TASK 1:

Person::Person(int ) called

Faculty::Faculty(int ) called

Person::Person(int ) called

Student::Student(int ) called

TA::TA(int ) called

3 FILE:

MAIN:

#include<iostream>

#include"Header.h"

using namespace std;

int main() {

TA ta1(30);

}

DEFINITON 2:

#include"Header.h"

Person::Person(int x) { cout << "Person::Person(int ) called" << endl; }

Person::Person() { cout << "Person::Person() called" << endl; }

Faculty::Faculty(int x) :Person(x) {

cout << "Faculty::Faculty(int ) called" << endl;

}

Student::Student(int x) :Person(x) {

cout << "Student::Student(int ) called" << endl;

}

TA::TA(int x) :Student(x), Faculty(x) {

cout << "TA::TA(int ) called" << endl;

}

HEADER :

#include<iostream>

using namespace std;

class Person {

public:

Person(int x);

Person();

};

class Faculty : virtual public Person {

public:

Faculty(int x);

};

class Student : virtual public Person {

public:

Student(int x);

};

class TA : public Faculty, public Student {

public:

TA(int x);

};

Text

Description automatically generated

TASK 2:

MAIN:

#include<iostream>

#include"header.h"

#include<string>

using namespace std;

int main()

{

Cat a("zxe");

a.Sound();

Deer b("ase");

b.Sound();

Dog c("asd");

c.Sound();

Lion d("asd");

d.Sound();

Tiger e("asd");

e.Sound();

}

HEADER:

#include<iostream>

using namespace std;

class Animal {

string name;

public:

Animal(string name)

{

this->name = "xyz";

}

string name1(string name)

{

return name;

}

void name2(string name)

{

this->name = name;

}

void Sound()

{

cout << endl;

cout << "Sound of the ::" << this->name ;

}

};

class Cat :public Animal {

public:

Cat(string name) :Animal(name) {

name = "Cat";

}

void Sound()

{

name2(name1("Cat"));

Animal::Sound();

}

};

class Dog : public Animal {

public:

Dog(string name) :Animal(name)

{

name = "Dog";

}

void Sound()

{

name2(name1("Dog"));

Animal::Sound();

}

};

class Tiger\_Fmaily : public Animal {

string name;

public:

Tiger\_Fmaily(string name) :Animal(name)

{

}

string name1(string name)

{

return name;

}

void name2(string name)

{

this->name = name;

}

void Sound()

{

Animal::name2(Animal::name1(this->name));

Animal::Sound();

}

};

class Deer :public Animal {

public:

Deer(string Deer) :Animal(Deer)

{

Deer = "Deer";

}

void Sound()

{

name2(name1("Deer"));

Animal::Sound();

}

};

class Tiger :public Tiger\_Fmaily {

public:

Tiger(string name) : Tiger\_Fmaily(name)

{

name = "Tiger";

}

void Sound()

{

name2(name1("Tiger"));

Tiger\_Fmaily::Sound();

}

};

class Lion :public Tiger\_Fmaily {

public:

Lion(string name) : Tiger\_Fmaily(name)

{

name = "Lion";

}

void Sound()

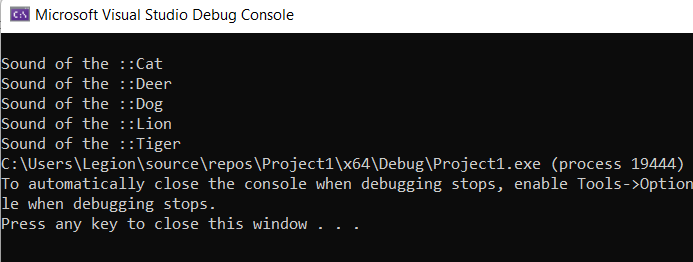
{

name2(name1("Lion"));

Tiger\_Fmaily::Sound();

}

};



TASK 3:

MAIN:

#include<iostream>

#include"Header.h"

