|  |  |  |  |
| --- | --- | --- | --- |
| animatedLOGO | **Assignment No. 02 Semester: Spring 2021**  **CS304- Object Oriented Programming** | | **Total Marks: 20**  **Due Date:** |
| #include<iostream>  #include<math.h>  #include<cstdlib>  #include<string>  using namespace std;  //////////////////////////////////////////////////  /////////////// Person Class  ////////////////////  //////////////////////////////////////////////////  class Person  {    private :      //data members      string FirstName;      string LastName;      int PersonalID ;      double Salary ;      double WorkingHours ;      double CostPerHour ;      public :      //function members to access data members      void setFirstName();      void setFamilyName();      void set\_PersonalID();      void setWorkingHour();      void setCostPerHour();      void setFieldName();      void getFieldName();      void confirmFieldName();  };    //////////////////////////////////////////////////  /////// Human Resource Management Class  /////////  //////////////////////////////////////////////////  class HRM  {    private :  //variables      Person e[10] ;      Person temp[10];    public : //members      void AddPerson();      void DeletePerson();      void UpdatePerson();  };  int n,i,check,x=8248001,y[100],z[100];  bool flag=0;  string h[100];  double sal[100];  /////////////////////////////////////////////////////////////////////  // Add an employee  void ::HRM::AddPerson() {      int x;    char repeate;      do    {      n++;      e[i].setFieldName();      i++;        cout<<"\nThe employee with the following information has been added to the system:"<<endl;      cout<<"\nFirst Name       Last Name       Personal ID         Salary per year (Rupees)";      cout<<"\n--------------   --------------  ------------       -------------------------"<<endl;        for(int i=0; i<n; i++){        e[i].getFieldName();      }        cout<<"Do you wont to add another employee(y/n): "<<endl;      cin>>repeate;    }    while((repeate=='y'||repeate=='Y'));    }        /////////////////////////////////////////////////////////////////////  //  Delete an Employee    void HRM::DeletePerson() {    int eID;    bool flag;    char repeate1,repeate2;    flag =false;    start:    cout << "ID of Employee to Remove: ";      while(!(cin>>eID))    {      cout << "Please  enter a number!  Try again: ";      cin.clear ();      cin.ignore (1000, '\n');  // Skip to next newline or 1000 char, whichever comes first.    }    for ( i = 0; i < n; ++i) {      if (y[i]==eID) {        e[i]=e[i+1];        flag=true;        e[i].confirmFieldName();        cin>>repeate2;        if(repeate2=='Y'||repeate2=='y'){          int a;          a=n;          cout<<"\nThe employee with the following information has been deleted from the system:"<<endl;          cout<<"\nFirst Name       Last Name       Personal ID         Salary per year (Rupees)";          cout<<"\n--------------   --------------  ------------       -------------------------"<<endl;          for(int i=0; i<a; i++){            e[i].getFieldName();            a--;            n=a;            n++;          }          ///      goto start1;        }        cout <<endl;        //Delete function      }    }    if (flag==false) {      cout<<"Sorry, there is not any employee with requested personal number."      <<" Do you want to repeat delete by entering the new personal number (y/n)? ";      cin>>repeate1;      if(repeate1=='Y'||repeate1=='y'){        goto start;        cout<<endl<<endl;      }    }    }  /////////////////////////////////////////////////////////////////////  //    Update an Employee  /////////////////////////////////////////////////////////////////////  void HRM::UpdatePerson(){      int eID;    char repeate1,repeate2;      start:    cout << "ID of employee to modify data: ";      while(!(cin>>eID))    {      cout << "Please  enter a number!  Try again: ";      cin.clear ();      cin.ignore (1000, '\n');  // Skip to next newline or 1000 chars, whichever comes first.    }    int flag1=0;    for (int i = 0; i < n; ++i) {      if (y[i]!