Pokémon DataBase Management System (PDBMS)

Version 99.9 (2017-2018)

Computer Science Project

Developed By

Debmeet Banerjee

Delhi Public School, R.K.Puram

www.dpsrkp.net

Index

Certificate

This is to certify that **Pokémon DataBase Management System** Computer Science project is developed by **Debmeet Banerjee** under my supervision in the computer lab of Delhi Public School, R.K.Puram in the session 2017-2018. The work done by him is original.

Ms Hema Jain
Computer Science Teacher
Date: _____

Acknowledgement

I would like to express my sincere gratitude to my computer teacher Ms Hema Jain for her support and encouragement. She helped me completing the project by giving ideas, thoughts and made this project easy and accurate.

I wish to thank my parents for their support and interest during times the project seemed impossible and who inspired me to go my own way, without which I would be unable to complete my project. I would also like to thank my classmate Rahul Shahi, who helped me debug some parts of my program.

Reference

1. https://www.pokemon.com/us/
2. https://www.nintendo.com/games/pokemon-games

Introduction

What Are Pokémon?

Pokémon are creatures of all shapes and sizes who live in the wild or alongside humans. For the most part, Pokémon do not speak except to utter their names. Pokémon are raised and commanded by their owners (called “Trainers”). During their adventures, Pokémon grow and become more experienced and even, on occasion, evolve into stronger Pokémon. There are currently more than 700 creatures that inhabit the Pokémon universe.

Since there are a large number of Pokémon, one would need a lot of time to study each one of them. There needs to be a more systematic method to undertake the studies of Pokémon since it’s impossible otherwise to understand each one’s uniqueness.

My project therefore is a small attempt to make this task a bit easier. My project helps to organise the large number of Pokémon. Various basis can be used to classify Pokémon, for example, their name, Hit-points (in simple terms the health), their unique Pokémon Pokédex Number or the fact that they are issued or not.

Using my program, Pokémon records can be added, edited, sorted, displayed, issue them and search them based on various criterion.