#include<string>

#include<cstring>

using namespace std;

int main(){

cout << "Please inut Number of the Ships you want ::";

int n;

cin >> n;

Ship\* a = new CargoShip[n];

Ship\* b = new CuriseShip[n];

cout << "Input the Data of the Cargo ship ";

cout << endl;

for (int i = 0; i < n; i++){

cout << "Data No ::" << i;

cout << endl;

cout << "Please input the Name of the Ship ::";

string name;

cin >> name;

a[i].name2(name);

cout << "Please input the Year of The Ship ::";

float year;

cin >> year;

a[i].year2(year);

}

cout << "Your data Stored In CargoShip ";

cout << endl;

for (int i = 0; i < n; i++){

cout << "Data No ::" << i;

cout << endl;

a[i].print();

}

cout << "Input the Data of the CuriseShip";

cout << endl;

for (int i = 0; i < n; i++){

cout << "Data No ::" << i;

cout << endl;

cout << "Please input the Name of the Ship ::";

string name;

cin >> name;

b[i].name2(name);

cout << "Please input the Year of The Ship ::";

float year;

cin >> year;

b[i].year2(year);

}

cout << "Your data Stored In CuriseShip ";

cout << endl;

for (int i = 0; i < n; i++){

cout << "Your Data No ::" << i;

cout << endl;

b[i].print();

}

}

SOURCE :

#include<iostream>

#include"Header.h"

Ship::Ship() {

this->name = "XYZ";

this->year = 123;

}

string Ship :: name1() {

return name;

}

void Ship :: name2(string const name) {

this->name = name;

}

float Ship :: year1() {

return year;

}

void Ship:: year2(float const year)

{

this->year = year;

}

void Ship :: print() {

cout << "The The Name of the Ship";

cout << endl;

cout << this->name;

cout << "The yaer of the Ship";

cout << this->year;

cout << endl;

}

int CuriseShip :: passengers1() {

int passengers;

cin >> passengers;

return passengers;

}

void CuriseShip :: passengers2(int const passengers) {

this->passengers = passengers;

}

void CuriseShip:: name() {

name2(name1());

}

void CuriseShip:: year() {

year2(year1());

}

void CuriseShip:: print() {

cout << "The Name of the Ship is ::";

cout << name1();

cout << endl;

cout << "The Maximum Number of the Passengers are ::";

cout << this->passengers;

cout << endl;

}

int CargoShip :: tonnage1() {

int tonnage;

cin >> tonnage;

return tonnage;

}

void CargoShip:: tonnage2(int const tonnage) {

this->tonnage = tonnage;

}

void CargoShip:: name() {

name2(name1());

}

void CargoShip::year() {

year2(year1());

}

void CargoShip::print() {

cout << "The Name of the Ship is ::";

cout << name1();

cout << endl;

cout << "The Maximum Tons of the Passengers is ::";

cout << this->tonnage;

cout << endl;

}

HEADER :

#pragma once

#include<iostream>

#include<string>

#include<cstring>

using namespace std;

class Ship {

string name;

float year;

public:

Ship();

string name1();

void name2(string const name);

float year1();

void year2(float const year);

virtual void print();

};

class CuriseShip :public Ship {

int passengers;

public:

CuriseShip() :Ship() {

this->passengers = 56;

}

int passengers1();

void passengers2(int const passengers);

void name();

void year();

void print();

};

class CargoShip :public Ship {

int tonnage;

public:

CargoShip() :Ship() {

this->tonnage = 90;

}

int tonnage1();

