

ТЕХНОЛОГИИ JAVA, ЛЕКЦИИ

1 Введение

Базовые технологии: ввод-вывод, collections framework, потоки, рефлексия, разные инструменты. Потом большой блок многопоточного программирования. На базе этого блока мы разберёмся с сетевыми приложениями. Затем приступим к блоку "разное", принимаются адекватные предложения по поводу доп. лекций.

2 Ввод-вывод

Потоки различаются по направлению: ввод или вывод. И по содержимому: байтовые и символьные. У Java двухбайтные символы, этим различается символьный от байтового.

	Байтовый	Символьный
Ввод	InputStream	Reader
Вывод	OutputStream	Writer

3 Исключительные ситуации

IOException — это корень иерархии исключений ввода-вывода.

EOFException — попытка чтения после конца потока.

FileNotFoundException — файл не найден.

UnsupportedEncodingException — неизвестная кодировка. Есть кодировки, которые не поддерживаются некоторыми JDK. Это плохой дизайн. Во-первых, как это обрабатывать — это вопрос. Во-вторых, проблеме программа узнает, когда она пытается открыть файл в данной кодировке, например, но этих мест в программах может быть очень много.

Проверяемые нужно обрабатывать, не нужно обрабатывать непроверяемые.

4 Потоки ввода

В сигнатурах тип T означает байт для байтовых потоков и символьный ип для символьных потоков соответственно.

- `int read()` — чтение элемента.
- `read(T[] array), read(T[] array, offset, length)` — чтение элементов в массив.
- `byte[] readNBytes(n)` — чтение массива.

Метод `read` не гарантирует, что он прочтёт ровно столько данных, сколько вы попросили. Файл может кончиться. Тогда `read` даст вам хвост и вернет -1.

Некоторые потоки поддерживают дополнительные операции:

- `skip(n)` — пропуск до n элементов.
- `available()` — сколько элементов доступно.

Закрытие потока.

Интерфейс `Closeable` extends ...

5 Пометки и возвраты

`markSupported()` — проверка поддержки.

Основные операции:

- `write(v)` — запись элемента.
- `write(T[] array)` — запись массива элементов
- `write(T[] array, offset, length)` — запись фрагмента.

Запись буфера:

- Интерфейс Flushable
- Метод flush

Успешное закрытие потока означает, что все данные были записаны.

6 Файловый ввод-вывод

Кодировка символьных потоков по умолчанию зависит от системы. Почти всегда неуказание конкретной кодировки будет ошибкой.

<Обсуждение кода>

7 Фильтрующие потоки