

Like



Share



Subscribe

AnhNguyenNgoc

ProIT4All



Giới thiệu

Luồng nhập xuất ký tự



Fan page: <http://facebook.com/Proit4All>



Luồng ký tự (Character stream)

- ❑ Luồng ký tự cho phép đọc/ghi dữ liệu ký tự theo dạng mã Unicode
 - ❖ Lớp trừu tượng Reader: đọc dữ liệu ký tự
 - ❖ Lớp trừu tượng Writer: Ghi dữ liệu ký tự





Tổ chức phân cấp lớp của Reader và Writer

Character Stream

16 bits carrier - Unicode

Reader

- **BufferedReader**
Used for Buffered Input Stream
- **CharArrayReader**
Used for reading from an array
- **StringReader**
Used for read from a string
- **FileReader** - Used for reading from a File
- **PipedReader** - Input pipe
- **InputStreamReader** - translates bytes to character
- **FilterReader** - filtered reader
- **LineNumberReader** - used to count lines

Writer

- **BufferedWriter**
Used for Buffered Output Stream
- **CharArrayWriter**
Used for writing into an array
- **StringWriter**
Used for write into a string
- **FileWriter** - Used for writing into a File
- **PipedWriter** - Output pipe
- **OutputStreamWriter** - characters to bytes
- **FilterWriter** - filtered writer
- **PrintStream** - Contains **print()** and **println()**

read() and **write()** both are key methods of byte stream



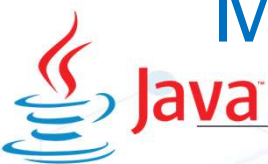
Unicode

☐ Unicode, chính thức là chuẩn Unicode

- ❖ Là chuẩn công nghệ thông tin
- ❖ Để mã hóa, biểu diễn và xử lý văn bản nhất quán được thể hiện trong hầu hết các hệ thống chữ viết trên thế giới

☐ Có nhiều dạng mã hóa khác nhau của Unicode. Nhưng có 2 dạng phổ biến:

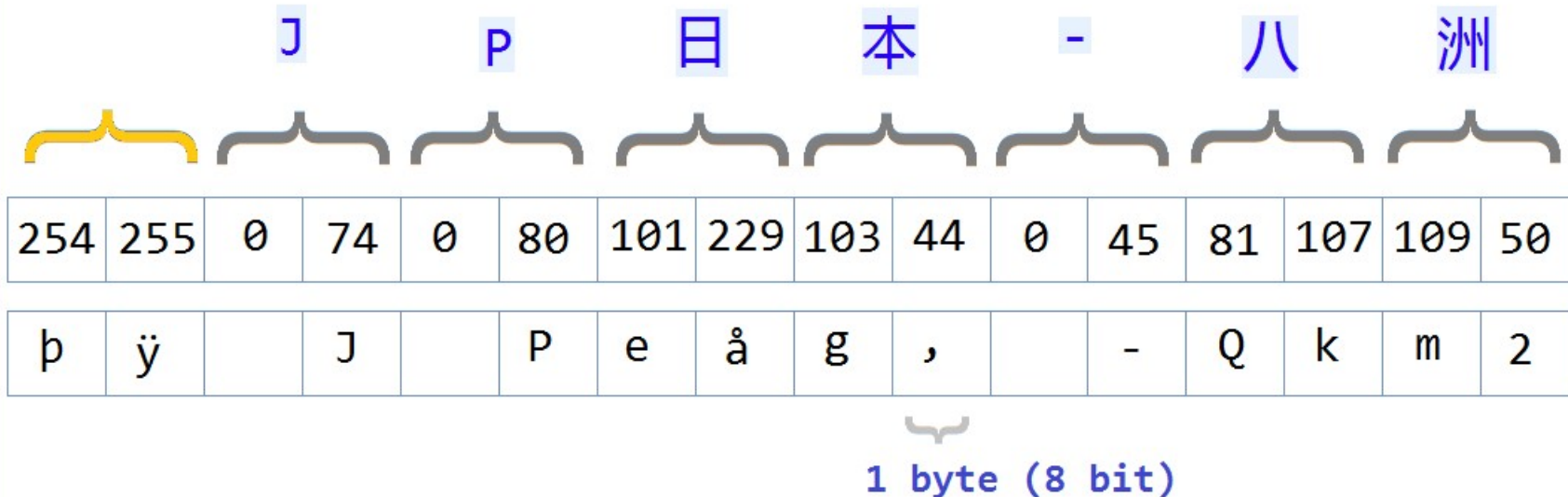
- ❖ ASCII-compatible UTF-8 (Dạng mã hóa tương thích ASCII UTF-8),
- ❖ ASCII-incompatible UTF-16 (Dạng mã hóa không tương thích ASCII UTF-16)