Source Code

```
/*
Project Title  :Pokemon DataBase Management System
Version        :99.9
Developed By   :Debmeet Banerjee
School         :D.P.S. R.K. Puram
*/
#include<fstream.h>
#include  <conio.h>
#include  <stdio.h>
#include <string.h>
#include  <ctype.h>
#include<iomanip.h>
//Class Used
class poke
{
 int pno;
 long hp;
 char pname[55];
 char is;
public:
 void input()
 {
  cout<<"enter unique Pokemon ID          :";cin>>pno;
  cout<<"enter Pokemon Name               :";gets(pname);
  cout<<"Pokemon Hitpoints (MAX 5digits) :";cin>>hp;
```

```
  cout<<"Issued(i) OR Unissued(u)         :";cin>>is;
  is=tolower(is);
 }
 void output();
 int issue();
 int r_pno()
 {return pno;}
 char* r_pname()
 {return pname;}
 int r_hp()
 {return hp;}
 char r_is()
 {return is;}
};
void poke::output()
{
                  cout<<setw(15)<<pno<<setw(23)<<pname<<setw(15)<<hp<<"
"<<is<<endl;
}
int poke::issue()
{
 if(is=='u')
 {
  is='i';
  return 1;
 }
 else
  return 0;
```

```
}
//Structure Used
struct allusers
{
 char UserName[100];
 long int Password;
};
//Member Function Prototypes
void adminpasscreate();
void intro();
void pdataenter();
void pdataread();
void login_adm();
void administrator();
void user();
void reg(allusers []);
void login_withdraw_depo(allusers []);
void recdel();
void recedit();
void recnew();
void recsort();
void recwith();
void sbypno();
void sbypname();
void sbyhp();
void seeissue();
void withdraw();
```

```
//Main Function
void main()
{
 adminpasscreate();
 int reply;
 clrscr();
 intro();
 do
 {
  clrscr();
  cout<<"Hello There! Choose Something from Below!!\n\n\n";
  cout<<"(1) Administrator Mode\n\n";
  cout<<"(2) User Mode\n\n";
  cout<<"(3) Quit                                    :";cin>>reply;
  switch(reply)
  {
   case 1:login_adm();break;
   case 2:user();break;
  }
 }while(reply!=3);
 getch();
}
//Member Function Declarations
void adminpasscreate()
{
 fstream fil;
 fil.open("pass.txt",ios::out);
```

```
 char pass[10]="abcd";
 fil<<pass;
 fil.close();
}
void user()
{
 textcolor(WHITE);
 textbackground(BLUE);
 allusers A[100];
 int resp;
 getch();
 clrscr();
 cout<<"You Need to Register First"<<endl;
 reg(A);
 do
 {
  getch();
  clrscr();
  cout<<"(1)Login and Withdraw Pkmn \n";
  cout<<"(2)Search Pkmn By Hitpoints \n";
  cout<<"(3)Search Pkmn By Name       \n";
  cout<<"(4)Search Pkmn By Unique Pkmn ID \n";
  cout<<"(5)See Issued/Unissued Pkmn      \n";
  cout<<"(6)Quit to Main Menu                 : ";
  cin>>resp;
  switch(resp)
  {
```

```
   case 1:login_withdraw_depo(A);break;
   case 2:sbyhp();break;
   case 3:sbypname();break;
   case 4:sbypno();break;
   case 5:seeissue();
  }
 }while(resp!=6);
}
void reg(allusers A[100])
{
 cout<<"How many Users want to register?  :";
 int n;cin>>n;
 for(int i=0;i<n;i++)
 {
  cout<<"Enter your Email Id of user "<<i+1<<" :";
  gets(A[i].UserName);
  cout<<"Choose Password(Only number, NO alphabet/characters) :";
  cin>>A[i].Password;
  cout<<"User Registered"<<endl;
 }
}
void login_withdraw_depo(allusers A[100])
{
 allusers B; int find=-1,logg=0,resp=1;
 do
 {
  cout<<"Login......."<<endl;
```

```
 cout<<"Email Id  :";gets(B.UserName);
 cout<<"PassWord  :";cin>>B.Password;
 for(int i=0;i<100&&(find==-1);i++)
 {
  if(strcmpi(B.UserName,A[i].UserName)==0)
   find=i;
 }
 if(B.Password==A[find].Password)
 {
  logg=1;resp=0;
  cout<<"Login Success"<<endl;
  withdraw();
 }
 else
 {
  cout<<"Login Failed"<<endl;
  cout<<"You need to Login To Withdraw Pokemon"<<endl;
  cout<<"Retry Login?   \n";
  cout<<"(1)Yes,        \n";
  cout<<"(2)No,Quit to Main Menu             :";cin>>resp;
  switch(resp)
  {
   case 1:resp=1;break;
   case 2:logg=0;
  }
 }
}while((logg==0)&&(resp==1));
```

```
}
void withdraw()
{
 int r;
 do
 {
  getch();
  clrscr();
  cout<<"What to do? \n";
  cout<<"(1)See Pokemon \n";
  cout<<"(2)Issue Pokemon  \n";
  cout<<"(3)Logout-Return to Menu    :";cin>>r;
  switch(r)
