**ASSIGNMENT 7 OOPS**

**NAME : SOUMIK GHOSH**

**ROLL NO: 21052924**

**CSE 21**

P1-A

//Multi-level inheritance.

#include<iostream>

using namespace std;

class A{

int a;

public:

void get(){

a=10;

}

void show(){

cout<<a;

}

};

class B: public A{

int b;

public :

void get(){

A::get();

b=20;

}

void show(){

cout<<b;

}

};

class C : public B{

int c;

public :

void get(){

B::get();

c=30;

}

};

int main()

{

C obj;

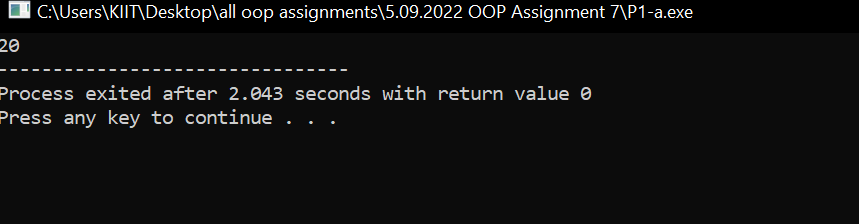
obj.get();

obj.show();

return 0;

}

OUTPUT :



P1-B

//Multiple inheritance

#include<iostream>

using namespace std;

class A{

public :

int a;

void get\_A(){

a=10;

}

};

class B{

public :

int b;

void get\_B(){

b=20;

}

};

class C : public A, public B{

public :

int c;

void get\_C(){

c=30;

}

void sum(){

cout<<a+b+c;

}

};

int main()

{

C obj;

obj.get\_A();

obj.get\_B();

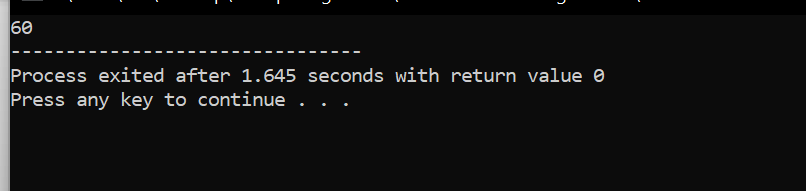
obj.get\_C();

obj.sum();

return 0;

}

OUTPUT :



P1-C

//HYBRID inheritance

#include<iostream>

using namespace std;

class A{

protected :

int a;

public :

void get\_a(){

cout<<"enter value of 'a' = "<<endl;

cin>>a;

}

};

class B:public virtual A{

protected :

int b;

public :

void get\_b(){

cout<<"enter value of 'b' = "<<endl;

cin>>b;

}

};

class C:public virtual A{

protected :

int c;

public :

void get\_c(){

cout<<"enter value of 'c' = "<<endl;

cin>>c;

}

};

class D: public B, public C{

protected :

int d;

public :

void multiply()

{

A::get\_a();

B::get\_b();

C::get\_c();

cout<<"The product of the three variables are : "<<a\*b\*c<<endl;

}

};

int main(){

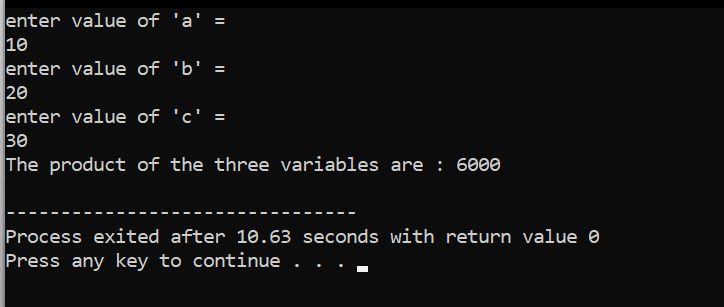
D object;

object.multiply();

return 0;

}

OUTPUT :



P1-D

//Single inheritance

#include<iostream>

using namespace std;

class A {

int a;

public :

int b;

void set\_ab();

int get\_a(void);

void show\_a(void);

};

class D : public A

{

int c;

public :

void mul(void);

void display(void);

