

# FileInputStream, FileOutputStream, ZipOutputStream, ZipInputStream

Java Collections  
1 уровень, 5 лекция

ОТКРЫТА

— Привет, Амиго!

— Привет, Элли!

— Сегодня мы будем изучать потоки ввода-вывода.

— Да, я уже там все знаю. Это делается с помощью классов `FileInputStream` и `FileOutputStream`.

— Да, а какие методы есть у этих классов, знаешь?

— Конечно, вот методы `FileInputStream`:

Метод	Описание
<code>int read()</code>	Читает один байт и возвращает его.
<code>int read(byte b[])</code>	Читает массив байт, возвращает количество.
<code>int read(byte b[], int off, int len)</code>	Читает из потока массив байт.
<code>long skip(long n)</code>	Пропускает n байт, используется при чтении из потока.
<code>int available()</code>	Возвращает количество байт, которое еще можно прочитать из потока.
<code>void close()</code>	Закрывает поток.

— Все верно! А методы `FileOutputStream`?

— Вот:

Метод	Описание
<code>void write(int b)</code>	Записывает один байт в поток.
<code>void write(byte b[])</code>	Записывает массив байт в поток.
<code>void write(byte b[], int off, int len)</code>	Записывает массив байт в поток.
<code>void close()</code>	Закрывает поток.

— Амиго, ты меня удивляешь!

— А то!

— Отлично, тогда дам тебе два новых класса `ZipInputStream` и `ZipOutputStream`.



— Zip? Это как zip-архив?

— Именно. Эти потоки предназначены для работы с файлами, которые являются zip-архивами. С помощью них можно что-то читать прямо из zip-фала или писать в него!

— Ух ты! Как интересно. Но ведь в zip-архиве может быть не один файл, как же они работают?

— А для этого есть еще один специальный класс – **ZipEntry**. Он представляет файл, хрянящийся в архиве. Из **ZipInputStream** можно читать только объекты типа **ZipEntry**, а в **ZipOutputStream** можно писать только объекты **ZipEntry**. А вот сами **ZipEntry** как раз можно читать и писать как обычные файлы.

— А можно пример, как это работает?

— Конечно, вот пример, где мы создаем архив и кладем в него файл:

Код	
1	// создаем архив
2	FileOutputStream zipFile = new FileOutputStream("c:/archive.zip");
3	ZipOutputStream zip = new ZipOutputStream(zipFile);
4	
5	//кладем в него ZipEntry - «архивный объект»
6	zip.putNextEntry(new ZipEntry("document.txt"));
7	
8	//копируем файл «document-for-archive.txt» в архив под именем «document.txt»
9	File file = new File("c:/document-for-archive.txt");
10	Files.copy(file.toPath(), zip);
11	
12	// закрываем архив
13	zip.close();

— Как интересно. А разархивировать файлы так же легко?

— Ага. Вот тебе краткое описание методов классов **ZipEntry**, **ZipInputStream** и **ZipOutputStream**

**ZipInputStream** – это поток, поэтому все **ZipEntry** могут быть прочитаны только последовательно. Вот его методы:

Метод	Описание
ZipEntry getNextEntry()	Возвращает объект, описывающий следующий ZipEntry (пропускает все байты текущего entry).
void closeEntry()	Закрывает чтение текущего ZipEntry (пропускает все байты текущего entry).
int available()	Возвращает 1, если есть доступные ZipEntry, иначе 0.

<code>long skip(long n)</code>	Пропускает n байт при чтении из потока.
<code>void close()</code>	Закрывает поток.

— Не очень понятно.

— Лучше всего — это представить, что ты читаешь текстовый файл, а **ZipEntry** – это строки в нем. Можно читать как данные текущей строки – текущий **ZipEntry**, так и прыгать по строкам – **getNextEntry**, **closeEntry**.

— Вроде понятно, хотя не уверен, что правильно.

— ZipOutputStream и его методы:

Метод	Описание
<code>void setComment(String comment)</code>	Устанавливает комментарий к архиву.
<code>void setMethod(int method)</code>	Указывает метод (тип) сжатия.
<code>void setLevel(int level)</code>	Указывает степень сжатия. Чем сильнее, тем медленнее.
<code>void putNextEntry(ZipEntry e)</code>	Добавляет новый ZipEntry.
<code>void closeEntry()</code>	Закрывает текущий ZipEntry.
<code>void write(byte[] b, int off, int len)</code>	Пишет набор байт в текущий ZipEntry.
<code>void close()</code>	Закрывает поток.

— А мы же в примере выше почти ничего этого не использовали.

— Это необязательные методы. Можно не указывать степень и метод сжатия, тогда будут использованы настройки по умолчанию.

— Гм. Тоже не плохо. А ZipEntry?

— Ага. В ZipEntry дополнительно хранится только служебная информация:

Метод	Описание
<code>String getName()</code>	Внутреннее имя файла.
<code>long getTime(), setTime(long)</code>	Время последней модификации entry.
<code>long getCrc(), setCrc(long)</code>	Контрольная сумма.
<code>long getSize(), setSize(long)</code>	Размер до компрессии.
<code>int getMethod(), setMethod(int)</code>	Метод сжатия.
<code>long get/setCompressedSize()</code>	Размер после архивации.
<code>boolean isDirectory()</code>	Является ли entry директорией.

