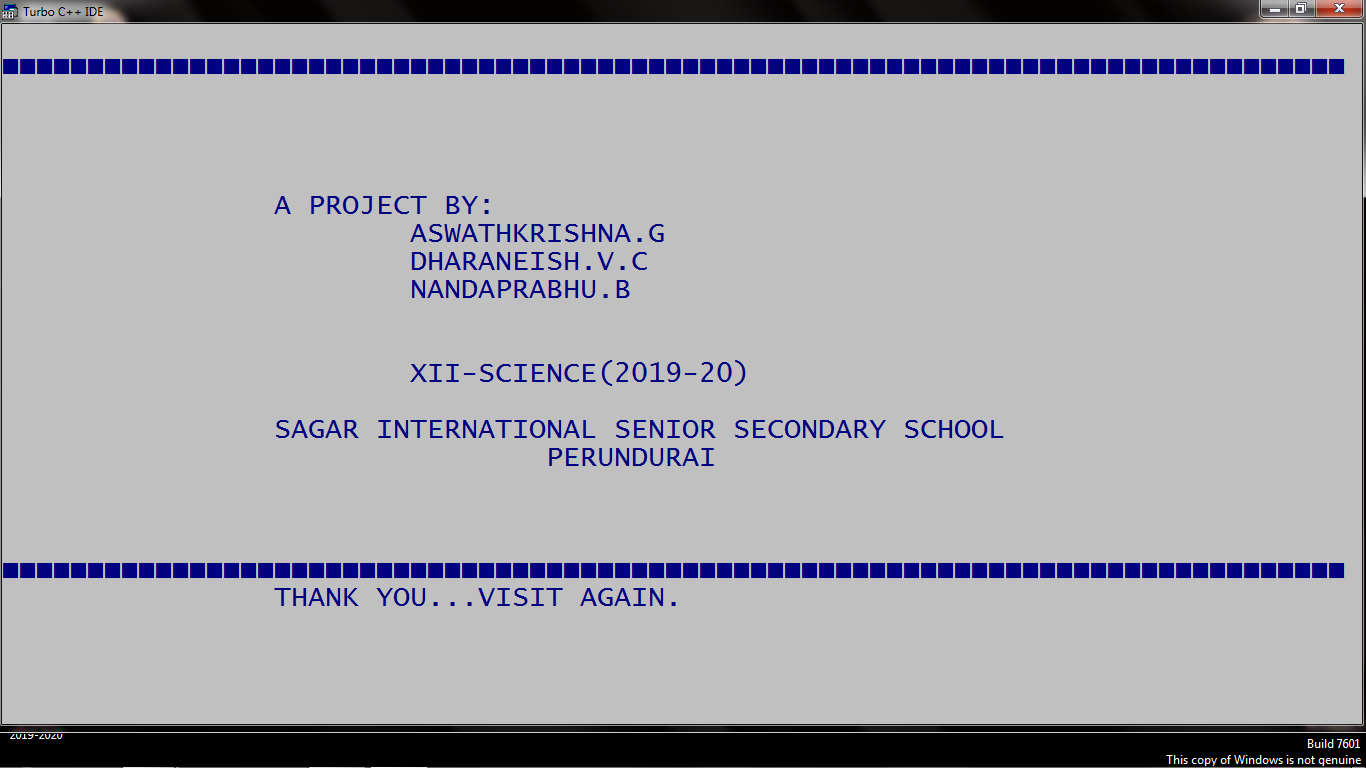
**View patient details**

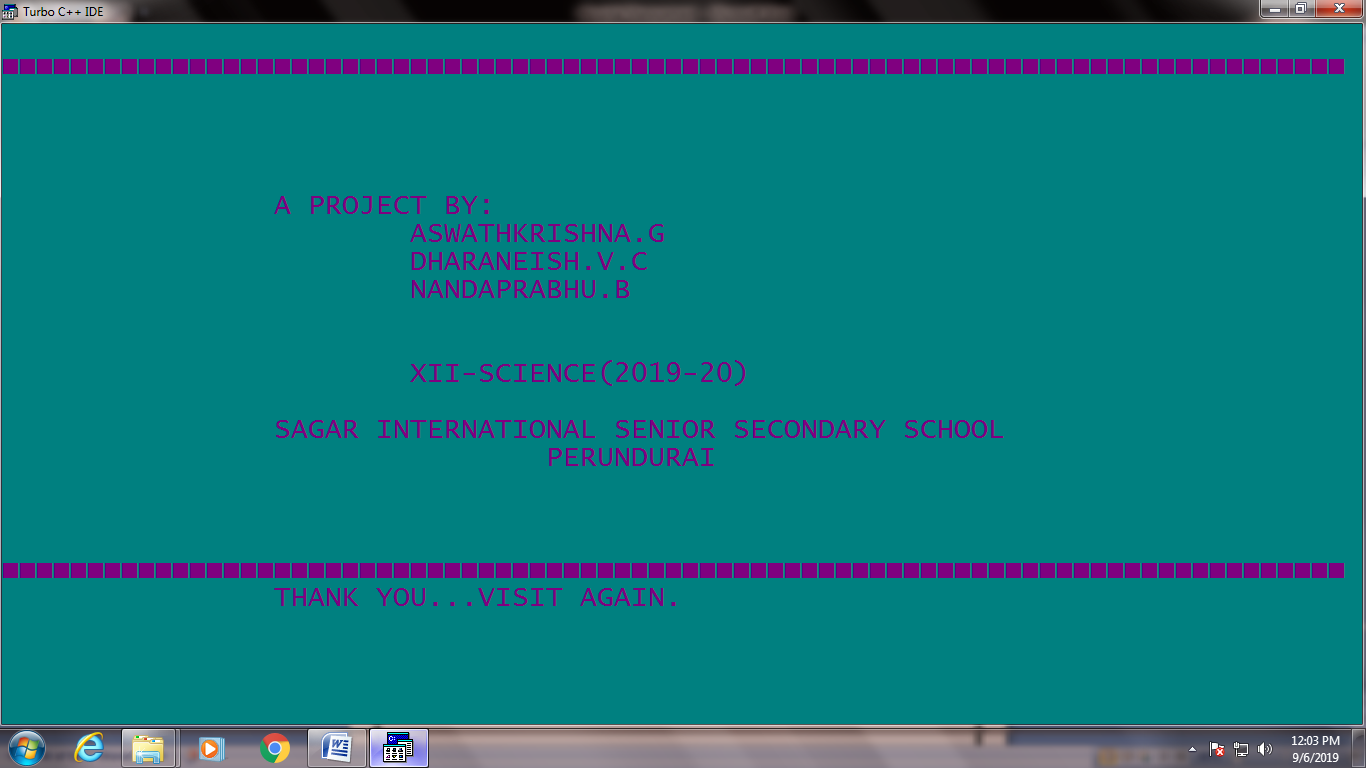
****

**More options for staff**

****

**Exit:**

****

****

**6.2. Discussion**

This software has many advantages:

* The system excludes the use of paper work by managing all the patient information electronically.
* Admin can keep updating the system by providing the new staff userIDs in system and delete userIDs thus Hospital management becomes easier and efficient.
* Many options such as Adding a patients details, modifying, deleting, viewing are included.
* Security is very high. So no unauthorized user will be able to access it.
* Thus, it saves human efforts and resources.

This software has many disadvantages:

* There is no human interaction if users have some enquiry.

**7. CONCLUSION AND FUTURE ENHANCEMENT**

**7.1. Conclusion**

After completing this project, sure the problems in the existing system would overcome. The “HOSPITAL MANAGEMENT SYSTEM” process made computerized to reduce human errors and to increase the efficiency. The main focus of this project is to lessen human efforts. The maintenance of the records is made efficient, as all the records are stored in the ACCESS database, through which data can be retrieved easily. The navigation control is provided in all the forms to navigate through the large amount of records. If the numbers of records are very large then user has to just type in the search string and user gets the results immediately. The editing is also made simpler. The user has to just type in the required field and press the update button to update the desired field.

The staff and Patients are given a particular unique id no.So that they can be accessed correctly and without errors. Our main aim of the project is to get the correct information about a particular Patient and access it when required. The problems, which existed in the earlier system, have been removed to a large extent. And it is expected that this project will go a long way in satisfying users’ requirement.