Mã hóa UTF-16

- Biểu diễn mỗi ký tự bao gồm cố định 2 byte

```
String text = "JP日本-八洲";
```





Mã hóa UTF-8

❑ Các ký tự ASCII được biểu diễn bằng 1 byte

❖ Ví dụ:

- Cần 1 byte biểu diễn ký tự 'J' với mã (74)
- Cần 1 byte biểu diễn ký tự 'P' với mã (80)

❑ UTF-8 có thể cần 1, 2, hoặc 3 byte để biểu diễn ký tự

- ❖ Đọc byte đầu tiên, nếu ≤ 127 , thì đó là ký tự ASCII
- ❖ Ngược lại, nếu > 127 , thì đọc tiếp byte thứ 2, và kiểm tra có biểu diễn ký tự nào trong bảng mã UTF-8 không
- ❖ Nếu 2 byte đầu tiên không biểu diễn được ký tự, đọc tiếp byte thứ 3 và kết hợp 3 byte thành 1 ký tự



Mã hóa UTF-8

UTF-8

```
String text = "JP日本-八洲";
```

J P 日 本 - 八 洲

74	80	230	151	165	230	156	172	45	229	133	171	230	180	178
----	----	-----	-----	-----	-----	-----	-----	----	-----	-----	-----	-----	-----	-----

J	P	æ		¥	æ		¬	-	å		«	æ	´	2
---	---	---	--	---	---	--	---	---	---	--	---	---	---	---

1 byte (8 bit)



Phương thức của lớp Reader

❑ `int read()`

❖ Đọc ký tự tiếp theo từ luồng nhập.

❑ `int read(char[] cbuffer)`

❖ Đọc đoạn ký tự từ luồng nhập và lưu trữ trong mảng byte: cbuffer.

❑ `int read(char[] cbuf, int off, int len)`

❖ Đọc các ký tự từ luồng nhập và lưu trữ thành một phần của mảng.

❑ `int read(CharBuffer target)`

❖ Đọc các ký tự vào bộ đệm ký tự.



Phương thức của lớp Reader

☐ `String readLine()`

❖ Đọc dòng ký tự

☐ `boolean ready()`

❖ Cho biết luồng đã sẵn sàng để đọc hay chưa.

☐ `void close()`

❖ Đóng luồng nhập và giải phóng mọi tài nguyên được kết nối với luồng



Phương thức của lớp Writer

❑ void flush()

❖ Đổ dữ liệu trong bộ đệm ra luồng

❑ void write(char[] cbuf)

❖ Ghi mảng ký tự ra luồng xuất

❑ void write(char[] cbuf, int off, int len)

❖ Ghi một phần của mảng ký tự ra luồng xuất

❑ void write(int c)

❖ Ghi ký tự đơn.

❑ void write(String str)

❖ Ghi chuỗi.



Phương thức của lớp Writer

❑ `void write(String str, int off, int len)`

❖ Ghi một phần của chuỗi.

❑ `Writer append(char c)`

❖ Nối thêm ký tự cho writer.

❑ `Writer append(CharSequence csq)`

❖ Nối thêm chuỗi ký tự cho writer

❑ `Writer append(CharSequence csq, int start, int end)`

❖ Nối thêm một phần của chuỗi cho writer

❑ `void close()`

❖ Đóng luồng xuất và giải phóng mọi tài nguyên được kết nối với luồng.

Like



Share



Subscribe

AnhNguyenNgoc

ProIT4All



Giới thiệu

Đọc file với Reader và FileReader



Fan page: <http://facebook.com/Proit4All>