  {
   case 1:pdataread();break;
   case 2:recwith();break;
   case 3:cout<<"Logged Out"<<endl;
  }
 }while(r!=3);
}
void login_adm()
{
 char pass[10],res[10],ch;
 int i=0,j=0;
 fstream fil("Pass.txt",ios::in);
 fil>>pass;
 do
```

```
 {
  ch=0;i++;j=0;
  cout<<"Enter administrator password (Attempt "<<i<<"): ";
  for(j;j<10&&ch!=13;j++)
  {
   ch=getch();
   cout<<"*";
   res[j]=ch;
  }
  j--;
  res[j]='\0';
  cout<<"\b \n";
 }while(strcmp(res,pass)&&i<3);
 if(strcmp(res,pass)==0)
  administrator();
 else
  cout<<"Failed 3 times. Try Again\n";
  getch();
}
void administrator()
{
 textcolor(YELLOW);
 textbackground(BLACK);
 int choice;
 clrscr();
 cout<<"Hello Administrator "<<endl;
 do
```

```
 {
  getch();
  clrscr();
  cout<<"You want to?\n\n\n";
  cout<<"(1)Add     Records\n";
  cout<<"(2)Read    Records\n";
  cout<<"(3)Delete  Record\n";
  cout<<"(4)Edit    Record\n";
  cout<<"(5)Insert  Record\n";
  cout<<"(6)Sort    Records\n";
  cout<<"(7)Quit to Main Menu                  :";
  cin>>choice;
  switch(choice)
  {
   case 1:clrscr();
   pdataenter();
   break;
   case 2:clrscr();
                                                cout<<setw(15)<<"Pkmn
No."<<setw(23)<<"Name"<<setw(15)<<"Hitpoints"<<setw(20)<<"Issue
Status\n";
   cout<<setw(80)<<"(i= issued | u= unissued)"<<endl;
   pdataread();
   break;
   case 3:clrscr();
   recdel();
   break;
   case 4:recedit();
```

```
   break;
   case 5:recnew();
   break;
   case 6:recsort();
   cout<<"All records Sorted"<<endl;
   break;
  }
 }while(choice!=7);
}
void recdel()
{
 fstream fil1,fil2;
 fil1.open("poke.dat",ios::in| ios::binary);
 fil2.open("temp.dat",ios::out|ios::binary);
 int pnum,del=0;
 cout<<"Enter Unique Pokemon Id to Delete Record :";
 cin>>pnum;
 poke p;
 while(fil1.read((char*)&p,sizeof(poke)))
 {
  if(p.r_pno()!=pnum)
   fil2.write((char*)&p,sizeof(p));
  else
   del++;
 }
 fil1.close();fil2.close();
 if(del)
```

```
 {
  cout<<"Pokemon Record successfully deleted"<<endl;
  getch();
  remove("poke.dat");
  rename("temp.dat","poke.dat");
 }
 else
 {
  cout<<"Pokemon with that Pokemon ID was Not Found!"<<endl;
  remove("temp.dat");
 }
}
void recnew()
{
 fstream fil1,fil2;
 fil1.open("poke.dat",ios::binary|ios::in );
 fil2.open("temp.dat",ios::binary|ios::out);
 poke p,pnew;
 int insert=0;
 cout<<"Enter Contents Of new Pokemon"<<endl;
 pnew.input();
 while(fil1.read((char*)&p,sizeof(p)))
 {
  if(p.r_pno()>pnew.r_pno()&&!insert)
  {
   fil2.write((char*)&pnew,sizeof(p));
   insert++;
```

```
  }
  fil2.write((char*)&p,sizeof(p));
 }
 if(!insert)
  fil2.write((char*)&pnew,sizeof(p));
 fil1.close();fil2.close();
 remove("poke.dat");
 rename("temp.dat","poke.dat");
}
void recsort()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in|ios::out);
 poke p1,p2;
 fil.seekg(0,ios::end);
 int n=fil.tellg()/sizeof(poke);
 for(int i=0;i<n-1;i++)
  for(int j=0;j<n-1-i;j++)
  {
   fil.seekg(j*sizeof(poke));
   fil.read((char*)&p1,sizeof(p1));
   fil.read((char*)&p2,sizeof(p2));
   if(p1.r_pno()>p2.r_pno())
   {
    fil.seekg(j*sizeof(p1));
    fil.write((char*)&p2,sizeof(p2));
    fil.write((char*)&p1,sizeof(p1));
```

```
   }
  }
 fil.close();
}
void recedit()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in|ios::out);
 poke p;
 int pnum,found=0;
 cout<<"Enter The Unique Pokemon ID to Edit Corresponding Record :";
 cin>>pnum;
 while(!found&&fil.read((char*)&p,sizeof(p)))
 {
  if(pnum==p.r_pno())
  {
   cout<<"Now re-enter The New Details!"<<endl;
   p.input();
   found++;
   fil.seekg(fil.tellg()-sizeof(p));
   fil.write((char*)&p,sizeof(p));
  }
 }
 if(found)
  cout<<"Record Edited"<<endl;
 else
  cout<<"That Pokemon ID is Not Registered"<<endl;
```

```
 fil.close();
}
void recwith()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in|ios::out);
 poke p;
 int r,pnum,issued=0,found=0;
 do
 {
  cout<<"Enter The Unique Pokemon ID to Withdraw :";
  cin>>pnum;
