#include<iostream>

using namespace std;

//class Test {

// int\* id;

// int age;

//

//public:

// Test()

// {

// age = 0;

// id = new int(0);

// }

// Test(int id, int age)

// {

// this->id = new int(id);

// this->age = age;

// }

// ~Test()

// {

// if (id != nullptr) {

// delete id;

// id = nullptr;

// }

// }

//

//};

//

//

//void main() {

//

// Test t(10, 35);

//

//

//}

//void demo() {

// static int id = 0;

// ++id;

// cout << id << endl;

//}

//

//

//void main() {

// demo();

// demo();

// demo();

//}

//class Network {

//public:

// static char\* IP;

// static int port;

// void Connect() {

// cout << "Connected" << endl;

// }

//

//};

//

//char\* Network::IP = new char [] {"192.168.1.1"};

//int Network::port = 4200;

//void main() {

//

//

// cout << Network::IP << endl;

// cout << Network::port << endl;

//

//}

//#include<assert.h>

//

//

//class Airplane {

// int myid;

// char\* model;

// char\* color;

// int engines;

// int capacity;

//public:

// static int ID;

// static int MAX;

// Airplane(const char\* model, const char\* color, const int& e,

// const int& c)

// {

// assert(ID < MAX && "Our garage is full");

// this->myid = ++ID;

// }

// void Show() {

// cout << "Airplane ID : " << this->myid << endl;

// }

// static int getCurrentId() {

// return ID;

// }

//

//};

//int Airplane::ID = 0;

//int Airplane::MAX = 5;

//

//void main() {

//

// Airplane a1("Boend-747", "White", 4, 120);

// Airplane a2("Airbus", "Blue", 4, 200);

// Airplane a3("T-300", "Black", 2, 4);

// Airplane a4("T-300", "Black", 2, 4);

// Airplane a5("T-300", "Black", 2, 4);

// Airplane a6("T-300", "Black", 2, 4);

//

// a1.Show();

// a2.Show();

// a3.Show();

// cout << Airplane::getCurrentId() << endl;

//}

//Task

//Product id,name,company,price,discount,description

//static id,method : getCurrentProductId

//

//getpriceWithDiscount

//showProduct

//#include<assert.h>

//class User {

// char\* username;

// char\* password;

//public:

// User() = default;

// User(const char\* username, const char\* pass) {

// SetUsername(username);

// SetPassword(pass);

// }

//

// User(const User& other) {

// SetUsername(other.username);

// SetPassword(other.password);

// }

//

// char\* getUsername() {

// return username;

// }

// char\* getPassword() {

// return password;

// }

//

// void SetUsername(const char\* username) {

// assert(username != nullptr && "username should be fill");

// int l = strlen(username);

// this->username = new char[l + 1]{};

// strcpy\_s(this->username, l + 1, username);

// }

// void SetPassword(const char\* pass) {

// assert(pass != nullptr && "username should be fill");

// int l = strlen(pass);

// this->password = new char[l + 1]{};

// strcpy\_s(this->password, l + 1, pass);

// }

// void Show() {

// cout << "Username : " << this->username << endl;

// cout << "Password : ";

// int l = strlen(this->password);

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

// {

// cout << "\* ";

// }cout << endl;

// }

//

// ~User()

// {

// delete[]username;

// delete[]password;

// }

//};

//

//

//void main() {

//

// User u("Elvin123", "Elvin\_123");

// u.Show();

// User u2(u);

//

// //Shallow copy

// // u2.getUsername=u.username

// //Deep copy

//}

//Store s(10,products)

//

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

//{

// this->products[i]=products[i]

//}

//#include<assert.h>

//

//

//class Airplane {

// int myid;

// char\* model;

// char\* color;

// int engines;

// int capacity;

//public:

// static int ID;

// static int MAX;

// Airplane(const char\* model, const char\* color, const int& e,

// const int& c)

// {

// assert(ID < MAX && "Our garage is full");

// this->myid = ++ID;

// }

// void Show() {

// cout << "Airplane ID : " << this->myid << endl;

// }

// static int getCurrentId() {

// return ID;

// }

//

//};

//int Airplane::ID = 0;

//int Airplane::MAX = 5;

//

//void main() {

//

// Airplane a1("Boend-747", "White", 4, 120);

// Airplane a2("Airbus", "Blue", 4, 200);

// Airplane a3("T-300", "Black", 2, 4);

// Airplane a4("T-300", "Black", 2, 4);

// Airplane a5("T-300", "Black", 2, 4);

// Airplane a6("T-300", "Black", 2, 4);

//

// a1.Show();

// a2.Show();

// a3.Show();

// cout << Airplane::getCurrentId() << endl;

//}