#include <iostream>

using namespace std;

//class Person

//{

//public:

// virtual void whoAmI()

// {

// cout << "I am a person" << endl;

// }

// virtual string GetClassName()

// {

// return "Person";

// }

//};

//

//class Superman :public Person

//{

// void whoAmI() override

// {

// cout << "I am a superman" << endl;

// }

// string GetClassName() override

// {

// return "Superman";

// }

//};

//

//class Batman :public Person

//{

// void whoAmI() override

// {

// cout << "I am a batman" << endl;

// }

// string GetClassName() override

// {

// return "Batman";

// }

//};

//

//int main()

//{

// auto persons = new Person \* [3]

// {

// new Person,

// new Superman,

// new Batman

// };

//

// for (int x = 0; x < 3; x++)

// {

// persons[x]->whoAmI();

// }

//

//

// return 0;

//}

//#pragma region Abstraction

//

//class Figure

//{

//protected:

// double area;

//public:

// virtual void WhatIsMyShape() = 0; // pure virtual method

// virtual void CalculateArea() = 0; // pure virtual method

//

// double GetArea() const

// {

// return area;

// }

//};

//

//class Rectangle:public Figure

//{

// double length;

// double width;

//public:

// Rectangle(double length, double width)

// :length(length), width(width) {}

//

//

// // Inherited via Figure

// virtual void WhatIsMyShape() override

// {

// cout << "My Shape Is Rectangle" << endl;

// }

// virtual void CalculateArea() override

// {

// area = length \* width;

// }

//

//};

//

//class Square :public Rectangle

//{

// double side;

//public:

// Square(int side) : Rectangle(side, side) {}

//

// void WhatIsMyShape() override

// {

// cout << "My Shape Is Square" << endl;

// }

//};

//

//class Circle:public Figure

//{

// int radius;

//public:

// Circle(int radius) : radius(radius) {}

//

// // Inherited via Figure

// virtual void WhatIsMyShape() override

// {

// cout << "My Shape Is Circular" << endl;

// }

//

// virtual void CalculateArea() override

// {

// area = 3.141516 \* radius \* radius;

// }

//};

//

//int main()

//{

// //Figure\* f;

// //Rectangle r(10, 20);

// //r.WhatIsMyShape();

// //r.CalculateArea();

// //cout << "Area : " << r.GetArea() << endl;

//

// //Figure\* figure = new Figure(); // error becuase Figure is abstract class

// Figure\* rectangle = new Rectangle(10, 20);

// Figure\* circle = new Circle(10);

// Figure\* square = new Square(20);

//

// Figure\*\* figures = new Figure \* [3]{ rectangle,circle,square };

//

// for (int x = 0; x < 3; x++)

// {

// figures[x]->WhatIsMyShape();

// figures[x]->CalculateArea();

// cout << "Area : " << figures[x]->GetArea() << endl;

// }

//

// //Circle circle2(5);

// //circle2.WhatIsMyShape();

// //circle2.CalculateArea();

// //cout << circle2.GetArea() << endl;

//

// return 0;

//}

//#pragma endregion

//class Animal

//{

//protected:

// string name;

//public:

// Animal(const string& name) : name(name) {}

//

// string GetName() const

// {

// return name;

// }

//

// virtual void Speak() = 0

// {

// cout << "I cry like all animals" << endl;

// }

//};

//

//class Dragon : public Animal

//{

//public:

// Dragon(const string& name) : Animal(name) {}

//

// // Inherited via Animal

// virtual void Speak() override

// {

// Animal::Speak();

// }

//

//

//

//};

//

//int main()

//{

//

// Animal\* dragon = new Dragon("bomba Ejdaha");

// dragon->Speak();

//

// return 0;

//}

//

//class AbstractDatabase

//{

//public:

// virtual string GetById(int id) = 0;

// virtual void DeleteById(int id) = 0;

// virtual void Add(int id, string data) = 0;

// virtual void Update(int id, string data) = 0;

//};