**7.2. Future Enhancement**

This application can be easily implemented under various situations. We can add new features as and when we require. Reusability is possible as and when require in this application. There inflexibility in all the modules.

SOFTWARE SCOPE

* **Extensibility**

This software is extendable in ways that its original developers may not expect. The following principles enhance extensibility like hide data structure, avoid traversing multiple links or methods, avoid case statements on object type and distinguish public and private operations.

* **Reusability**

Reusability is possible as and when require in this application. We can update it next version. Reusable software reduces design, coding and testing cost by amortizing effort over several designs. Reducing the amount of code also simplifies understanding, which increases the likelihood that the code is correct.•

* **Understandability**

A method is understandable if someone other than the creator of the method can understand the code (as well as the creator after a time lapse). We use the method, which small and coherent helps to accomplish this.•

* **Cost-effectiveness**

Its cost is under the budget and make within given time period. It is desirable to aim for a system with a minimum cost subject to the condition that it must satisfy the entire requirement. Scope of this document is to put down the requirements, clearly identifying the information needed by the user, the source of the information and outputs expected from the system

**8. BIBLIOGRAPHY**

* [www.programmingsimplified.com](http://www.programmingsimplified.com/)
* [www.cplusplus.com](http://www.cplusplus.com/)
* projects.icbse.com
* en.wikipedia.org
* [www.slideshare.net](http://www.slideshare.net/)
* [www.cppforschool.com](http://www.cppforschool.com/)
* Computer Science with C++, Class XII, Sumita Arora