};

void A :: set\_ab(void)

{

a=28;

b=76;

}

int A ::get\_a()

{

return a;

}

void A :: show\_a()

{

cout<< "a = "<<a<<"\n";

}

void D :: mul()

{

c= b\*get\_a();

}

void D :: display()

{

cout<<"a ="<<get\_a()<<"\n";

cout<<"b ="<<b<<endl;

cout<<"c ="<<c<<endl;

}

int main()

{

D obj;

obj.set\_ab();

obj.mul();

obj.display();

obj.b=20;

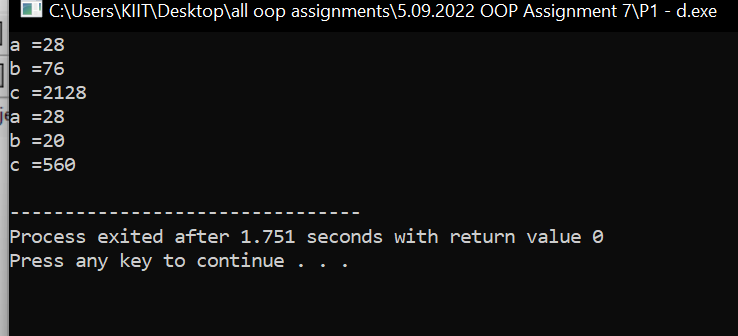
obj.mul();

obj.display();

return 0;

}

OUTPUT :



P2 .

#include<iostream>

using namespace std;

class student {

protected:

string name;

int rollno;

int age;

public:

void getdata(string name,int rollno,int age){

this->name=name;

this->age=age;

this->rollno=rollno;

}

void showdata(){

cout<<"Student Details ...\n";

cout<<"Name: "<<name<<endl;

cout<<"Roll No: "<<rollno<<endl;

cout<<"Age: "<<age<<endl;

}

};

class test : protected student{

protected :

int H,G,M,E,MS;

public:

void getdata(){

cout<<"Enter Name:";

// cin>>name;

getline(cin,name);

cout<<"Enter Rollno:";

cin>>rollno;

cout<<"Age:";

cin>>age;

cout<<"Enter History,Geography,Maths,English & Moral Science marks :";

cin>>H>>G>>M>>E>>MS;

student::getdata(name,rollno,age);

}

void showdata(){

student::showdata();

cout<<"Displaying Marks...\n";

cout<<"History :"<<H<<endl;

cout<<"Geography :"<<G<<endl;

cout<<"Maths :"<<M<<endl;

cout<<"English :"<<E<<endl;

cout<<"Moral Science :"<<MS<<endl;

}

};

int main()

{

test T;

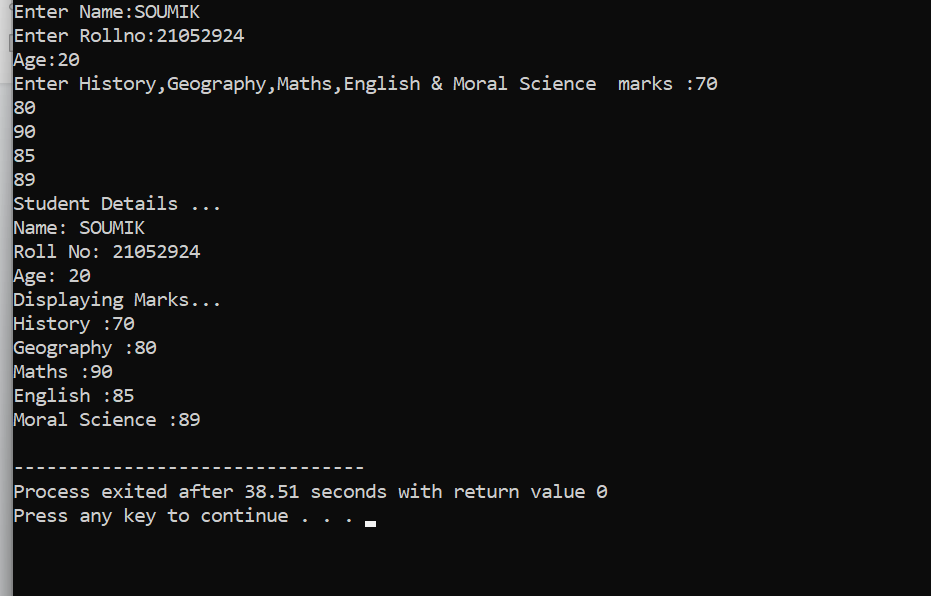
T.getdata();

