

# Лабораторна робота №1

**Тема:** Вивчення базових понять класу.

**Мета:** Закріпити базові знання про клас. Навчитись створювати класи засобами мови C++

## ***Завдання 1.***

Створіть клас `Int`, що імітує стандартний тип `int`. Єдине поле цього класу повинно мати тип `int`. Створіть методи, які будуть встановлювати значення поля рівним нулю, ініціалізувати його цілим значенням, виводити значення поля на екран і складати два значення типу `Int`.

Напишіть програму, в якій будуть створені три об'єкти класу `Int`, два з яких будуть ініціалізованими. Додайте два ініціалізованних об'єкта, надайте результат третьому, а потім відобразіть результат на екрані.

## **КОД ПРОГРАМИ**

```
#include <iostream>
using namespace std;

class Int
{
private:
    int field;
public:

    int getField()
    {
        return field;
    }

    void setField(int field)
    {
        this->field = field;
    }

    void setZero()
    {
        this->field = 0;
    }

    void setIntegerValue()
    {
        cout << "Enter the number" << endl;
```

```

        int value;
        cin >> value; cout << endl;

        this->field = value;
    }

    void printToConsole()
    {
        cout << "Field = " << field << endl;
    }

    int addTwoIntObject(Int object1, Int object2)
    {
        return object1.getField() + object2.getField();
    }
};

int main()
{
    Int object1;
    object1.setIntegerValue();

    Int object2;
    object2.setIntegerValue();

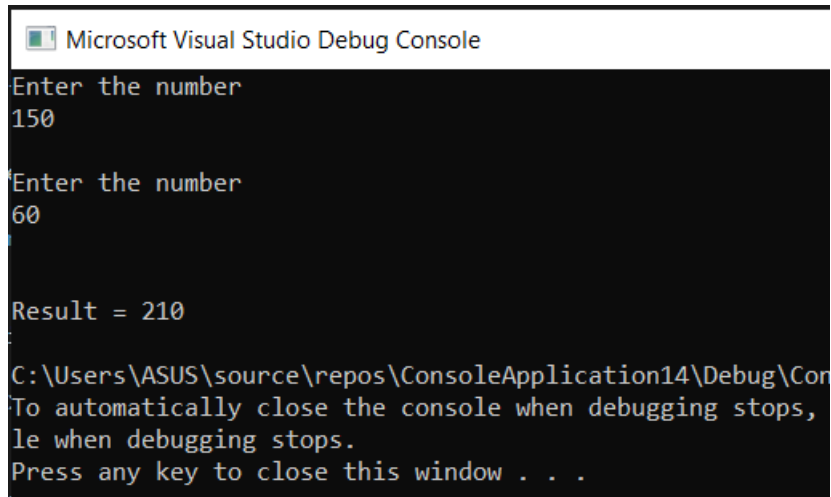
    Int object3;
    object3.setField(object3.addTwoIntObject(object1, object2));

    cout << endl << "Result = " << object3.getField() << endl;

    return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ



```

Microsoft Visual Studio Debug Console

Enter the number
150

Enter the number
60

Result = 210

C:\Users\ASUS\source\repos\ConsoleApplication14\Debug\Con
To automatically close the console when debugging stops,
le when debugging stops.
Press any key to close this window . . .

```

## Завдання 2.

Уявіть пункт для прийому платежів за проїзд по автостраді. Кожна проїжджаюча машина повинна заплатити за проїзд 50 центів, однак частина машин платить за проїзд, а частина проїжджає безкоштовно. У касі ведеться облік числа проїхавших машин і сумарна виручка від плати за проїзд.

Створіть модель такої каси за допомогою класу Kasa. Клас повинен містити два поля. Одне з них, типу unsigned int, призначене для обліку кількості проїхали автомобілів, а друге, що матиме тип double, міститиме сумарну виручку від оплати проїзду. Конструктор повинен ініціалізувати обидва поля нульовими значеннями. Метод payingCar () інкрементує число машин і збільшує на 0,50 сумарну виручку. Інший метод, payCar (), збільшує на одиницю число автомобілів, але залишає без зміни виручку. Метод display () виводить обидва значення на екран. Там, де це можливо, зробіть методи константними.

Створіть програму, яка продемонструє роботу класу. Програма повинна запропонувати користувачеві натиснути одну клавішу для того, щоб зімітувати оплату водієм, і іншу клавішу, щоб зімітувати несумлінного водія. Натискання клавіші Esc повинно привести до видачі поточних значень кількості машин і виручки, і до завершення програми.

## КОД ПРОГРАМИ

