



# CÁC LUỒNG VÀO RA DỮ LIỆU

---

- Khái niệm về luồng dữ liệu
- Luồng và tệp
- Lớp File
- Truy cập tệp tuần tự
- Truy cập tệp ngẫu nhiên



# CÁC LUỒNG VÀO RA DỮ LIỆU

---

## Luồng dữ liệu (data streams)

- Chương trình Java nhận và gửi dữ liệu thông qua các đối tượng là các thực thể thuộc một kiểu luồng dữ liệu nào đó
- Luồng (stream) là một dòng dữ liệu đến từ một nguồn (source) hoặc đi đến một đích (sink)
- Nguồn và đích có thể là tệp (file), bộ nhớ, một tiến trình (process), hay thiết bị (bàn phím, màn hình, ...)

## Luồng byte và char

- Luồng byte: thao tác theo đơn vị byte
  - InputStream
  - OutputStream
- Luồng char: thao tác với ký tự
  - Reader
  - Writer

## InputStream

- `int read()`
- `int read(byte buf[])`
- `int read(byte buf[], int offset, int length)`
- `void close()`

## OutputStream

- `int write(int c)`
- `int write(byte buf[])`
- `int write(byte buf[], int offset, int length)`
- `void close()`
- `void flush()`

## Reader

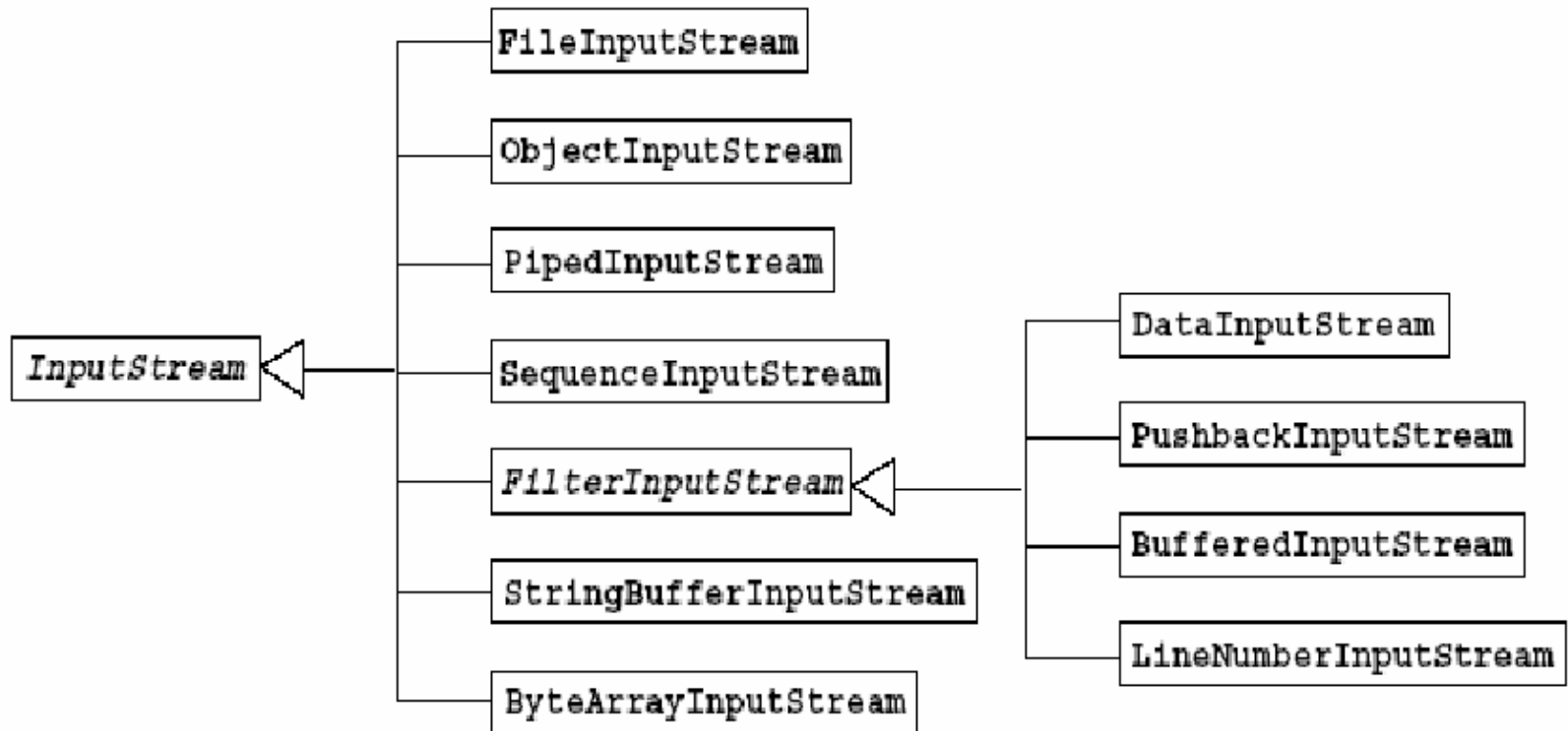
- `int read()`
- `int read(char buf[])`
- `int read(char buf[], int offset, int length)`
- `void close()`

