



Pliki

Czym jest obsługa plików w Javie?

Obsługa plików w Javie to mechanizm umożliwiający programom:

- zapisywanie danych na dysku
- odczytywanie wcześniej zapisanych informacji
- przechowywanie danych w sposób trwały

Co umożliwia programowi?

Dzięki plikom program może:

- zapamiętać dane użytkownika
- zapisać wyniki obliczeń
- przechowywać konfigurację
- tworzyć logi i raporty

Jak Java „widzi” pliki?

Java nie operuje bezpośrednio na plikach jako całości.
Każdy plik jest traktowany jako strumień danych.

Strumień:

- to uporządkowany przepływ danych, dane są czytane lub zapisywane krok po kroku

Podział strumieni w Javie

1. Ze względu na kierunek
2. Ze względu na typ danych

Ze względu na kierunek

- strumień wejściowy – odczyt danych z pliku
- strumień wyjściowy – zapis danych do pliku

Ze względu na typ danych

- strumień bajtowe – operują na bajtach
(pliki binarne: obrazy, PDF, ZIP)
- strumień znakowe – operują na znakach
(pliki tekstowe: TXT, CSV)

Pakiety do obsługi plików

Java udostępnia dwa główne podejścia:

1. `java.io`
2. `java.nio.file`

java.io

- starsze, klasyczne API
- oparte na strumieniach
- bardziej rozbudowane
- nadal bardzo często używane

java.nio.file

- nowoczesne API (od Java 7)
- prostsze i bardziej czytelne
- zalecane w nowych aplikacjach

Klasa File (java.io.File)

Klasa File:

- reprezentuje plik lub katalog
- nie służy do zapisu ani odczytu danych
- pozwala zarządzać plikami

Za pomocą klasy File można:

- sprawdzić czy plik istnieje
- sprawdzić typ (plik / katalog)
- pobrać ścieżkę i nazwę
- sprawdzić rozmiar
- utworzyć katalog
- usunąć plik

Obsługa plików w java.io

Pakiet java.io opiera się na:

- strumieniach wejścia i wyjścia
- rozdzieleniu operacji na:
 - odczyt
 - zapis
- ręcznym zarządzaniu zasobami

Jest to podejście:

- bardzo elastyczne
- dokładne
- ale wymagające więcej kodu

Odczyt tekstu w java.io

1. Odczyt znak po znaku
2. Odczyt linia po linii

Odczyt znak po znaku

Charakterystyka działania:

- tekst jest czytany pojedynczymi znakami
- daje dużą kontrolę nad danymi
- jest wolniejszy
- rzadko używany w praktyce

Zastosowanie:

- analiza znaków
- specjalne algorytmy przetwarzania tekstu

Odczyt linia po linii

Charakterystyka działania:

- czyta cały wiersz tekstu naraz
- dużo szybszy dzięki buforowi
- najczęściej stosowany sposób odczytu tekstu

Zastosowanie:

- pliki tekstowe
- konfiguracje
- logi
- dane użytkownika

Zapis tekstu w java.io

Zapis tekstu odbywa się za pomocą strumieni znakowych.

Możliwości zapisu:

- nadpisanie pliku
- dopisywanie do istniejącego pliku
- zapisywanie wielu linii

Często wykorzystywany do:

- logów aplikacji
- raportów
- zapisu danych użytkownika

Obsługa wyjątków przy pracy z plikami

Podczas operacji na plikach mogą wystąpić błędy:

- plik nie istnieje
- brak uprawnień
- błąd zapisu lub odczytu

Java wymaga obsługi takich sytuacji za pomocą:

- bloków try-catch
- wyjątków typu IOException

Nowoczesne podejście – java.nio.file

API java.nio.file wprowadza:

- prostsze metody
- czytelniejszy kod
- operacje na całych plikach

Nie wymaga bezpośredniej pracy na strumieniach w większości przypadków.

Kluczowe klasy `java.nio.file`

- `Path` – reprezentuje ścieżkę do pliku
- `Paths` – tworzy obiekty `Path`
- `Files` – zawiera metody do operacji na plikach

Operacje dostępne w Files

Za pomocą klasy Files można:

- odczytać cały plik tekstowy
- zapisać dane do pliku
- skopiować plik
- przenieść plik
- usunąć plik
- sprawdzić czy plik istnieje

Różnice między java.io a java.nio.file

Cecha	java.io	java.nio.file
Ilość kodu	duża	mała
Czytelność	średnia	wysoka
Wydajność	dobra	bardzo dobra
Nowoczesność	starsze	aktualny standard

Słownik

- Plik - Zbiór danych zapisany na nośniku danych (np. dysku). Umożliwia trwałe przechowywanie informacji.
- Strumień (Stream) - Uporządkowany przepływ danych między programem a źródłem lub miejscem zapisu (np. plikiem).
- Strumień wejściowy - Strumień służący do odczytu danych z pliku do programu.
- Strumień wyjściowy - Strumień służący do zapisu danych z programu do pliku.
- Strumień bajtowy - Strumień operujący na bajtach. Używany do plików binarnych (obrazy, PDF, ZIP).
- Strumień znakowy - Strumień operujący na znakach (tekst). Używany do plików tekstowych.

- Bufor (Buffer) - Tymczasowy obszar pamięci, który przyspiesza odczyt i zapis danych poprzez operowanie większymi porcjami danych.
- `BufferedReader` / `BufferedWriter` - Klasy wykorzystujące bufor do szybkiego odczytu i zapisu danych tekstowych.
- Wyjątek (Exception) - Sytuacja błędna, która może wystąpić podczas działania programu (np. brak pliku).
- `IOException` - Najczęstszy wyjątek przy pracy z plikami – informuje o problemach z odczytem lub zapisem.
- `Path` - Obiekt reprezentujący ścieżkę do pliku w nowoczesnym API `java.nio.file`.
- `Files` - Klasa zawierająca metody do operacji na plikach.