

```

// W01.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include <iostream>
using namespace std;

//deklaracje klasy
class MY_CLASS
{
    int i;           //zmienna prywatna
public:
    int k;           //zmienna publiczna
    int get_i();      //funkcja publiczna - zwraca wartosc zmiennej prywatnej
    void set_i(int ii); //funkcje publiczna - ustawia wartosc zmiennej prywatnej
    void disp();
};

//implementacje klasy MY_CLASS
int MY_CLASS::get_i()
{
    return i;
}

void MY_CLASS::set_i(int ii)
{
    i = ii;
}

void MY_CLASS::disp()
{
    //wewnatrz klasy dostep do danych prywatnych i danych publicznych jest zezwolony
    cout << "obiekt MY_CLASS:" << endl;
    cout << "private  i = " << i << endl;
    cout << "public   k = " << k << endl;
}

int main()
{
    //deklaracje obiektow klasy MY_CLASS

```

```
MY_CLASS obl;
```

```
int ii;
```

```
cout << "wprowadz i" << "\n";
```

```
cin >> ii;
```

```
//dostep bezposredni do elementow prywatnych klasy jest wzbroniony
```

```
//obl.i = ii;    //blad kompilacji
```

```
//to jest hermetyzacja skladowych prywatnych: dostep do nich jest mozliwy
```

```
//tylko przez metode klasy get_i(), set_i(...);
```

```
obl.set_i(ii);
```

```
//bezposredni dostep do skladowych publicznych - OK
```

```
obl.k = 10;
```

```
//nastepna instrukcja zawiera blad - obl.i - !OK
```

```
//cout << "obiekt obl:\t" << " i = " << obl.i << "\t k = " << obl.k << endl;
```

```
//dostep do skladowej prywatnej obiektu klasy jest mozliwy przez metode get_i()
```

```
cout << "obiekt obl:\t" << " i = " << obl.get_i() << "\t k = " << obl.k << endl;
```

```
//wywolujemy metode disp() - inny sposob wydruku
```

```
obl.disp();
```

```
//zatrzymujemy czarne okienko - console application
```

```
system("pause");
```

```
return 0;
```

```
}
```