//node.h

#ifndef NOD\_H

#define NOD\_H

#include"employee.h"

class Node

{

CEmployee \*e;

Node \*next;

public:

Node();

void setnext(Node \*);

Node\* getnext()const;

void setE(CEmployee \*);

CEmployee\* getE()const;

};

#endif

//node.cpp

#include"node.h"

#include"programer.h"

#include"salesperson.h"

Node::Node()

{

/\*int empid,hrs;

char name[20];

double comm;

cout << "Enter empid ;";

cin >> empid;

cout << "Enter name :";

cin >> name;

cout << "Enter hrs ;";

cin >> hrs;

cout << "Enter comm ;";

cin >> comm;\*/

int choice;

cout << "\n1.programmer \n2.Salesmaneger " << endl;

cout << "Enter the choice :";

cin >> choice;

switch (choice)

{

case 1:this->e = new CProgrammer();

break;

case 2:this->e = new CSalesmanager();

break;

default:exit(0);

}

this->next = NULL;

}

void Node::setnext(Node \*next)

{

this->next = next;

}

Node\* Node::getnext()const

{

return this->next;

}

void Node::setE(CEmployee \*e)

{

this->e = e;

}

CEmployee\* Node::getE()const

{

return this->e;

}

//linklist.h

#ifndef LNL\_H

#define LNL\_H

#include"node.h"

#include<iostream>

using namespace std;

class CLinkedlist

{

Node \*head;

public:

CLinkedlist();

bool empty();

bool addFirst();

void Display();

void Displaypro();

void Displaysales();

bool insert(int);

//bool AdditionNode(int);

};

#endif

//linklist.cpp

#include"node.h"

#include"linkedlist.h"

#include"programer.h"

#include"salesperson.h"

CLinkedlist::CLinkedlist()

{

this->head = NULL;

}

bool CLinkedlist::empty()

{

if (this->head == NULL)

return 0;

else

return 1;

}

bool CLinkedlist::addFirst()

{

Node \*newNode = new Node ();

if (newNode == NULL)

{

return false;

}

if (this->head==NULL)

{

this->head = newNode;

return true;

}

else

{

newNode->setnext(this->head);

this->head = newNode;

return true;

}

}

bool CLinkedlist::insert(int position) {

Node \*newNode = new Node();

if (newNode == NULL)

return false;

if (head == NULL) {

head = newNode;

return true;

}

else

{

Node \*trav = this->head;

for (int i = 1; i < position - 1; i++)

trav = trav->getnext();

newNode->setnext(trav->getnext());

trav->setnext(newNode);

}

return true;

}

//bool CLinkedlist::AdditionNode(int position)

//{

// //Node \* newnode = new Node();

//

// Node \* trav = head;

// for(int i=1;i<position-1;i++)

// {

// trav = trav->getnext();

// }

// Node \*ptr = trav->getnext();

// Node \*newNode = new Node(trav->getdata() + ptr->getdata() );

// newNode->setnext(this->head);

// this->head->setnext(newNode);

// return true;

//

//}

void CLinkedlist::Display()

{

Node \*trav = this->head;

while (trav != NULL)

{

trav->getE()->Display();//Employee

trav = trav->getnext();

}

}

void CLinkedlist::Displaypro()

{

Node \*trav = this->head;

while (trav != NULL)

{

if (typeid(\*trav->getE()) == typeid(CProgrammer))

{

trav->getE()->Display();//Employee

}

trav = trav->getnext();

}

}

void CLinkedlist::Displaysales()

{

Node \*trav = this->head;

while (trav != NULL)

{

if (typeid(\*trav->getE()) == typeid(CSalesmanager))

{

trav->getE()->Display();//Employee

}

trav = trav->getnext();

}

}

//employee.h

#ifndef EMP\_H

#define EMP\_H

#include<iostream>

using namespace std;

class CEmployee

{

int empid;

char name[20];

public:

CEmployee();

CEmployee(int,const char\*);

void setid(int);

int getid()const;

void setname(const char \*);

const char\* getname()const;

void Accept();

void Display();

};

#endif

//employee.cpp

#include"employee.h"

