**Program to illustrate the access control in public derivation of a class**

#include<iostream>

using namespace std;

const int len=25;

class Employee

{

private:

char name[len];

long enumb;

public:

void getdata()

{

cout<<"Enter name of the employee\t";

cin>>name;

cout<<"Enter employee number\t";

cin>>enumb;

}

void putdata()

{

cout<<"Name of the Employee:"<<name<<"\t";

cout<<"Employee Number:"<<enumb<<"\t";

cout<<"Basic Salary:"<<basic;

}

protected:

float basic;

void getbasic()

{

cout<<"Enter Basic";

cin>>basic;

}

};

class Manager : public Employee

{

private:

char title[len];

public:

void getdata()

{

Employee :: getdata();

getbasic();

cout<<"Enter title";

cin>>title;

}

void putdata()

{

Employee :: putdata();

cout<<"Title"<<title<<"\n";

}

} ;

int main()

{

Manager m1,m2;

cout<<"Enter the details of Manager1\n";

m1.getdata();

cout<<"Enter the details of manager2\n";

m2.getdata();

cout<<"\t\tDetails of Manager1\n";

m1.putdata();

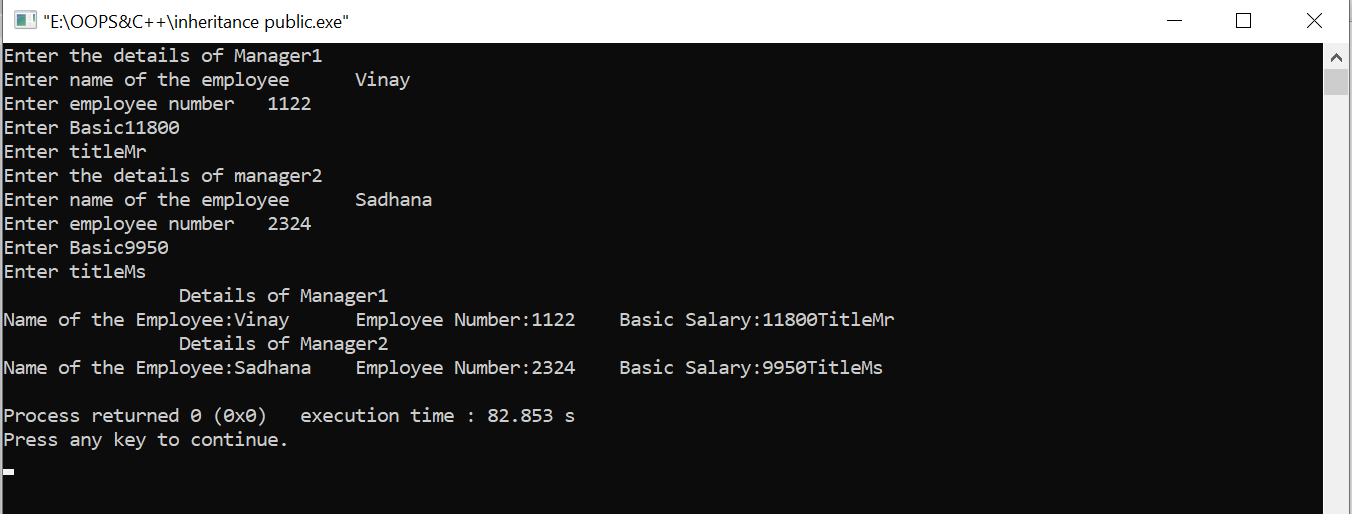
cout<<"\t\tDetails of Manager2\n";

m2.putdata();

return 0;

}

Output:



**Program to illustrate access control of inherited members in the privately derived class**

#include<iostream>

using namespace std;

const int len=25;

class Employee

{

private:

char name[len];

long enumb;

public:

void getdata()

{

cout<<"Enter name of the employee\t";

cin>>name;

cout<<"Enter employee number\t";

cin>>enumb;

}

void putdata()

{

cout<<"Name of the Employee:"<<name<<"\t";

cout<<"Employee Number:"<<enumb<<"\t";

cout<<"Basic Salary: "<<basic;

}

protected:

float basic;

void getbasic()

{

cout<<"Enter Basic ";

cin>>basic;

}

};

class Manager : private Employee

{

private:

char title[len];

public:

void getdata()

{

Employee :: getdata();

getbasic();

cout<<"Enter title ";

cin>>title;

}

void putdata()

{

Employee :: putdata();

cout<<"Title "<<title<<"\n";

}

} ;

int main()

{

Manager m1,m2;

cout<<"Enter the details of Manager1\n";

m1.getdata();

cout<<"Enter the details of manager2\n";

m2.getdata();

cout<<"\t\tDetails of Manager1\n";

m1.putdata();

cout<<"\t\tDetails of Manager2\n";

m2.putdata();

return 0;

}