```
#include <iostream>
#include <conio.h>
#include <string>
using namespace std;

class Kasa
{
private:
    unsigned int numberCars;
    double revenue;

public:
    Kasa()
    {
        this->numberCars = 0;
        this->revenue = 0;
    }

    void payingCar()
    {
        numberCars++;
        revenue += 0.5;
    }
}
```

```

void nopayCar()
{
    numberCars++;
}

void display()const
{
    cout << endl;

    cout << "Cars = " << numberCars << " " << " Suma = " << revenue << "$" <<
endl;
}

};

int main()
{
    Kasa MainKasa;

    cout << "to simulate the drivers payment, press the letter g" << endl;

    cout << "to imitate a dishonest driver, press the letter b (to exit press ESC)" <<
endl;
    cout << "SWITCH TO THE ENGLISH KEYBOARD!!!" << endl << endl;
    char choice = ' ';
    char g = 'g'; char b = 'b';

    while (choice != 27)
    {
        choice = _getch();
        if (choice == g) {
            MainKasa.payingCar(); cout << (char)choice << " " << endl;
        }
        if (choice == b) {
            MainKasa.nopayCar(); cout << (char)choice << " " << endl;
        }

        if ((int)choice == 27) {
            MainKasa.display();
            break;
        }
    }

    return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ

```
Microsoft Visual Studio Debug Console

to simulate the drivers payment, press the letter g
to imitate a dishonest driver, press the letter b (to exit press ESC)
SWITCH TO THE ENGLISH KEYBOARD!!!

g
g
b
b
g
b
g
b
b
g

Cars = 10  Suma = 2.5$

C:\Users\ASUS\source\repos\ConsoleApplication14\Debug\ConsoleApplicati
To automatically close the console when debugging stops, enable Tools-
le when debugging stops.
Press any key to close this window . . .
```

## Завдання 3.

Створіть клас з ім'ям `time`, що містить три поля типу `int`, призначених для зберігання годин, хвилин і секунд. Один з конструкторів класу повинен ініціалізувати поля нульовими значеннями, а інший конструктор - заданим набором значень. Створіть метод класу, який буде виводити значення полів на екран у форматі `11:59:59`, і метод, складає значення двох об'єктів типу `time`, переданих в якості аргументів.

У функції `main ()` слід створити два ініціалізованих об'єкта (подумайте, чи повинні вони бути константними) і один неініціалізований об'єкт. Потім складіть два ініціалізованих значення, а результат надайте третьому об'єкту і виведіть його значення на екран. Де можливо, зробіть методи константними.

## КОД ПРОГРАМИ

```
#include <iostream>
using namespace std;

class time
{
private:
    int hour;
    int minute;
```

```

        int seconds;

public:
    int getHour() { return hour; }
    int getMinute() { return minute; }
    int getSeconds() { return seconds; }

    void setHour(int hour) { this->hour = hour; }
    void setMinute(int minute) { this->minute = minute; }
    void setSeconds(int seconds) { this->seconds = seconds; }

    time()
    {
        this->hour = 0;
        this->minute = 0;
        this->seconds = 0;
    }

    time(int hour, int minute, int seconds)
    {
        this->hour = hour;
        this->minute = minute;
        this->seconds = seconds;
    }

    void printTime()
    {
        cout << "HH:MM:SS--> " << hour << ":" << minute << ":" << seconds << endl <<
endl;
    }

    time addTime(time time1, time time2)
    {
        int sumaHour = time1.getHour() + time2.getHour();
        int sumaMinute = time1.getMinute() + time2.getMinute();
        int sumaSeconds = time1.getSeconds() + time2.getSeconds();

        int Seconds = sumaHour * 60 * 60 + sumaMinute * 60 + sumaSeconds;

        int numberHours = 0, numberMinute = 0, numberSeconds = 0;

        while ((Seconds - 3600) >= 0)
        {
            Seconds -= 3600;
            numberHours++;
        }

        while ((Seconds - 60) >= 0)
        {
            Seconds -= 60;
            numberMinute++;
        }

        numberSeconds = Seconds;

        time time3;
        time3.setHour(numberHours); time3.setMinute(numberMinute);
        time3.setSeconds(numberSeconds);
    }