T.showdata();

return 0;

}

OUTPUT :



P3:

#include<iostream>

using namespace std;

class student {

protected:

string name;

int rollno;

int age;

public:

void getdata(string name,int rollno,int age){

this->name=name;

this->age=age;

this->rollno=rollno;

}

void showdata(){

cout<<"Student Details ...\n";

cout<<"Name: "<<name<<endl;

cout<<"Roll No: "<<rollno<<endl;

cout<<"Age: "<<age<<endl;

}

};

class test : protected student{

protected :

int P,C,M,E,CS;

public:

void getdata(){

cout<<"Enter Name:";

cin>>name;

cout<<"Enter Rollno:";

cin>>rollno;

cout<<"Age:";

cin>>age;

cout<<"Enter Physics,Chemistry,Maths,English & CS marks :";

cin>>P>>C>>M>>E>>CS;

student::getdata(name,rollno,age);

}

void showdata(){

student::showdata();

cout<<"Displaying Marks...\n";

cout<<"Physics :"<<P<<endl;

cout<<"Chemistry :"<<C<<endl;

cout<<"Maths :"<<M<<endl;

cout<<"English :"<<E<<endl;

cout<<"Computer Science :"<<CS<<endl;

}

};

class result : protected test{

protected :

int tmarks;

float percent;

public:

void getdata(){

test::getdata();

tmarks=P+C+M+E+CS;

percent=(float)tmarks/(100\*5);

}

void showdata(){

test::showdata();

cout<<"\nAcademics Details ...\n";

cout<<"Total Marks:"<<tmarks<<endl;

cout<<"Total Percentage :"<<percent\*100<<endl;

}

};

int main()

{

result t1;

t1.getdata();

t1.showdata();

// result t2;

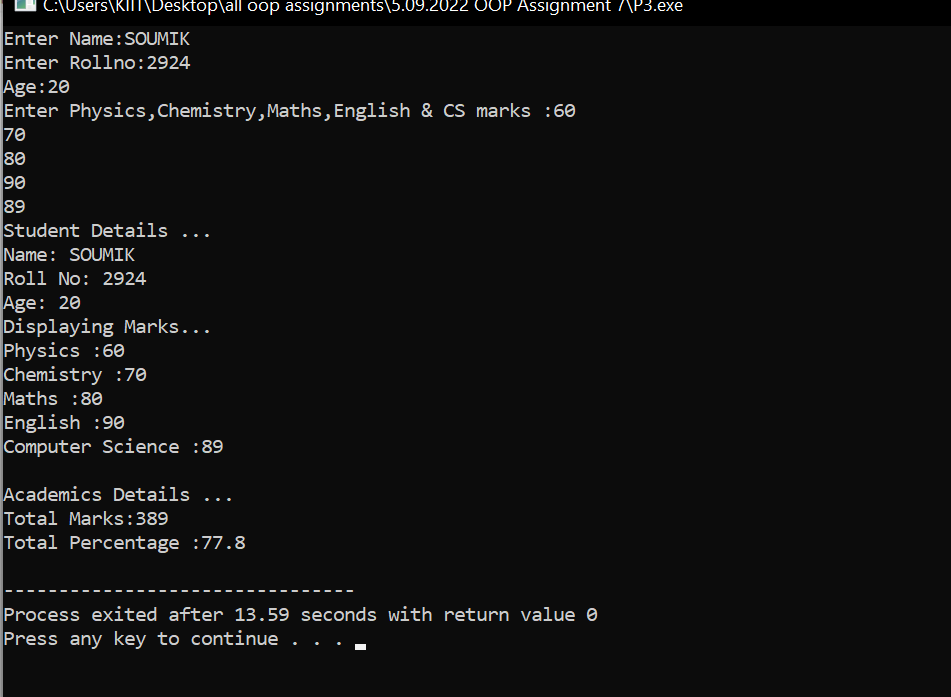
// t2.getdata();

// t2.showdata();

return 0;

}

OUTPUT :



P4.