## Writer

- `int write(int c)`
- `int write(char buf[])`
- `int write(char buf[], int offset, int length)`
- `void close()`
- `void flush()`

# CÁC LUỒNG VÀO RA DỮ LIỆU

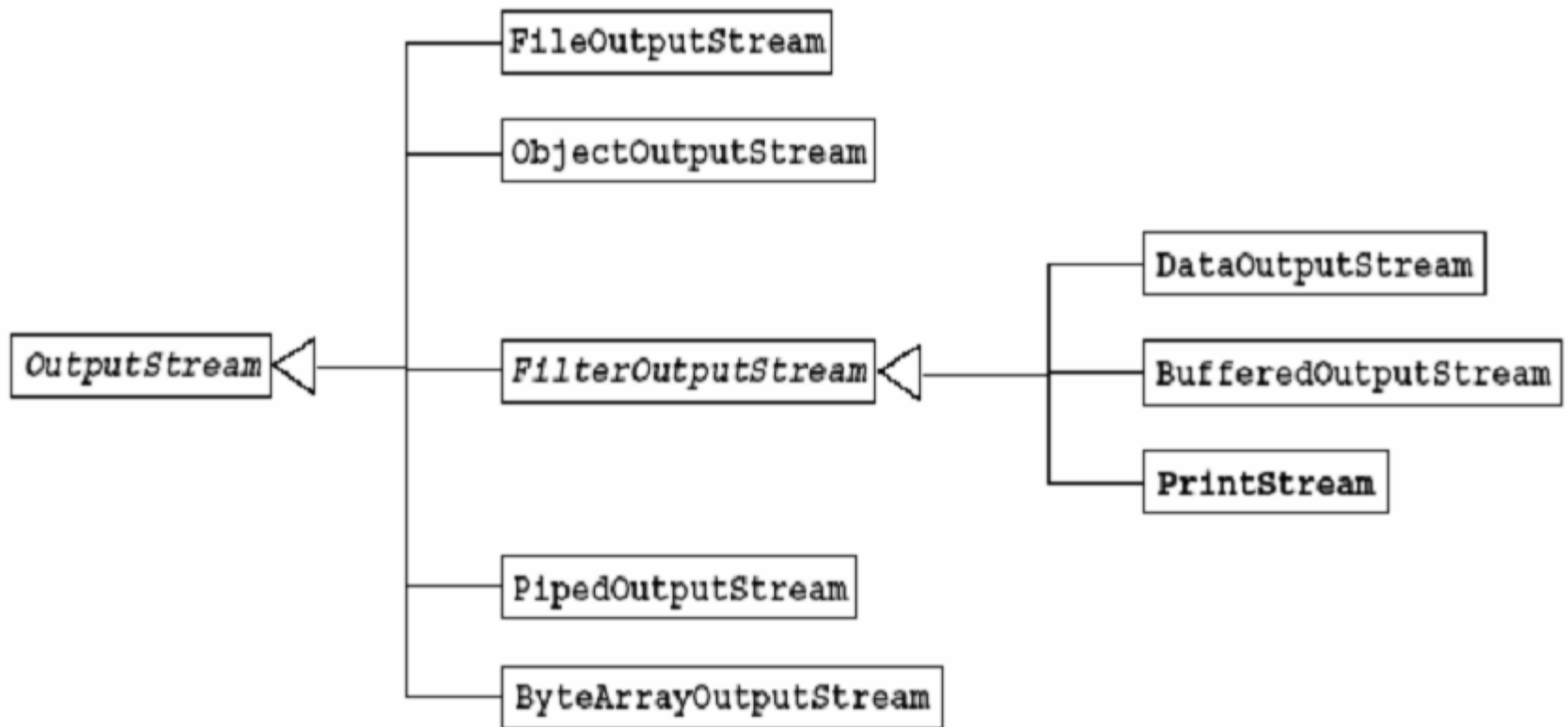
## Phả hệ của InputStream





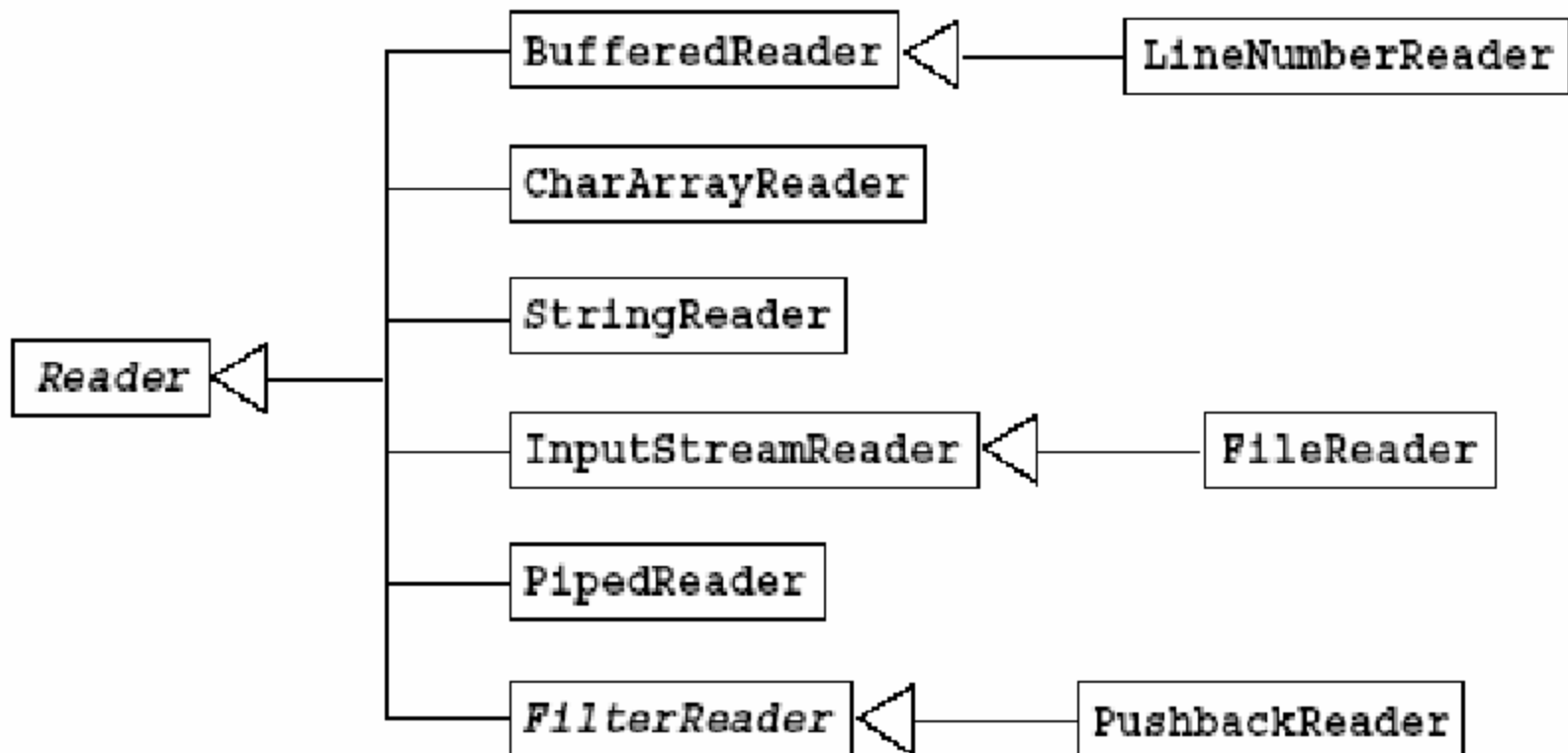
# CÁC LUỒNG VÀO RA DỮ LIỆU

## Phả hệ của OutputStream



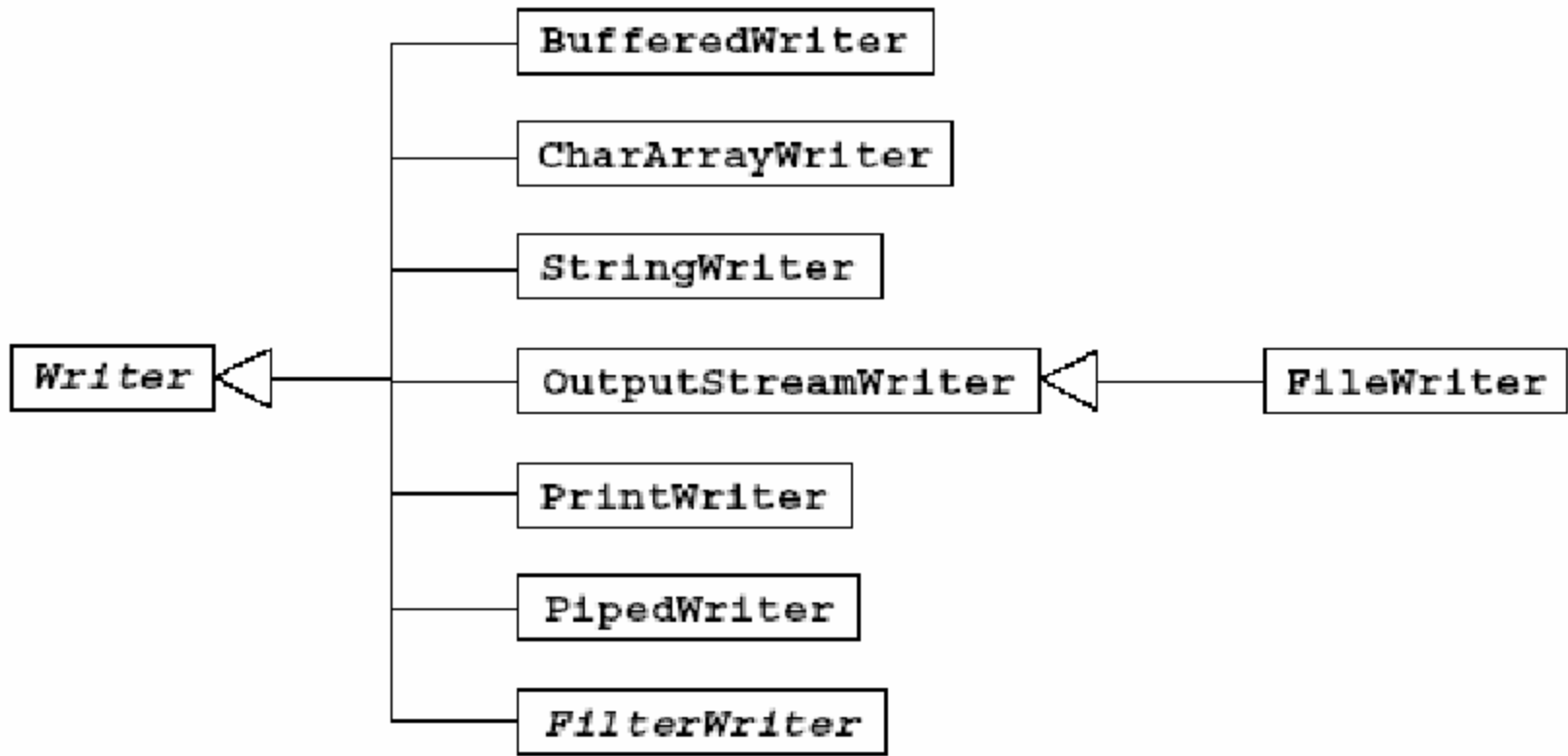
# CÁC LUỒNG VÀO RA DỮ LIỆU

## Phả hệ của Reader



# CÁC LUỒNG VÀO RA DỮ LIỆU

## Phả hệ của Writer



## Đối tượng vào / ra

- Để nhập hoặc xuất dữ liệu, chúng ta phải tạo ra đối tượng vào hoặc ra
- Đối tượng vào hoặc ra thuộc kiểu luồng tương ứng và phải được gắn với một nguồn dữ liệu hoặc một đích tiêu thụ dữ liệu



