Oops Assignment 5

Made by : Arnav Bansal

SID : 23103046

Group : CSE - G1

1. C++ Program to Use Static Member Functions to Check the Number of Objects Created for a Class

#include <bits/stdc++.h>

using namespace std;

class Counter {

static int objectCount; // Static data member to keep track of the number of objects

public:

Counter() {

objectCount++; // Increment count when an object is created

}

static int getObjectCount() {

return objectCount; // Static member function to return object count

}

};

int Counter::objectCount = 0; // Initialize static data member

int main() {

Counter obj1;

Counter obj2;

Counter obj3;

cout << "Number of objects created: " << Counter::getObjectCount() << endl;

return 0;

}



2. C++ Program to Input Today's Date and Print Tomorrow's Date Correctly

#include <bits/stdc++.h>

using namespace std;

class Date {

int day, month, year;

public:

Date(int d, int m, int y) : day(d), month(m), year(y) {}

void printTomorrow() {

static const int daysInMonth[] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

day++;

if (day > daysInMonth[month - 1]) {

day = 1;

month++;

if (month > 12) {

month = 1;

year++;

}

}

cout << "Tomorrow's date is: " << setw(2) << setfill('0') << month << "/"

<< setw(2) << setfill('0') << day << "/" << year << endl;

}

};

int main() {

int d, m, y;

cout << "Enter today's date (mm/dd/yyyy): ";

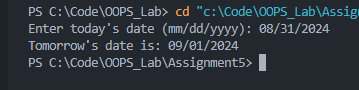
scanf("%d/%d/%d", &m, &d, &y);

Date today(d, m, y);

today.printTomorrow();

return 0;

}



3. C++ Program to Read Any Five Real Numbers and Print the Average Using a Static Member Class

#include <bits/stdc++.h>

using namespace std;

class Average {

static float sum;

static int count;

public:

Average(float num) {

sum += num;

count++;

}

static float getAverage() {

return sum / count;

}

};

float Average::sum = 0;

int Average::count = 0;

int main() {

float num;

cout << "Enter five real numbers: ";

for (int i = 0; i < 5; i++) {

cin >> num;

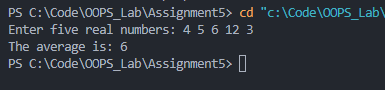
Average obj(num);

}

cout << "The average is: " << Average::getAverage() << endl;

return 0;

}



4. C++ Program to Demonstrate the Use of Parameterized and Copy Constructors

#include <bits/stdc++.h>

using namespace std;

class Box {

int length, breadth, height;

public:

// Parameterized constructor

Box(int l, int b, int h) : length(l), breadth(b), height(h) {}

// Copy constructor

Box(const Box &b) : length(b.length), breadth(b.breadth), height(b.height) {}

void display() {

cout << "Box dimensions (LxBxH): " << length << "x" << breadth << "x" << height << endl;

}

};

int main() {

Box box1(10, 20, 30); // Parameterized constructor

Box box2 = box1; // Copy constructor

cout << "Box1 ";

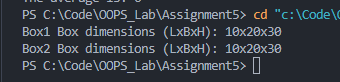
box1.display();

cout << "Box2 ";

box2.display();

return 0;

}



5. C++ Program to Demonstrate the Use of Default Constructors of Employee’s Data

#include <bits/stdc++.h>

using namespace std;

class Employee {

string name;

int id;

float salary;

public:

// Default constructor

Employee() : name("John Doe"), id(0), salary(30000.0f) {}

void display() {

cout << "Employee Name: " << name << endl;

cout << "Employee ID: " << id << endl;

cout << "Employee Salary: " << salary << endl;

}

};

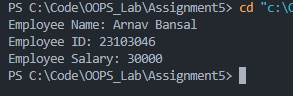
int main() {

Employee emp; // Default constructor will be called

emp.display();

return 0;

}



6. C++ Program to Demonstrate the Use of This Pointer

#include <bits/stdc++.h>

using namespace std;

class Rectangle {

int length, breadth;

public:

Rectangle(int l, int b) : length(l), breadth(b) {}

void setDimensions(int l, int b) {

this->length = l;

this->breadth = b;

}

void displayArea() {

cout << "Area of the rectangle: " << this->length \* this->breadth << endl;

}

};

int main() {

Rectangle rect(10, 20);

rect.displayArea();

rect.setDimensions(15, 25);

rect.displayArea();

return 0;

}