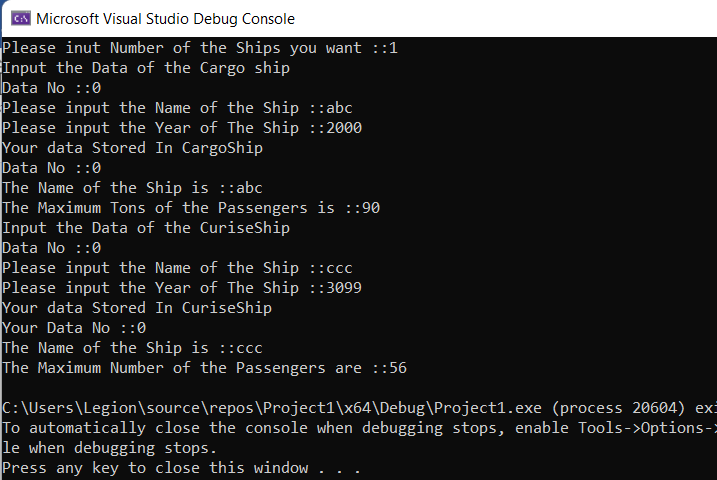
void tonnage2(int const tonnage);

void name();

void year();

void print();

};



TASK 4:

MAIN:

#include<iostream>

#include"header.h"

using namespace std;

int main()

{

polygon\* Bptr = new Triangle;

Bptr->Display();

delete Bptr;

Bptr = new Square;

Bptr->Display();

delete Bptr;

Bptr = new Rectangle;

Bptr->Display();

delete Bptr;

}

HEADER :

#include<iostream>

using namespace std;

class polygon

{

float length;

float width;

public:

polygon()

{

this->length = 33;

this->width = 12;

}

virtual float Area()=0

{

float Area1;

Area1 = (this->length) \* (this->width);

return Area1;

}

virtual float Perimeter()=0

{

float Perimeter1;

Perimeter1 = (this->length) + (this->width);

return Perimeter1;

}

virtual void Display()=0

{

cout << "The Width ::";

cout << this->width;

cout << endl;

cout << "The Length ::";

cout << this->length;

cout << endl;

cout << "The Perimter of polygon::";

cout << Perimeter();

cout << endl;

cout << "The Area of polygon";

cout << Area();

cout << endl;

}

};

class Square : public polygon

{

float length;

float width;

public:

Square() :polygon()

{

this->length = 17;

this->width = 18;

}

float Area()

{

float Area;

Area = length \* width;

return Area;

}

float Perimeter()

{

float Perimter1;

Perimter1 = 4 \* length;

return Perimter1;

}

virtual void Display()

{

cout << "The Width ::";

cout << this->width;

cout << endl;

cout << "The Length ::";

cout << this->length;

cout << endl;

cout << "The Perimter of Square ::";

cout << Perimeter();

cout << endl;

cout << "The Area of Square ::";

cout << Area();

cout << endl;

}

};

class Rectangle :public polygon {

float length;

float width;

public:

Rectangle() :polygon()

{

this->length = 98;

this->width = 90;

}

float Area()

{

float Area1;

Area1 = 2 \* (length \* width);

return Area1;

}

float Perimeter()

{

float Perimter1;

Perimter1 = 2 \* (length + width);

return Perimter1;

}

virtual void Display()

{

cout << "The Width ::";

cout << this->width;

cout << endl;

cout << "The Length ::";

cout << this->length;

cout << endl;

cout << "The Perimter of the Rectangle::";

cout << Perimeter();

cout << endl;

cout << "The Area of Rectangle";

cout << Area();

cout << endl;

}

};

class Triangle :public polygon {

float length;

float width;

public:

Triangle() :polygon()

{

this->length = 11;

this->width = 43;

}

float Area()

{

float Area;

Area = (length \* width) / 2;

return Area;

}

float Perimeter()

{

float Perimter1;

Perimter1 = (width + length);

return Perimter1;

}

virtual void Display()

{

cout << "The Width ::";

cout << this->width;

cout << endl;

cout << "The Length ::";

cout << this->length;

cout << endl;

cout << "The Perimter of Triangle ::";

cout << Perimeter();

cout << endl;

cout << "The Area of Triangle ::";

cout << Area();

cout << endl;

}

};