=eID) {        flag1++;      }    }      cout <<endl;    start1:    for (int i = 0; i < n; ++i) {      if (y[i]==eID) {        cout<<"Matching: "<< y[i];        flag=true;        int choice = 0;        char repeate;          do {          cout << endl << endl;          cout << "Please enter the related number of field which you would like to update" << endl;          cout <<"1. First name" << endl;          cout << "2. Family name" << endl;          cout << "3. Working hours per week" << endl;          cout << "4. Payment for one hour" << endl;          cout << std::endl;          cin >> choice;          if (choice == 1) {            cout << " First name: ";            e[i].setFirstName();          }          else if (choice == 2) {            cout << " Family name: ";            e[i].setFamilyName();          }          else if (choice == 3) {            cout << " Working hours per week: ";            e[i].setWorkingHour();          }          else if (choice == 4) {            cout << " Payment for one hour: ";            e[i].setCostPerHour();          }          cout<<"Do you want to update another Field (Y/N)? ";          cin>>repeate;        } while (repeate=='y'||repeate=='Y');      }    }    int a;    a=n;    cout<<"\nThe employee with the following information has been updated to the system:"<<endl;    cout<<"\nFirst Name       Last Name       Personal ID         Salary per year (Rupees)";    cout<<"\n--------------   --------------  ------------       -------------------------"<<endl;    for(int i=0; i<a; i++){      e[i].getFieldName();      }  }  /////////////////////////////////////////////////////////////////////  void ::Person::setFirstName()  {    cin>>FirstName ;  }  /////////////////////////////////////////////////////////////////////  void::Person::setFamilyName()  {    cin>>LastName;  }  /////////////////////////////////////////////////////////////////////  void::Person::setWorkingHour()  {    cin>>WorkingHours;    Salary=WorkingHours\*CostPerHour\*52;  }  /////////////////////////////////////////////////////////////////////  void::Person::setCostPerHour()  {    cin>>CostPerHour;    Salary=WorkingHours\*CostPerHour\*52;  }    /////////////////////////////////////////////////////////////////////  void::Person ::getFieldName(){    cout<<FirstName<<"\t\t"<<LastName<<"\t\t\t"<<PersonalID<<"\t\t\t"<<Salary<<"\t"<<endl;  }  /////////////////////////////////////////////////////////////////////  void::Person ::confirmFieldName(){    cout<<"Do you really want to delete Employee: "<<FirstName<<" "<< LastName<< " (y/n)? ";  }      int k=0;  /////////////////////////////////////////////////////////////////////  void::Person ::setFieldName(){    PersonalID=x;    x++;    cout<<n;    cout<<"\nFirst Name: ";      while(!(cin>>FirstName))    {      cout << "Please  enter a string!  Try again: ";      cin.clear ();      cin.ignore (1000, '\n');  // Skip to next newline or 1000 chars, whichever comes first.    }    cout<<"Last Name: ";      while(!(cin>>LastName))    {      cout << "Please  enter a string!  Try again: ";      cin.clear ();      cin.ignore (1000, '\n');  // Skip to next newline or 1000 chars, whichever comes first.    }      cout<<"How many hours a week is his/her work? ";    ;    while(!(cin>>WorkingHours))    {      cout << "Please  enter a number!  Try again: ";      cin.clear ();      cin.ignore (1000, '\n');  // Skip to next newline or 1000 chars, whichever comes first.    }    cout<<"How much per hour is his/her salary? ";      while(!(cin>>CostPerHour))    {      cout << "Please  enter a number!  Try again: ";      cin.clear ();      cin.ignore (1000, '\n');  // Skip to next newline or 1000 chars, whichever comes first.    }      Salary=WorkingHours\*CostPerHour\*52;    z[i]= Salary;    sal[i]=Salary;    h[i]=LastName;    y[k]=PersonalID;    k++;  }    /////////////////////////////////////////////////////////////////////  int main()  {    HRM xyz ;//HRM Class instance    int c;    char ch;    cout<<        "Welcome to the  HRM written by TAMKEEN SAJJAD MC2004000003 "<<endl;    cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;    cout<<endl<<endl<<endl;    do    {        cout<<"Welcome to Human Resource Management (HRM) Software of Company XYZ.";      cout<<"To perform specific task please choose one of the following commands!"<<endl<<endl<<endl;      cout<<"    1. Add new Employee"<<endl;      cout<<"    2. Delete Employee information"<<endl;      cout<<"    3. Update Employee information"<<endl;      cout<<"    4. Quit"<<endl<<endl;          while(!(cin>>c))  //Reciving vaiables from input : is it no/character ?      {        cout << "Please  enter a number!  Try again: ";        cin.clear ();        cin.ignore (1000, '\n');  // Skip to next newline or 1000 chars,                                    // whichever comes first.      }      switch(c)      {        case 1:          cout<<"\nEnter the Information of the new Employee to be added: "<<endl;          xyz.AddPerson();          break;        case 2:          cout<<"\nEnter an Item to delete: ";          xyz.DeletePerson();          break;        case 3:          cout<<"\nEnter an Item to update: ";          xyz.UpdatePerson();          break;        default :          cout<<"\nInvalid option try again: ";      }      cout<<"\nDo you Want to Continue (y/n)? ";      cin>>ch;    }    while(ch=='y'||ch=='Y');    system("pause");    return 0;  } | | | |
| **Assignment** | |  | |