ABSTRACT CLASS

#include<iostream>

#include<iostream>

#include<string>

using namespace std;

class animal

{

public:

virtual void eat()=0;

};

class dog:public animal

{

public:

void eat()

{

cout<<"dog eats bones";

}

};

class cat:public animal

{

public:

void eat()

{

cout<<"cat eats food"<<endl;

}

};

int main()

{

cat c1;

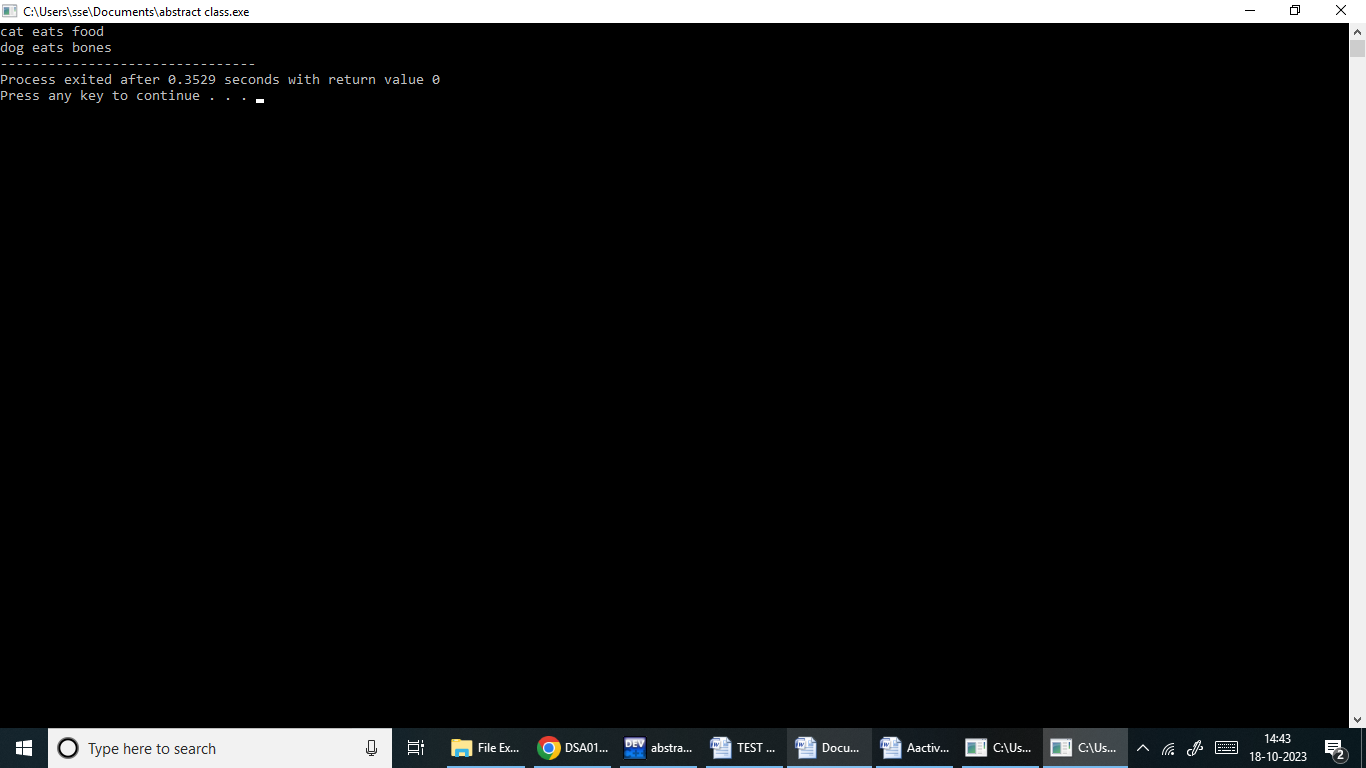
dog d1;

c1.eat();

d1.eat();

return 0;

}



Friend function with operator

#include <iostream>

using namespace std;

class B;

class A {

private:

double a;

public:

A() {

cout << "Enter the number: ";

cin >> a;

}

friend void operator+(A, B);

friend void operator-(A, B);

friend void operator\*(A, B);

friend void operator/(A, B);

};

class B {

private:

double b;

public:

B() {

cout << "Enter the number: ";

cin >> b;

}

friend void operator+(A, B);

friend void operator-(A, B);

friend void operator\*(A, B);

friend void operator/(A, B);

};

void operator+(A A1,B B1)

{

cout<<"Addition : "<<A1.a+B1.b<<"\n";

}

void operator-(A A1,B B1)

{

cout<<"Subtraction :"<<A1.a-B1.b<<"\n";

}

void operator\*(A A1,B B1)

{

cout<<"Multiplication :"<<A1.a\*B1.b<<"\n";

}

void operator/(A A1,B B1)

{

cout<<"Division :"<<A1.a/B1.b<<"\n";

}

int main() {

A w;

B e;

w+e;

w-e;

w\*e;

w/e;

return 0;

}

