

Laboratorium 1 – Klasy i obiekty

Zad. 1

Napisz klasę `Pies` zawierającą publiczne pole `imie` (np. `char*`). Następnie w programie głównym stwórz dwa obiekty typu `Pies` (np. `wilczur`, `jamnik`) i nadaj im imiona (np. „`Szarik`”, itp.). Wypisz imiona psów na ekran.

Zad. 2

Do klasy `Pies` dopisz metodę `Spaceruj()`. Metoda ma za zadanie wyświetlenie komunikatu, że obiekt typu `Pies` wykonuje daną czynność. Następnie w programie głównym stwórz dwa obiekty typu `Pies` (np. `wilczur`, `jamnik`), nadaj im imiona i wykonaj dla nich metodę `Spaceruj()`.

Zad.3

Zmodyfikuj wyświetlany komunikat tak, aby zawierał imię danego psa, czyli żeby był postaci: „`Szarik` teraz jest na spacerze”. Trzeba wtedy wykorzystać w komunikacie zmienną `imie`. Zauważ, że nie ma potrzeby przekazywania zmiennej `imie` do metody `Spaceruj()` jako parametr.

Zad. 4

Dopisz drugą metodę `Jedz()` również wyświetlającą odpowiedni komunikat. Wywołaj ją w programie głównym dla obu psów.

Zad. 5

Dodaj pole `barwa` typu wyliczeniowego (enum `Kolor`), a następnie wklej polecenie zmieniające kolor czcionki `SetConsoleTextAttribute(hk, barwa)` w każdej z metod `Spaceruj()` i `Jedz()` przed wypisaniem komunikatu na ekran. Wywołaj te metody kilkakrotnie na przemian. Szkielet programu:

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

enum Kolor { jasno_szary=7, ciemno_szary, czerwony=12, zolty=14, bialy };
HANDLE hk = GetStdHandle(STD_OUTPUT_HANDLE); // uchwyt

class Pies
{
public:
    Kolor barwa;
    // tutaj definiujemy pole imie oraz metody Jedz() i Spaceruj()
};

int main(void)
{
    Pies wilczur; // tworzymy obiekt wilczur typu Pies
    wilczur.imie="Szarik";
    wilczur.barwa=czerwony;
    // tutaj stworzymy drugi obiekt typu Pies, nadamy mu imie i barwe,
    // i jeszcze wywołamy metody Jedz() i Spaceruj() dla obu czworonogów
    SetConsoleTextAttribute(hk, bialy); // wracamy do koloru białego
    system("PAUSE");
    return 0;
}
```