```

```

        return time3;
    }
};

int main()
{
    int hour, minute, seconds;
    cout << "hour = "; cin >> hour; cout << endl;
    cout << "minute = "; cin >> minute; cout << endl;
    cout << "seconds = "; cin >> seconds; cout << endl;

    time time1(hour, minute, seconds);
    time1.printTime();
    cout << "-----" << endl;

    cout << "hour = "; cin >> hour; cout << endl;
    cout << "minute = "; cin >> minute; cout << endl;
    cout << "seconds = "; cin >> seconds; cout << endl;

    time time2(hour, minute, seconds);
    time2.printTime();
    cout << "-----" << endl;

    time time3;
    time3 = time().addTime(time1, time2);
    cout << endl << "TIME1+TIME2--> " << time3.getHour() << ":" << time3.getMinute() <<
":" << time3.getSeconds() << endl;

    return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ

```
Select Microsoft Visual Studio Debug Console

hour = 12
minute = 17
seconds = 59
HH:MM:SS--> 12:17:59
-----
hour = 14
minute = 60
seconds = 12
HH:MM:SS--> 14:60:12
-----
TIME1+TIME2--> 27:18:11

C:\Users\ASUS\source\repos\ConsoleApplication14\Debug\Cor
To automatically close the console when debugging stops,
le when debugging stops.
Press any key to close this window . . .
```

## Варіант(3)

### Завдання 4.

\* У кожному завданні потрібно реалізувати клас. У програмі обов'язково повинні бути продемонстровані різні способи створення об'єктів і масивів об'єктів. Програма повинна демонструвати використання всіх функцій і методів.

У всіх завданнях обов'язково повинні бути присутні:

- метод ініціалізації `Init( )`, метод повинен контролювати значення аргументів на коректність;
- введення з клавіатури `Read( )`;
- виведення на екран `Display( )`.



Атрибут first – ціле додатнє число, чисельник; атрибут second – ціле додатнє число, знаменник. Реалізувати метод ipart() – виділення цілої частини дробу fist/second. Метод повинен перевіряти нерівність знаменника нулю.

## КОД ПРОГРАМИ

```
#include <iostream>
using namespace std;

class Number
{
private:
    int first;
    int second;

public:
    int getFirst() { return first; }
    int getSecond() { return second; }
    void setFirst(int first) { this->first = first; }
    void setSecond(int second) { this->second = second; }

    void Read()
    {
        cout << "-----" << endl << "first = ";
        int first; cin >> first;

        cout << endl << "second = ";
        int second; cin >> second;

        if (Init(first, second) == true) {
            this->first = first;
            this->second = second;
            Display();
            cout << "RESULT = " << ipart() << endl;
        }
        else
        {
            cout << "WRONG PARAMETERS !!! " << endl;
        }
    }

    bool Init(int first, int second)
    {
        if (first > 0 && second > 0)
            return true;
        else
            return false;
    }

    void Display()
    {
        cout << endl << "fist/second --> " << this->first << "/" << this->second <<
endl;
    }
}
```

```

    int ipart()
    {
        if (second != 0)
            return this->first / this->second;
    }
};

int main()
{
    cout << "-----" <<
endl;
    Number SomeNumber;
    SomeNumber.Read();

    cout << "-----" <<
endl;
    cout << "How many objects do you want to create" << endl;
    int numObject = 0;
    cin >> numObject;

    Number *ArrSomeNumber = new Number[numObject];

    for (int i = 0; i < numObject; i++)
    {
        ArrSomeNumber[i].Read();
    }

    return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ

```
Microsoft Visual Studio Debug Console

first/second --> 12/5
RESULT = 2
-----
How many objects do you want to create
3
-----
first = 27
second = 6
first/second --> 27/6
RESULT = 4
-----
first = 30
second = 0
WRONG PARAMETERS !!!
-----
first = 9
second = 7
first/second --> 9/7
RESULT = 1

C:\Users\ASUS\source\repos\ConsoleApplication14\Debug\ConsoleApplication14.exe
To automatically close the console when debugging stops, enable Tools->
Press any key to close this window . . .
```

**ВИСНОВОК:** Закріпив базові знання про клас.  
Навчився створювати класи засобами мови C++