

# Programowanie obiektowe

---

## Zapis i odczyt pliku

Zapis i odczyt pliku działa na podstawie tych samych definicji klas co obiekty `cin` i `cout`.

Aby **zapis pliku** był możliwy należy utworzyć obiekt klasy **ofstream**, a następnie skojarzyć go z konkretnym plikiem (nazwą pliku).

```
ofstream zapis;
...
zapis.open("notatka.txt");
```

W odniesieniu do obiektów klasy `ofstream` sposób otwarcia plików jest określany jako

```
ios :: out (ios_base:: out )
```

Aby **odczyt pliku** był możliwy należy utworzyć obiekt klasy **ifstream**, a następnie skojarzyć go z konkretnym plikiem (nazwą pliku).

```
ifstream czytanie;
...
czytanie.open("notatka.txt");
```

W odniesieniu do obiektów klasy `ifstream` sposób otwarcia plików jest określany przez domniemanie jako

```
ios :: in (ios_base:: in )
```

Kojarzenie obiektu klasy **fstream** z plikiem `notatka.txt` przebiega według schematu:

```
fstream obiekt;
obiekt.open('polozenie_i_nazwa_pliku' , int tryb , int dostep);
lub
fstream obiekt.open('polozenie_i_nazwa_pliku' , ios :: openmpde TRYB);
```

`polozenie_i_nazwa_pliku` – może zostać zastąpione ścieżką bezwzględną do pliku np:

```
C:\cwiczenia\notatka.txt
```

Sposób otwarcia pliku określa stała – `tryb`, a prawa dostępu zamienna `dostep`

---

## Przykłady:

**Tworzenie i zapis danych do pliku - obiekt klasy ofstream :**

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    ofstream zapis; //Tworzę obiekt zapis należący do klasy ofstream
    cout << "Zapisuje do pliku" << endl;
    zapis.open("notatka.txt"); // kojarzę obiekt zapis z plikiem notatka.txt
    zapis << "To jest test zapisu do pliku"; //zapisuje tekst do pliku
    zapis.close(); // zamykam połączenie wyjściowe z plikiem
    cout << "Zapisalem";
    system("PAUSE");
}
```

```

    return EXIT_SUCCESS;
}

```

### Odczyt danych z pliku - obiekt klasy **ifstream** :

```

#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    ifstream czytanie; //Tworzę obiekt czytanie należący do klasy ifstream
    cout << "\n Teraz odczytam wcześniejszy plik " << endl;
    cout << "\n-----\n";
    czytanie.open("notatka.txt"); // kojarzę obiekt czytanie z plikiem
notatka.txt
    char pobierz;
    if(!czytanie.is_open()) return 1; //zapisuje tekst do pliku
    czytanie >> noskipws >> pobierz;
    while (czytanie.good() ) {
        cout << pobierz;
        czytanie >> pobierz;
    };
    czytanie.close(); // zamykam połączenie wejściowe z plikiem
    cout << "\n-----";
    cout << "\n Przeczytałem";
    system("PAUSE");
    return EXIT_SUCCESS;
}

```

Funkcje klasy ios:

Stałe klasy ios	Funkcja
<b>ios :: in</b>	Czyta plik, który musi istnieć
<b>ios :: ate</b>	Po otwarciu pliku ustawi a wskaźnik na koniec pliku
<b>ios :: out</b>	Tworzy lub nadpisuje istniejący plik i otwiera plik do zapisu
<b>ios :: app</b>	Otwiera plik w celu dopisywania, zachowując zawartość
<b>ios :: trunc</b>	Tworzy nowy lub nadpisuje istniejący plik
<b>ios :: binary</b>	W systemie MS-DOS i Windows wskazuje, że dane pliku mają być traktowane jako binarne, a nie tekstowe