#include<iostream>

using namespace std;

class student {

protected:

string name;

int rollno;

int age;

public:

void getdata(string name,int rollno,int age){

this->name=name;

this->age=age;

this->rollno=rollno;

}

void showdata(){

cout<<"Student Details ...\n";

cout<<"Name: "<<name<<endl;

cout<<"Roll No: "<<rollno<<endl;

cout<<"Age: "<<age<<endl;

}

};

class test : protected student{

protected :

int P,C,M,E,CS;

public:

void getdata(){

cout<<"Enter Name:";

cin>>name;

cout<<"Enter Rollno:";

cin>>rollno;

cout<<"Age:";

cin>>age;

cout<<"Enter Physics,Chemistry,Maths,English & CS marks :";

cin>>P>>C>>M>>E>>CS;

student::getdata(name,rollno,age);

}

void showdata(){

student::showdata();

cout<<"Displaying Marks...\n";

cout<<"Physics :"<<P<<endl;

cout<<"Chemistry :"<<C<<endl;

cout<<"Maths :"<<M<<endl;

cout<<"English :"<<E<<endl;

cout<<"Computer Science :"<<CS<<endl;

}

};

class sports {

protected :

int PE;

public:

void getdata(){

cout<<"\nEnter Sports Activity'Marks:"<<endl;

cin>>PE;

}

void showdata(){

cout<<"\nSport's Activity Marks :"<<PE<<endl;

}

};

class result : protected sports,protected test{

protected :

int tmarks;

float percent;

public:

void getdata(){

test::getdata();

sports::getdata();

tmarks=P+C+M+E+CS+PE;

percent=(float)tmarks/(100\*6);

}

void showdata(){

test::showdata();

sports::showdata();

cout<<"\nAcademics Details ...\n";

cout<<"Total Marks:"<<tmarks<<endl;

cout<<"Total Percentage :"<<percent\*100<<endl;

}

};

int main()

{

result t1;

t1.getdata();

t1.showdata();

// result t2;

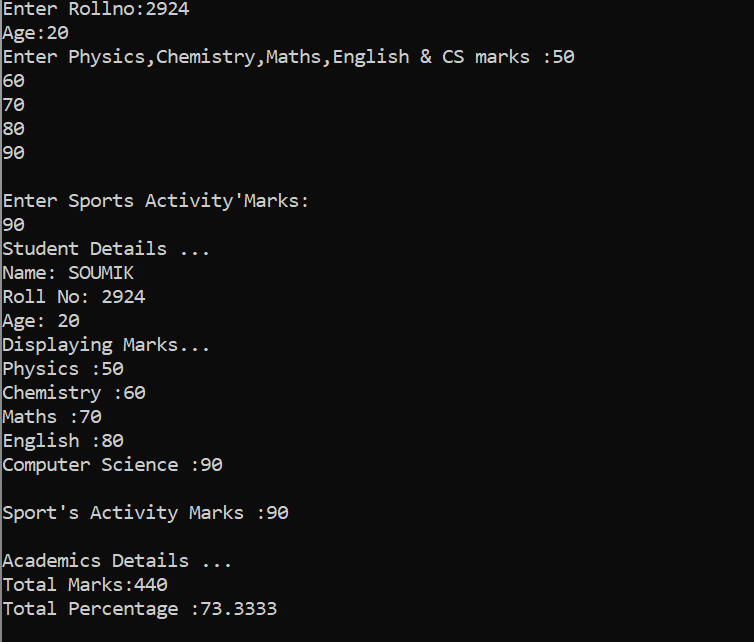
// t2.getdata();

// t2.showdata();

return 0;

}

OUTPUT :



P5:

#include<iostream>

using namespace std;

class shape {

public:

float l,b,r,h;