  while(!found&&fil.read((char*)&p,sizeof(p)))
  {
   if(pnum==p.r_pno())
   {
    found++;
    issued=p.issue();
    fil.seekp(fil.tellg()-sizeof(p));
    fil.write((char*)&p,sizeof(p));
   }
  }
  if(issued)
   cout<<"Issued\n";
  else
   cout<<"Not available\n";
  cout<<"Continue withdrawing? 1.Yes 2.No :";
```

```
  cin>>r;
 }while(r==1);
 fil.close();
}
void intro()
{
 textcolor(YELLOW);
 textbackground(GREEN);
 clrscr();
 cout<<"\n\n\n\n\n\n\n";
 cout<<"#########                              #                          ##
#"<<endl;
 cout<<"#        #                          #                          #  #
#"<<endl;
 cout<<"#        #            #      #      #                          #
#       #"<<endl;
 cout<<"######### #######   #    #      #######  #        #  #######  #
#     #"<<endl;
 cout<<"#          #      #  # #        #         # #   # #  #     #  #
#   #"<<endl;
 cout<<"#    #     #  #   #     #####    #   #   #  #     #  #        #
#"<<endl;
 cout<<"#          #      #  #      #   #         #       #  #     #  #
##"<<endl;
 cout<<"#          #######  #        # #######  #        #  #######  #
#"<<endl;
 cout<<"\n\n\n";
 cout<<"                     DATA-BASE MANAGEMENT SYSTEM "<<endl;
 cout<<"\n\n\n\n";
  cout<<"NOTE: Pokemon and Pokemon character names are trademarks of
NINTENDO \n\n"<<endl;
```

```
 getch();
 clrscr();
}
void pdataenter()
{
 char rep;
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::app);
 poke p;
 do
 {
  p.input();
  fil.write((char*)&p,sizeof(p));
  cout<<"Record Added, More?(Y/N)"<<endl;
  cin>>rep;
 }while(rep=='y'||rep=='Y');
 fil.close();
}
void pdataread()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in);
 poke p;
 while(fil.read((char*)&p,sizeof(p)))
  p.output();
 fil.close();
}
```

```
void sbypno()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in);
 poke p;
 int pnumber,found=0;
 cout<<"Enter the Unique Pokemon ID "<<endl;
 cin>>pnumber;
 while(found==0&&fil.read((char*)&p,sizeof(p)))
  if(pnumber==p.r_pno())
    {
     p.output();
     found++;
    }
 if(!found)
  cout<<"A Pokemon With That ID is not Registered yet!!\n"<<endl;
 fil.close();
}
void sbypname()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in);
 poke p;
 char name[100];
 cout<<"Enter the Pokemon Name "<<endl;
 gets(name);
 int found=0;
```

```
 while(found==0&&fil.read((char*)&p,sizeof(p)))
  if(strcmpi(name,p.r_pname())==0)
    {
     p.output();
     found++;
    }
 if(!found)
  cout<<"A Pokemon With That Name is not Registered yet!!\n"<<endl;
 fil.close();
}
void sbyhp()
{
 fstream fil;
 fil.open("poke.dat",ios::binary|ios::in);
 poke p;
 int hit,found=0;
 cout<<"Enter the Pokemon Hitpoints "<<endl;
 cin>>hit;
 while(found==0&&fil.read((char*)&p,sizeof(p)))
  if(hit==p.r_hp())
    {
     p.output();
     found++;
    }
 if(!found)
       cout<<"Could Not find Pokemon with that amount of
Hitpoints!!\n"<<endl;
 fil.close();
```

```
}
void seeissue()
{
 char issue;
 cout<<"See issued Pokemon(i) or Unissued(u) ";
 cin>>issue;
 issue=tolower(issue);
 if(issue=='i')
 {
  fstream fil;
  fil.open("poke.dat",ios::binary|ios::in);
  poke p;
  cout<<"\nIssued Pokemon are  "<<endl;
  while(fil.read((char*)&p,sizeof(p)))
   if(issue==p.r_is())
    {
     p.output();
    }
    fil.close();
 }
 else if(issue=='u')
 {
  fstream fil;
  fil.open("poke.dat",ios::binary|ios::in);
  poke p;
  cout<<"\n Unissued Pokemon are  "<<endl;
  while(fil.read((char*)&p,sizeof(p)))
```

```
   if(issue==p.r_is())
    {
     p.output();
    }
    fil.close();
 }
}
```

Output Screen

Hardware & Software Requirement

Hardware Requirement

Intel 2.66 GHz LGA 775 Core 2 Duo OR
AMD Sempron 145 2.8 GHz Single Core Processor
Or Higher
With at least 1 GB RAM at 400 MHz Memory Clock
15 MB free space on Hard Disk
Colour Monitor/LCD with min 40fps.

Operating System & Compiler

Microsoft Windows 7, 8, 8.1, 10 or higher.
Turbo C++ 3.1 Compiler