# CÁC LUỒNG VÀO RA DỮ LIỆU

---

## Sử dụng bộ đệm

- Bộ đệm là một kỹ thuật để tăng tính hiệu quả của thao tác vào / ra
  - đọc và ghi dữ liệu theo khối
  - giảm số lần thao tác với thiết bị
- Thay vì ghi trực tiếp tới thiết bị thì chương trình ghi lên bộ đệm
  - khi bộ đệm đầy thì dữ liệu được ghi ra thiết bị theo khối
  - có thể ghi vào thời điểm bất kỳ bằng phương thức flush()
- Thay vì đọc trực tiếp từ thiết bị thì chương trình đọc từ bộ đệm
  - khi bộ đệm rỗng thì dữ liệu được đọc theo khối từ thiết bị



# CÁC LUỒNG VÀO RA DỮ LIỆU

---

## Nhập xuất qua thiết bị chuẩn

Console I/O

- `System.out` cho phép in ra luồng ra chuẩn
  - là đối tượng của lớp `PrintStream`
- `System.err` cho phép in ra luồng thông báo lỗi chuẩn
  - là đối tượng của lớp `PrintStream`
- `System.in` cho phép đọc vào từ thiết bị vào chuẩn
  - là đối tượng của lớp `InputStream`

## Đọc dữ liệu từ luồng vào chuẩn

- System.in không sử dụng được trực tiếp
- Chúng ta muốn đọc một dòng ký tự
  1. tạo đối tượng luồng ký tự (InputStreamReader)
  2. tạo đối tượng luồng có bộ đệm (BufferedReader)



# CÁC LUỒNG VÀO RA DỮ LIỆU

---

Ví dụ:

```
InputStreamReader reader = new InputStreamReader(System.in);  
BufferedReader in = new BufferedReader(reader);  
---  
  
String s;  
try {  
    s = in.readLine();  
}  
catch (IOException e) {...}
```





# BÀI TẬP

---