};

class circle:public shape{

public:

void getdata(){

cout<<"Enter Radius of Circle :";

cin>>r;

}

void showarea(){

cout<<"Area Of Circle :"<<3.14\*r\*r<<endl;

}

};

class Triangle : public shape{

public:

void getdata(){

cout<<"Enter Base and Height :";

cin>>b>>h;

}

void showarea(){

cout<<"Area Of Triangle :"<<0.5\*b\*h<<endl;

}

};

class Rectangle : public shape{

public:

void getdata(){

cout<<"Enter Length and Breadth :";

cin>>l>>b;

}

void showarea(){

cout<<"Area Of Rectangle :"<<l\*b<<endl;

}

};

int main()

{

Rectangle A1;

A1.getdata();

A1.showarea();

Triangle A2;

A2.getdata();

A2.showarea();

circle A3;

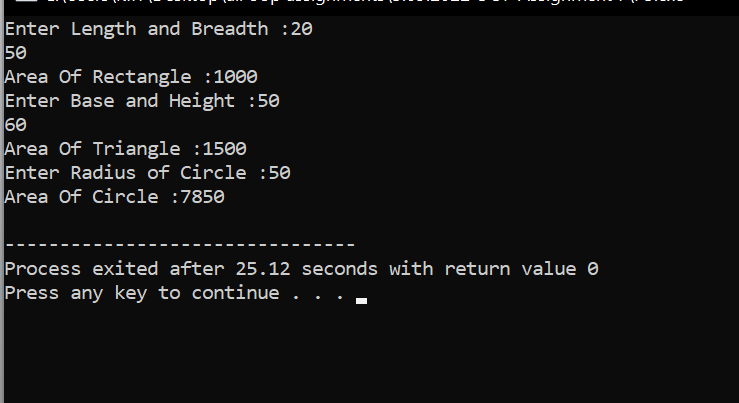
A3.getdata();

A3.showarea();

return 0;

}

OUTPUT :



P6:

#include<iostream>

using namespace std;

class Employee {

protected:

string empname;

int empid;

long long salary;

public:

void getdata(){

cout<<"Enter Employee Name :";

cin>>empname;

cout<<"Enter Employee ID :";

cin>>empid;

}

void showdata(){

cout<<"Employee details ...\n";

cout<<"Emp Name :";

cout<<empname<<endl;

cout<<"Emp ID :";

cout<<empid<<endl;

cout<<"Emp Salary :";

cout<<salary<<endl;

}

};

class Parttime : protected Employee{

protected:

int hh;

double pay=300;

public:

void getdata(){

Employee::getdata();

cout<<"\nNo of Hours work :";

cin>>hh;

salary=hh\*pay;

}

void showdata(){

Employee::showdata();

}

};

class Regular : protected Employee{

protected:

float hra=0.27;

float da=0.15;

long long bsalary;

public:

void getdata(){

Employee::getdata();

cout<<"\nEnter Basic Salary :";

cin>>bsalary;

salary=bsalary+bsalary\*hra+bsalary\*da;

}

void showdata(){

Employee::showdata();

}

};

int main()

{

Parttime e1;

e1.getdata();

e1.showdata();

Regular e2;

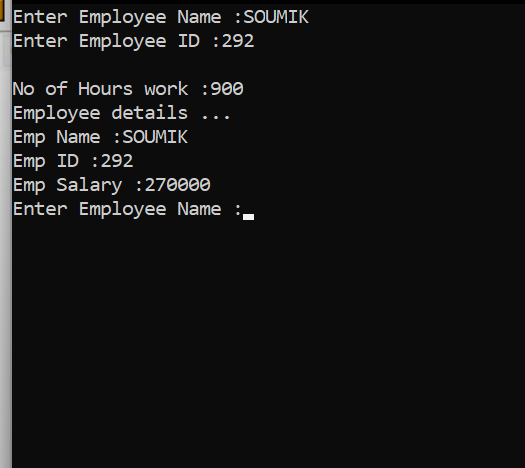
e2.getdata();

e2.showdata();

return 0;

}

OUTPUT :



P7:

#include<iostream>

#include<string.h>

using namespace std;

class Account{

private:

long long ac;

string name;

protected:

long long balance;

long long cash;

public:

void getdata(){

cout<<"Accessing A/c Details...\n";

cout<<"Enter A/C no :";

cin>>ac;

cin.ignore();

cout<<"Enter Customer Name :";

getline(cin, name);

cout<<"Enter Amount to deposit in new account:";

cin>>balance;

}

void showdata(){

cout<<"Displaying A/c Details...\n";

cout<<"A/C no :"<<ac<<endl;

cout<<"Enter Customer Name :"<<name<<endl;

cout<<"Enter Balance:"<<balance<<endl;

}

void withdraw(){

cout<<"Enter Amount to withdraw :";

cin>>cash;

if(balance>cash){

balance=balance-cash;

cout<<"Cash withdrawn successfully!"<<endl;

}

else{

cout<<"Insufficient Balance!"<<endl;

}

}

void deposit(){

cout<<"Enter Amount to Deposit :";

cin>>cash;

balance=balance+cash;

cout<<"Cash deposited successfully!"<<endl;

}

};

class Saving : public Account{

private:

long long mab=10000;

public:

void getdata(){

Account::getdata();

}

void deposit(){

Account::deposit();

}

void withdraw(){

Account::withdraw();

}

void Checker(){

if(balance>=mab){

cout<<"Your Account Balance is above Minimum A/c Balance!"<<endl;

}

else{

cout<<"Your Account Balance is below Minimum A/c Balance!"<<endl;

cout<<"Please deposit ! Amount"<<endl;

}

}

void showdata(){

Account::showdata();

}

};

class Current : public Account{

protected:

long long credit=100000;

public:

void getdata(){

Account::getdata();

}

void deposit(){

Account::deposit();

}

void withdraw(){

Account::withdraw();

}

void Overdue(){

balance=balance-credit;

cout<<"Your Over Due Account Balance -"<<balance<<endl;

cout<<"Please deposit ! Amount"<<endl;

}

void showdata(){

Account::showdata();

}

};

int main()

{

Saving C1;

Current C2;

int option;

while(option!='\n'){

cout<<"Account Mode :\tPress\t 1.SAVING \t 2.CURRENT"<<endl;

cin>>option;

if(option==2){

C2.getdata();

C2.deposit();

C2.withdraw();

C2.showdata();

C2.Overdue();

}

else{

C1.getdata();

C1.deposit();

C1.withdraw();

C1.showdata();

C1.Checker();

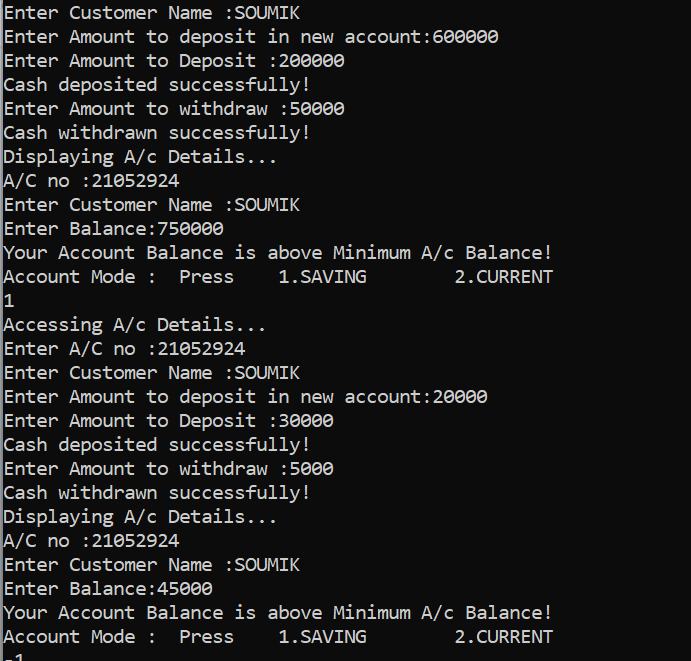
}

}

return 0;

}

OUTPUT :



IK