Выглядит необычно и интересно, что не может не поработать!

НАЧАТЬ ОБУЧЕНИЕ

 +97 

Комментарии (101)

популярные

новые

старые

JavaCoder

Введите текст комментария

Ilia lenskii

Уровень 22, Санкт-Петербург

3 июля, 13:18

...

Язык мой – друг мой.

"Мой" – это глагол.

Ответить

 +2 

Zhanbek

Программист в АО «АЛСЕКО»

28 апреля, 15:24

...

Скорее всего в комментарий к коду закралась опечатка: "//копируем файл «document-for-archive.txt» в архив под именем «**document.txt**»".  
На мой взгляд должно быть "**c:/archive.zip**"

Ответить

 0 

Jh-007

Уровень 47

10 мая, 11:02

...

В архиве может быть несколько документов с разными именами

Ответить

 0 

Zlata

Уровень 49, Олыштын, Польша

21 апреля, 18:20

...

+++

Ответить

 0 

Shekerbek

Backend Developer

17 марта, 06:10

...

thanks

Ответить

 0 

KOTNinja

Уровень 32, Санкт-Петербург, Россия

14 марта, 10:58

...

*Сегодня мы будем изучать потоки ввода-вывода.* Черт подери, мы что-нибудь кроме потоков ввода-вывода изучать будем? Первый блок курса - целый уровень для потоков, второй блок - еще больше потоков (то же самое по сути) и вот, **COLLECTIONS** - и тоже поток на потоке и потоком погоняет. Вы бы лучше что-нибудь толковое и малопонятное рассказали.

Ответить

 +1 

тимур

Уровень 33, Москва, Russian Federation

10 апреля, 21:16

...

ответ от разработчиков: "Мы понимаем Ваше негодование, чтобы Вам стало легче, почитайте про потоки ввода-вывода по этой ссылке:"

Ответить

 +1 

Сергей

Уровень 40, Санкт-Петербург, Russian Federation

9 февраля, 14:46

...

Очередной рерайт документации Oracle дополненный бессмысленным диалогом ("воды") . Автор лекции выдохся на тексте диалога? Документация Oracle не самый лучший источник для рерайта =). Для рерайта можно взять тексты других авторов, раскрывающие особенности работы методов, ну это если креатив закончился на диалогах и лень самим придумывать действительно ценный текст для читателя. Если же цель диалоги , то можно ещё упростить содержимое лекции до названия методов, ссылки на документацию Oracle (тут конечно если останутся силы, дать ссылку на соответствующую страницу в документации) и ссылку на поисковую систему . Даже диалоги тогда будут не нужны Экономия!

Такие лекции(а их много на javarush) кладезь для троллинга.  
ЗЫ. Для тех, кто любит писать "поиск - это часть обучения" . помню об этом. Спасибо

НАЧАТЬ ОБУЧЕНИЕ

EgorQA Automation Engineer

22 декабря 2021, 17:05

Ребят, очепятку поправьте:  
"А для этого есть еще один специальный класс – ZipEntry. Он представляет файл, хрянящийся в архиве."  
Хорошо хоть не хрянящийся)

Ответить

+3

LuneFoxинженер по сопровождению в BIFIT

EXPERT

17 декабря 2021, 13:49

1

`long getSize(), setSize(long)`

Размер до компрессии.

Какой смысл устанавливать размер до компрессии? Мы же можем только получить размер файла, но не задать, пол. Иначе бы я установил размер всем своим файлам на ПК в 1КБ и радовался жизни, но увы, магии не существует.

Ответить

0

ArsУровень 41

15 ноября 2021, 13:13

Понятно, что повторение - мать учения. И что это фишка джавараша: сначала дать задачи, потом объяснить. Но тогда эту фишку надо нормально реализовать, блин. Пускай Амиго тогда говорит, что уже де работал с этими ZipStream, и даже целую задачу сделал. Но уже подзабыл немного что и как. А некоторые моменты не понял. И рад вернуться к этой теме и подтянуть её.  
В таком изложении было бы совсем другое дело.

Ответить

0

Лиза ВоренувкинаУровень 43, Кривой Рог, Ukraine

22 сентября 2021, 22:14

— Язык мой – враг мой.  
Так хорошо сказано ..... Соглашусь .

Ответить

+1

Показать еще комментарии

ОБУЧЕНИЕ

- Курсы программирования
- Курс Java
- Помощь по задачам
- Подписки
- Задачи-игры

СООБЩЕСТВО

- Пользователи
- Статьи
- Форум
- Чат
- Истории успеха
- Активности

КОМПАНИЯ

- О нас
- Контакты
- Отзывы
- FAQ
- Поддержка



JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

Русский

НАЧАТЬ ОБУЧЕНИЕ



"Программистами не рождаются" © 2022 JavaRush