1. Example of Static Data Member and Static Member Function:

#include<iostream>

using namespace std;

class A {

public:

int a,b;

static int c;

void increment ()

{

a++;

b++;

c++;

}

void display()

{

cout<<"\n a is:"<<a;

cout<<"\n b is:"<<b;

cout<<"\n c is:"<<c<<endl;

}

static void show()

{

cout<<"\n c is :"<<c;

}

};

int A :: c;

int main()

{

A ob1,ob2,ob3;

ob1.a=10;

ob1.b=5;

ob2.a=7;

ob2.b=12;

ob1.display();

ob1.increment();

ob1.display();

ob2.increment();

ob3.a=5;

ob3.b=7;

ob3.increment();

ob2.display();

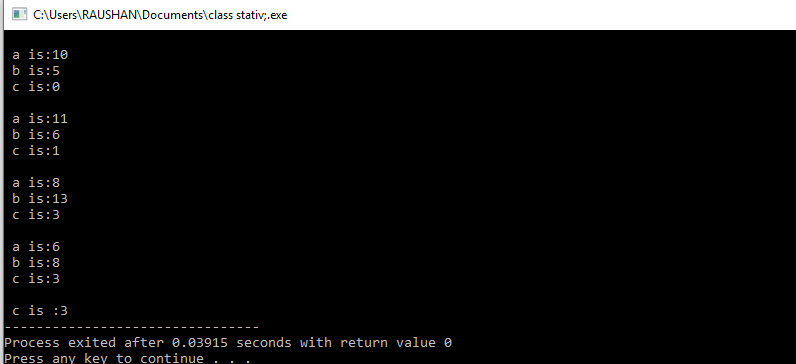
ob3.display();

A::show();

return 0;

}

OUTPUT



1. Use of Const:

Write a program to create a class number having two data member of class of type integer. Create function for input and display

Ans:

#include<iostream>

using namespace std;

class number {

private:

int a,b;

public:

void input ( int x, int y)

{

a=x;

b=y;

}

void display () const;

};

void number :: display() const

{

cout<<"\n a is:"<<a;

cout<<"\n b is:"<<b<<endl;

}

int main()

{

number n1, n2;

n1.input(5,7);

n1.display();

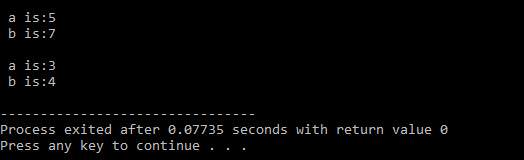
n2.input(3,4);

n2.display();

return 0;

}

OUTPUT:



1. Empty class :

#include<iostream>

using namespace std;

class empty {

};

int main()

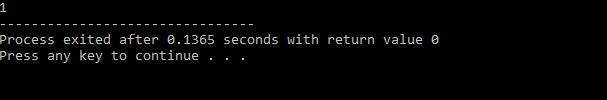
{

cout<<sizeof(empty);

return 0;

}

OUTPUT:



4.Friend function:

#include<iostream>

using namespace std;

class number {

private:

int a,b;

public:

void input( int x, int y)

{

a=x;

b=y;

}

void display()

{

cout<<"\n a is :"<<a;

cout<<"\n b is :"<<b;

}

friend int add (number ob);

};

int add ( number ob)

{

int r=ob.a+ob.b;

return r;

}

int main()

{

number ob1, ob2;

ob1.input(2,3);

ob1.display();

cout<<"\n sum is :" <<add(ob1)<<endl;

ob2.input(5,9);

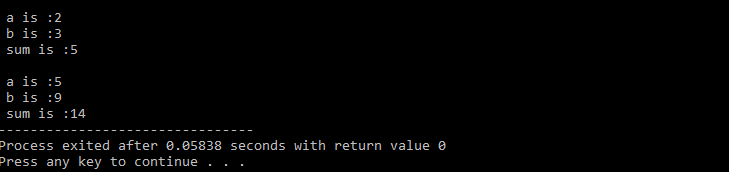
ob2.display();

cout<<"\n sum is :"<<add(ob2);

return 0;

}

OUTPUT:



5.Bit Field and Class

#include<iostream>

using namespace std;

#define diesel 2

#define petrol 1

#define old 5

#define new 7

class vehicle {

public:

unsigned int type :2;

unsigned int model: 3;

void display()

{

if (type==1)

{

cout<<"vehicle is of type petrol";

}

else if (type ==2)

{

cout<<"vehicle is of type diesel";

}

if (model==5)

{

cout<<"\n model is old";

}

else if (model ==7)

{

cout<<"\n model is new";

}

}

};

int main()

{

vehicle v1;

v1.type=petrol;

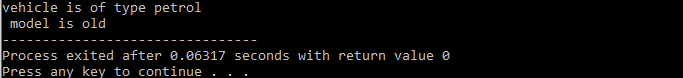
v1.model=old;

v1.display();

return 0;

}

OUTPUT:



6.Nested Class

#include<iostream>

using namespace std;

class outer {

public:

class inner {

public:

int a;

void show()

{

cout<<"\n a is :"<<a;

}

void input(int x)

{

a=x;

}

};

};

int main()

{

outer::inner ob1;

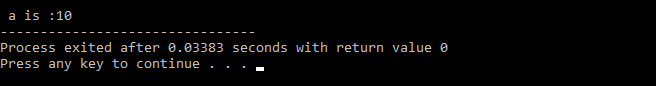
ob1.input(10);

ob1.show();

return 0;

}

OUTPUT:



7.Constructor, use of default , argumented and copy constructor in a program.

