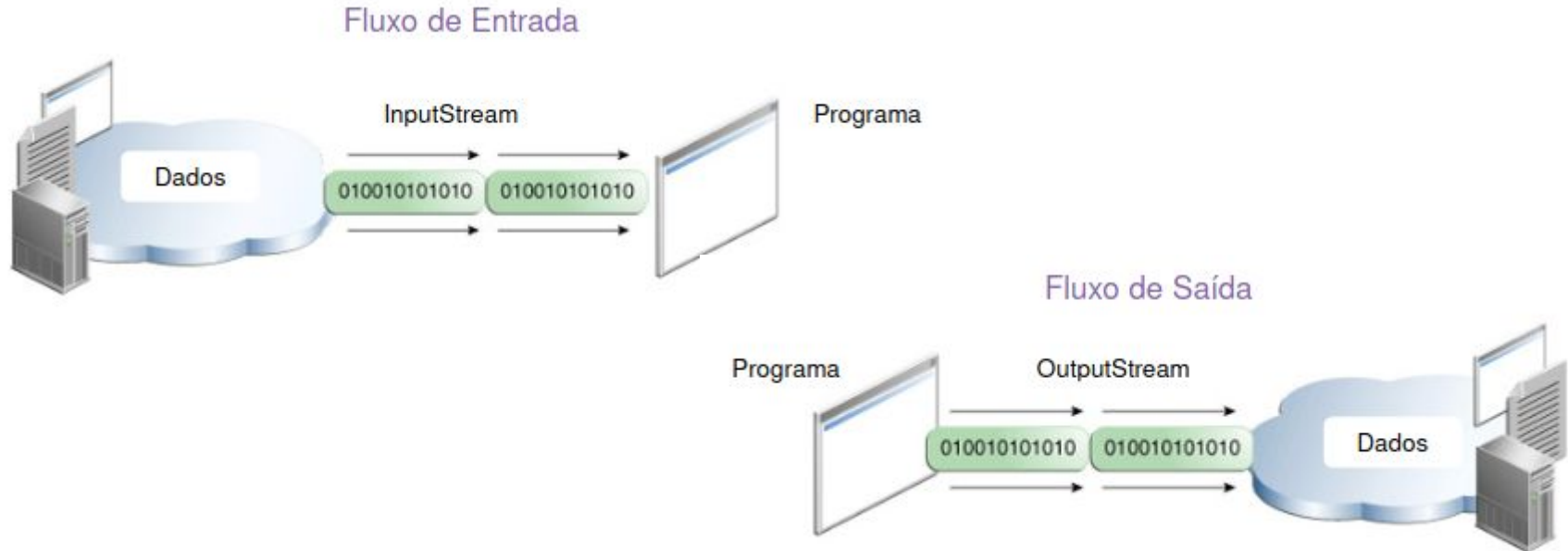


# Arquivos

# Fluxo de entrada e saída

I/O Streams representam uma entrada de dados ou uma saída de dados. Fluxos de dados podem ter arquivos como origem ou destino.



# Fluxo de Byte

Leem e escrevem dados tendo o byte (8 bits) como unidade mínima de informação.

São tratados por subclasses de:

- `InputStream`, para leitura temos os metodos: `read()`, `read(byte[])`, `skip(long)`, `close()`, etc
- `OutputStream`, para escreite temos os metodos: `write(int)`, `write(byte[])`, `flush()`, `close()`, etc

Após leitura/escrita, o cursor avança dentro do arquivo. O fluxo de dados é sequencial. `InputStream` e `OutputStream` são classes **abstratas**.

# Fluxo de Byte em Arquivos

As classes `FileInputStream` e `FileOutputStream` são especializações de `InputStream` e `OutputStream` que lidam com fluxo de dados. Para manipularmos arquivos, vamos utilizar as classes que estendem `InputStream` e `OutputStream`, pois essas classes são abstratas e uma classe abstrata não pode ser instanciada.

# Fluxo de Caracteres em Arquivo Binário

As classes `DataInputStream` e `DataOutputStream` são especializações de `FilterInputStream` e `FilterOutputStream`.

Essas classes lidam com fluxos modificados de informações recebidas/enviadas através de um fluxo de dados existente. Utilizaremos as classes filhas para modificar o fluxo de um `InputStream/OutputStream` que retornam bytes para conseguirmos ler/escrever caracteres.

# Fluxo de Caracteres em Arquivo Texto

Existem duas classes que fazem a ponte entre fluxos de caracteres e fluxos de byte **InputStreamReader** e **OutputStreamWriter**. Essas classes são estendidas por **FileReader** e **FileWriter**.

Para lermos o texto de um fluxo de dados, vamos utilizar a classe **BufferedReader**. Esta classe lê o texto armazenando os caracteres em um buffer para fornecer uma leitura eficiente.

Para escrevermos um conjunto de caracteres em um arquivo, vamos utilizar a classe **PrintWriter**. Esta classe imprime representações formatadas de objetos em um fluxo de saída de texto.

# Exceções Típicas

- **IOException**

Superclasse de toda classe de exceção relacionada a entrada e saída através de arquivos

- **FileNotFoundException**

Disparada quando tenta-se criar um objeto de fluxo para uma fonte que não pode ser localizada

- **EOFException**

Disparada quando o final de arquivo é atingido inesperadamente enquanto os dados são lidos através de um fluxo de entrada