#include<iostream>

using namespace std;

class Rational

{

public:int  denominator;

Rational()

                {

                }

Rational(int i)

                {

                denominator=i;

                }

**friend Rational operator + (const Rational &, int);// adding constant to obj using friend function**

**friend Rational operator + (int, const Rational &);//adding constant to obj using friend function**

    void show()

    {

                cout<<denominator<<endl;

                }

};

Rational operator + (const Rational &r, int i)

{

    Rational temp;

                temp.denominator =  r.denominator+i;

                return temp;

}

Rational operator + (int i, const Rational &r)

{

    Rational temp;

                temp.denominator =  i+r.denominator;

                return temp;

}

int main()

{

    Rational r(10);

    Rational result1 = r + 1; // works

    result1.show();

    Rational result2 = 1 + r; // works

    result2.show();

}

**Adding obj and a constant by creating member function**

#include<iostream>

using namespace std;

class Rational

{

public:int  denominator;

Rational()

                {

                }

Rational(int i)

                {

                denominator=i;

                }

**Rational operator+(int i);// dding obj and a constant by creating member function**

    void show()

    {

                cout<<denominator<<endl;

                }

};

**Rational Rational:: operator + ( int i)**

**{**

**Rational temp;**

**temp.denominator =  denominator+i;**

**return temp;**

**}**

int main()

{

    Rational r(10);

    Rational result1 = r + 1; // works

    result1.show();

    Rational result2 = 1 + r; // does not works

    result2.show();

}

}