#include<iostream>

using namespace std;

class student {

private:

int age;

int roll\_no;

public:

student()

{

age=10;

roll\_no=2;

}

void display()

{

cout<<"\n age is: "<<age;

cout<<"\n roll no is:"<<roll\_no<<endl;

}

student( int x,int y)

{

age=x;

roll\_no=y;

}

student (student &ob)

{

age=ob.age;

roll\_no=ob.roll\_no;

}

};

int main()

{

student ob1;

ob1.display();

student ob2(5,7);

ob2.display();

student ob3(ob2);

ob3.display();

student ob4(ob1);

ob4.display();

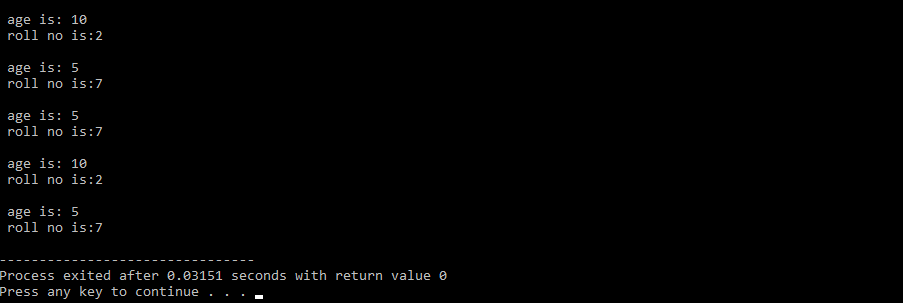
student ob5(ob3);

ob5.display();

return 0;

}

OUTPUT:



8. Write a program having two data member real and imaginary of type float.Create a function fo displaying the value program must contain a parameterized and copy constructor.

Ans:

#include<iostream>

using namespace std;

class complex {

private:

float real;

float imag;

public:

complex ()

{

}

complex( float x,float y)

{

real=x;

imag=y;

}

void display();

complex( complex &ob)

{

real=ob.real;

imag=ob.imag;

}

};

void complex :: display()

{

cout<<"\n real is :"<<real;

cout<<" \n imag is :"<<imag<<endl;

}

int main()

{

complex ob1( 5.33, 6.98);

ob1.display();

complex ob2(ob1);

ob2.display();

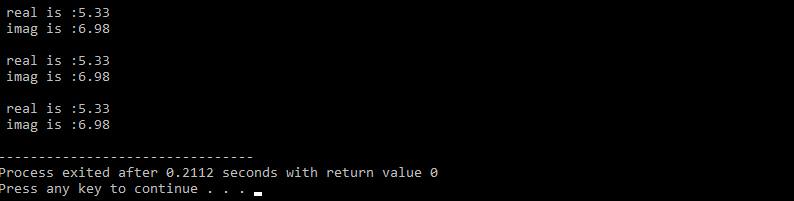
complex ob3(ob2);

ob3.display();

return 0;

}

OUTPUT:



9.Create a class number which will have three data member of type integer.Create function for input and display. Show functionality using two object.

Ans:

#include<iostream>

using namespace std;

class number {

private:

int x,y,z;

public:

void input( int p, int q, int r)

{

x=p;

y=q;

z=r;

}

void display()

{

cout<< "\n value of x :" <<x;

cout<<"\n value of y :"<<y;

cout<<"\n valur of z :"<<z<<endl;

}

};

int main()

{

number ob1, ob2;

ob1.input(4, 7, 8);

ob1.display();

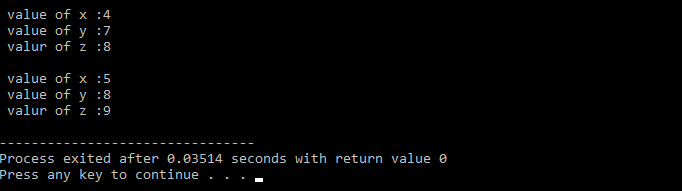
ob2.input(5,8,9);

ob2.display();

return 0;

}

OUTPUT:



10.DESTRUCTOR:

Use of destructor in class.

#include<iostream>

using namespace std;

class student{

public:

static int count;

student ()

{

count++;

cout<<"\n object created:"<<count;

}

~student ()

{

count--;

cout<<"\n onject destroyed:"<<count;

}

};

int student:: count;

int main()

{

student ob1, ob2;

{

student ob3;

}

{

student ob4, ob5;

}

return 0;

}

OUTPUT:

