/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Online C++ Compiler.

Code, Compile, Run and Debug C++ program online.

Write your code in this editor and press "Run" button to compile and execute it.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

using namespace std;

class standard

{

string std;

public:

standard()

{

this->std="NULL";

}

virtual void detail()

{

cout<<endl<<"Standard of the student is:"<<this->std;

}

virtual void input()

{

cout<<endl<<"Enter the standard of the student:";

cin>>this->std;

}

};

class section: public standard

{

string sect;

public:

section()

{

this->sect="NULL";

}

void detail()

{

cout<<endl<<"Section of the student is:"<<this->sect;

}

void input()

{

cout<<endl<<"Enter the section of the student:";

cin>>this->sect;

}

};

class rollnumber: public section

{

int rollno;

public:

rollnumber()

{

this->rollno=NULL;

}

void detail()

{

cout<<endl<<"Roll Number of the student is:"<<this->rollno;

}

void input()

{

cout<<endl<<"Enter the roll no. of the student:";

cin>>this->rollno;

}

};

class nameofstudent: public section

{

string name;

public:

nameofstudent()

{

this->name="NULL";

}

void detail()

{

cout<<endl<<"Name of the student is:"<<this->name;

}

void input()

{

cout<<endl<<"Enter the name of the student:";

cin>>this->name;

}

};

int main()

{

cout<<endl<<"Implementing Virtual Functions...........................................";

standard \*base,\*base1;

section child1;

rollnumber child2;

nameofstudent child3;

standard baseobj;

base=&baseobj;

base->input();

base=&child1;

base->input();

base=&child2;

base->input();

base=&child3;

base->input();

cout<<endl<<"Base pointer points to base class standard............................";

base = &baseobj;

cout<<endl<<"Base class object details:";

base->detail();

cout<<endl<<"Base pointer points to child class section...........................";

base=&child1;

cout<<endl<<"Child class object details:";

base->detail();

cout<<endl<<"Base pointer points to child class rollno...........................";

base=&child2;

cout<<endl<<"Child class object details:";

base->detail();

cout<<endl<<"Base pointer points to child class name...........................";

base=&child3;

cout<<endl<<"Child class object details:";

base->detail();

cout<<endl<<"Ending Program............................";

return 0;

}