Pesquisa sobre Files

Antonio Filho TPSI 0619 25-11-2019

1.FileInputStream e FileOutputStream:

Compreendendo a classe FileInputStream

O FileInputStream é uma classe de fluxo de entrada de bytes que fornece métodos para ler bytes de um arquivo. Podemos criar uma instância dessa classe fornecendo um nome de arquivo ou caminho, usando esses dois construtores:

• FileInputStream (arquivo de arquivo)

• FileInputStream (nome da sequência)

E a lista a seguir descreve os principais métodos implementados pela classe FileInputStream :

• int available () : retorna uma estimativa do número de bytes restantes que podem ser lidos.

• int read () : lê um byte de dados, retorna o byte como um valor inteiro. Retorne -1 se o final do arquivo for atingido.

• int read (byte []) : lê um pedaço de bytes para a matriz de bytes especificada, até o tamanho da matriz. Este método retorna -1 se não houver mais dados ou se o final do arquivo for atingido.

• int read (byte [], int offset, int length) : lê até o comprimento bytes de dados do fluxo de entrada.

• long skip (long n) : pula e descarta n bytes de dados do fluxo de entrada. Este método retorna o número real de bytes ignorados.

• FileChannel getChannel () : retorna o objeto FileChannel exclusivoassociado a este fluxo de entrada de arquivo. O FileChannelpode ser usado para manipulação avançada de arquivos (Nova IO).

• void close () : fecha este fluxo de entrada de arquivo e libera todos os recursos do sistema associados ao fluxo.

Compreendendo as classes FileOutputStream

O FileOutputStream é uma classe de fluxo de saída de bytes que fornece métodos para gravar bytes em um arquivo. Podemos criar uma instância dessa classe fornecendo um nome de arquivo ou caminho e / ou especificar para substituir ou anexar a um arquivo existente, usando os seguintes construtores:

• FileOutputStream (arquivo de arquivo)

• FileOutputStream (arquivo, anexo booleano) : se o anexo for verdadeiro , os bytes serão gravados no final de um arquivo existente e não no início.

• FileOutputStream (nome da string)

• FileOutputStream (nome da string, anexo booleano)

E a lista a seguir descreve os principais métodos implementados pela classe FileOutputStream :

• void write (int) : grava o byte especificado no fluxo de saída.

• void write (byte []) : grava a matriz especificada de bytes no fluxo de saída.

• void write (byte [], int offset, int length) : escrevebytes decomprimentoda matriz de bytes especificada, começando nooffsetno fluxo de saída.

• FileChannel getChannel () : retorna oobjetoFileChannelexclusivoassociado a este fluxo de saída do arquivo. OFileChannelpode ser usado para manipulação avançada de arquivos (Nova IO).

• void close () : fecha este fluxo de saída do arquivo e libera todos os recursos do sistema associados ao fluxo.

Quase métodos lançam IOException, então lembre-se de manipular ou declarar lançá-lo em seu código.

2. BufferedInputStream e BufferedOutputStream

A classe BufferedInputStream é usada para fornecer buffer ao inputStream. Em vez de ler um byte de cada vez, ele pode ler um grande bloco de dados. Melhora o desempenho da leitura.

A classe BufferedOutputStream usa um buffer interno para armazenar os dados primeiro. Em vez de gravar um byte por vez no arquivo de destino, os dados armazenados em um buffer serão gravados por vez. Não é necessário fazer uma chamada do sistema para cada byte gravado, portanto, acelere a gravação.

Isso geralmente é muito mais rápido, pode ser usado para acesso ao disco e grande quantidade de dados. Para adicionar buffer ao outputStream, basta adicioná-lo através do BufferedOutputStream.

3. DataInputStream e DataOutputStream

A DataInputStream classe lê tipos de dados Java primitivos de um fluxo de entrada subjacente de maneira independente da máquina. Enquanto a DataOutputStream classe grava tipos de dados Java primitivos em um fluxo de saída de maneira portátil.

4.ObjectInputStream e ObjetcOutputStream

ObjectInputStream e ObjectOutputStream. Elas são responsáveis por inserir e recuperar objetos serializados do stream.

Um ObjectInputStream desserializa dados e objetos primitivos escritos anteriormente usando um ObjectOutputStream.

5.Reader e Writer

Reader - É uma classe abstrata sendo igual à InputStream, é voltada para a manipulação de caracteres. Nessa classe não é possível criar objetos diretos dela. Para criar objetos é necessário instanciar uma das subclasses concretas para ter o acesso à funcionalidade especificada por Reader.

Writer - É uma classe abstrata que escreve stream de caracteres. Os únicos métodos que uma subclasse writer deve implementar são: write(char [], int, int), flush () e close ().

6.FileReader e FileWriter

FileReader e FileWritersão classes de conveniência para leitura e gravação de arquivos de caracteres, respectivamente. Ambos FileReader e FileWriter classes usam codificação padrão do sistema.