CEmployee::CEmployee()

{

this->empid = 0;

strcpy(this->name, " ");

}

CEmployee::CEmployee(int id, const char \*name)

{

this->empid = id;

strcpy(this->name, name);

}

void CEmployee::setid(int id)

{

this->empid = id;

}

int CEmployee::getid()const

{

return this->empid;

}

void CEmployee::setname(const char \*name)

{

strcpy(this->name, name);

}

const char\* CEmployee::getname()const

{

return this->name;

}

void CEmployee::Accept()

{

cout << "Emp ID :";

cin >> this->empid;

cout << "Name :";

cin >> this->name;

}

void CEmployee::Display()

{

cout << "Emp ID :" << this->empid << endl;

cout << "Name :" << this->name << endl;

}

//salesperson.h

#ifndef SAL\_H

#define SAL\_H

#include"employee.h"

class CSalesmanager:public CEmployee

{

int target;

double comm;

public:

CSalesmanager();

CSalesmanager(int, const char\*, int, double);

void settarget(int);

int gettarget()const;

void setcomm(double);

double getcomm()const;

void Display();

};

#endif

// salesperson.cpp

#include"salesperson.h"

CSalesmanager::CSalesmanager()

{

CEmployee::Accept();

cout << "Target : ";

cin>>this->target;

cout << "Comm : ";

cin>>this->comm ;

}

CSalesmanager::CSalesmanager(int id,const char \*name,int traget,double comm):CEmployee(id,name)

{

this->target = target;

this->comm = comm;

}

void CSalesmanager::settarget(int target)

{

this->target = target;

}

int CSalesmanager::gettarget()const

{

return this->target;

}

void CSalesmanager::setcomm(double comm)

{

this->comm = comm;

}

double CSalesmanager::getcomm()const

{

return this->comm;

}

void CSalesmanager::Display()

{

CEmployee::Display();

cout << "Target : " << this->target << endl;

cout << "Comm : " << this->comm << endl;

}

//programmer.h

#ifndef PRG\_H

#define PRG\_H

#include"employee.h"

class CProgrammer: public CEmployee

{

int hrs;

double comm;

public:

CProgrammer();

CProgrammer(int ,const char\*,int, double);

void sethrs(int);

int gethrs()const;

void setcomm(double);

double getcomm()const;

void Display();

};

#endif

//programmer.cpp

#include"programer.h"

CProgrammer::CProgrammer()

{

CEmployee::Accept();

cout << "Hours :";

cin >> this->hrs;

cout << "Commision :";

cin >> this->comm;

}

CProgrammer::CProgrammer(int empid,const char \*name,int hrs, double comm):CEmployee(empid,name)

{

this->hrs = hrs;

this->comm = comm;

}

void CProgrammer::sethrs(int hrs)

{

this->hrs = hrs;

}

int CProgrammer::gethrs()const

{

return this->hrs;

}

void CProgrammer::setcomm(double comm)

{

this->comm = comm;

}

double CProgrammer::getcomm()const

{

return this->comm;

}

void CProgrammer::Display()

{

CEmployee::Display();

cout << "Hours :" << this->hrs << endl;

cout << "Commision :" << this->comm << endl;

}

//main.cpp

#include"linkedlist.h"

//#include"employee.h"

//#include"programer.h"

#include"node.h"

#include<conio.h>

int menu\_list()

{

int choice;

cout << "0.Exit" << endl;

cout << "1.Add First" << endl;

cout << "2.Dislpay" << endl;

cout << "3.Insert" << endl;

//cout << "4.AdditionNode" << endl;

cout << "Enter the choice :";

cin >> choice;

return choice;

}

int main()

{

int choice,position;

//CEmployee emp;

//CProgrammer pro;

CLinkedlist list;

while ((choice = menu\_list()) != 0)

{

switch (choice)

{

case 1:

list.addFirst();

break;

case 2:

list.Display();

break;

case 3:

cout << "Enter the position :";

cin >> position;

list.insert(position);

break;

case 4:

list.Displaypro();

break;

case 5:

list.Displaysales();

break;

}

}

\_getch();

return 0;

}