OOP LAB 05

24k-0753

QUESTION 01

#include <iostream>

#include <cstring>

using namespace std;

class Apartment {

int id;

string address;

char\* owner;

public:

Apartment(int i, string add, const char\* own) {

id = i;

address = add;

owner = new char[strlen(own) + 1];

strcpy(owner, own);

}

Apartment(const Apartment& a) {

id = a.id;

address = a.address;

owner = a.owner;

}

void display() {

cout << "ID: " << id << "\nAddress: " << address << "\nOwner: " << owner << endl;

}

~Apartment() {

delete[] owner;

}

};

int main() {

Apartment a1(101, "Street 5, NY", "AliMuhammad");

Apartment a2 = a1;

cout << "Original Apartment:\n";

a1.display();

cout << "\nCopied Apartment:\n";

a2.display();

return 0;

}

QUESTION 02

#include <iostream>

#include <cstring>

using namespace std;

class Student {

int id;

char\* name;

int\* scores;

int numScores;

public:

Student(int i, const char\* n, int sc[], int num) {

id = i;

name = new char[strlen(n) + 1];

strcpy(name, n);

numScores = num;

scores = new int[num];

for (int j = 0; j < num; j++) {

scores[j] = sc[j];

}

}

Student(const Student& s) {

id = s.id;

name = new char[strlen(s.name) + 1];

strcpy(name, s.name);

numScores = s.numScores;

scores = new int[numScores];

for (int j = 0; j < numScores; j++) {

scores[j] = s.scores[j];

}

}

void display() {

cout << "ID: " << id << "\nName: " << name << "\nScores: ";

for (int j = 0; j < numScores; j++) {

cout << scores[j] << " ";

}

cout << endl;

}

~Student() {

delete[] name;

delete[] scores;

}

};

int main() {

int marks[] = {90, 85, 78};

Student s1(1, "Ali", marks, 3);

Student s2 = s1;

cout << "Original Student Record:\n";

s1.display();

cout << "\nCopied Student Record:\n";

s2.display();

return 0;

}

QUESTION 03

#include <iostream>

using namespace std;

class Employee {

public:

string name, designation;

Employee(string n, string d) {

name = n;

designation = d;

}

void display() {

cout << "Employee: " << name << ", Designation: " << designation << endl;

}

};

class Project {

public:

string title, deadline;

Employee\* employees[5];

int empCount;

Project(string t, string d) {

title = t;

deadline = d;

empCount = 0;

}

void addEmployee(Employee\* e) {

if (empCount < 5) {

employees[empCount++] = e;

}

}

void displayDetails() {

cout << "Project: " << title << ", Deadline: " << deadline << endl;

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

employees[i]->display();

}

}

};

int main() {

Employee e1("ALI", "Developer"), e2("AHMED", "Tester");

Project p1("Website", "2025-06-01");

p1.addEmployee(&e1);

p1.addEmployee(&e2);

p1.displayDetails();

return 0;

}

QUESTION 04

#include <iostream>

using namespace std;

class AlarmSystem {

public:

string securityLevel;

AlarmSystem(string level) {

securityLevel = level;

}

void display() {

cout << "Alarm Security Level: " << securityLevel << endl;

}

};

class SmartHome {

public:

AlarmSystem alarm;

SmartHome(string level) : alarm(level) {}

void display() {

cout << "Smart Home Details:\n";

alarm.display();

}

};

int main() {

SmartHome home("High");

home.display();

return 0;

}

QUESTION 05

#include <iostream>

using namespace std;

class Doctor {

public:

string name, specialization;

Doctor(string n, string s) {

name = n;

specialization = s;

}

void display() {

cout << "Doctor: " << name << ", Specialization: " << specialization << endl;

}

};

class Hospital {

public:

string name;

Doctor\* doctors[5];

int docCount;

Hospital(string n) {

name = n;

docCount = 0;

}

void addDoctor(Doctor\* d) {

if (docCount < 5) {

doctors[docCount++] = d;

}

}

void displayDetails() {

cout << "Hospital: " << name << endl;

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

doctors[i]->display();

}

}

};

int main() {

Doctor d1("DR.ALI", "Cardiologist"), d2("DR.FAHAD", "Dermatologist");

Hospital h1("NOOR HOSPITAL");

h1.addDoctor(&d1);

h1.addDoctor(&d2);

h1.displayDetails();

return 0;

}

QUESTION 06

#include <iostream>

using namespace std;

class Level {

public:

int levelNumber;

Level(int n) {

levelNumber = n;

}

void display() {

cout << "Level: " << levelNumber << endl;

}

};

class VideoGame {

public:

string title, genre;

Level\* levels[5];

int levelCount;

VideoGame(string t, string g) {

title = t;

genre = g;

levelCount = 0;

}

void addLevel(Level\* l) {

if (levelCount < 5) {

levels[levelCount++] = l;

}

}

void displayDetails() {

cout << "Game: " << title << ", Genre: " << genre << endl;

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

levels[i]->display();

}

}

};

int main() {

VideoGame game("Fifa 2020", "RPG");

Level l1(1), l2(2);

game.addLevel(&l1);

game.addLevel(&l2);

game.displayDetails();

return 0;

}

QUESTION 07

#include <iostream>

using namespace std;

class Product {

public:

int id;

string name;

int quantity;

Product(int i, string n, int q) {

id = i;

name = n;

quantity = q;

}

void display() {

cout << "ID: " << id << " | Name: " << name << " | Quantity: " << quantity << endl;

}

};

class Warehouse {

public:

Product\* inventory[5];

int count;

Warehouse() {

count = 0;

}

void addProduct(Product\* p) {

if (count < 5) {

inventory[count++] = p;

}

}

void searchByID(int id) {

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

if (inventory[i]->id == id) {

cout << "Product found:\n";

inventory[i]->display();

return;

}

}

cout << "Product not found!\n";

}

void displayAll() {

cout << "Warehouse Inventory:\n";

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

inventory[i]->display();

}

}

};

int main() {

Warehouse w;

Product p1(101, "Laptop", 5), p2(102, "Chair", 10);

w.addProduct(&p1);

w.addProduct(&p2);

cout << "Before sorting:\n";

w.displayAll();

cout << "\nSearching for Product ID 102:\n";

w.searchByID(102);

return 0;

}