//

//class OracleDb :public AbstractDatabase

//{

// // Inherited via AbstractDatabase

// virtual string GetById(int id) override

// {

// cout << "Get From Oractle Database" << endl;

// return "";

// }

// virtual void DeleteById(int id) override

// {

// cout << "Delete From Oractle Database" << endl;

// }

// virtual void Add(int id, string data) override

// {

// cout << "Add To Oracle Database" << endl;

// }

// virtual void Update(int id, string data) override

// {

// cout << "Update In Oracle Database" << endl;

// }

//};

//

//class MsSql :public AbstractDatabase

//{

//public:

// // Inherited via AbstractDatabase

// virtual string GetById(int id) override

// {

// cout << "Get From MsSql Database" << endl;

// return "";

// }

//

// virtual void DeleteById(int id) override

// {

// cout << "Delete From MsSql Database" << endl;

// }

//

// virtual void Add(int id, string data) override

// {

// cout << "Add To MsSql Database" << endl;

// }

//

// virtual void Update(int id, string data) override

// {

// cout << "Update At MsSql Database" << endl;

// }

//};

//

//class Website

//{

// string url;

// AbstractDatabase\* database;

//public:

// Website(AbstractDatabase\* db)

// {

// database = db;

// }

//

// void MakeSomething()

// {

// database->Add(1, "New Data");

// database->DeleteById(2);

// database->DeleteById(116);

//

// // some codes

//

// database->GetById(10);

// }

//};

//

//int main()

//{

// Website website(new OracleDb());

// Website website2(new MsSql());

//

// website.MakeSomething();

// website2.MakeSomething();

//

//

// return 0;

//}

//class Musician

//{

//protected:

// string name;

// string instrument;

//public:

// Musician() = delete;

// Musician(string name, string instrument) : name(name),instrument(instrument) {}

//

// virtual ~Musician() = 0

// {

//

// }

//};

//

//class A :public Musician

//{

//public:

// A(string s1,string s2) : Musician(s1,s2)

// {

//

// }

//

//

//

//};

//

//int main()

//{

// A a("", "");

//

//

// return 0;

//}

//class Musician

//{

//protected:

// string name;

// string instrument;

//public:

// Musician() = delete;

//

// Musician(string n, string i)

// {

// name = n;

// instrument = i;

// }

//

// virtual void perform() = 0

// {

// cout << "Sazina qurban . . . . . . Sozune qurban" << endl;

// }

// virtual ~Musician() = 0 // pure virtual destructor

// {

//

// }

//};

//

//class Guitarist:public Musician

//{

// string nickname;

//public:

// Guitarist() = delete;

//

// Guitarist(string name,string instrument,string nickname) : Musician(name,instrument)

// {

// this->nickname = nickname;

// }

//

// void perform() override

// {

// cout << "Guitar sound" << endl;

// }

//};

//

//class Ashiq :public Musician

//{

// bool isEloglu;

//public:

// Ashiq() = delete;

// Ashiq(string name,string instrument) : Musician(name,instrument)

// {

// this->isEloglu = true;

// }

//

// void perform() override

// {

// Musician::perform();

// }

//};

//

//class Band

//{

// Musician\*\* musicians;

// int count = 0;

//public:

// void AddMusician(Musician\* musician)

// {

// auto newdata = new Musician \* [count + 1]{};

// for (int x = 0; x < count; x++)

// {

// newdata[x] = musicians[x];

// }

// newdata[count] = musician;

// ++count;

// musicians = newdata;

// newdata = nullptr;

// }

//

// void Start()

// {

// for (size\_t i = 0; i < count; i++)

// {

// musicians[i]->perform();

// }

// }

//

//

//

//};

//

//int main()

//{

// Band stars;

// stars.AddMusician(new Guitarist("Jordon", "Guitar", "John"));

// stars.AddMusician(new Ashiq("Telli", "Saz"));

// stars.AddMusician(new Ashiq("Arif Telsiz", "Saz"));

//

// stars.Start();

//

//}