

9.5 Character Stream Classes

Character streams can be used to read and write 16-bit Unicode characters. Like byte streams, there are two kinds of character stream classes, namely, reader stream classes and writer stream classes.

9.5.1 Reader Stream Classes:

Reader stream classes that are used to read characters include a super class known as Reader and a number of subclasses for supporting various input-related functions.

Reader stream classes are functionally very similar to the input stream classes, except input streams use bytes as their fundamental unit of information, while reader streams use characters.

Methods	Description
Abstract int read() throws IOException	Reads a character.
Int read(char b[])throws IOException	Reads character into b .
Int read(char b[], int n ,int len) throws IOException	Reads n characters into array b starting from n th byte till the len.
Void skip(n) throws IOException	Skips over n characters from stream.
Void mark(int n) throws IOException	This method is used to mark current position in stream.
Void reset()throws IOException	Goes back to the the beginning of the stream.
Void close()throws IOException	Closes the stream.
Boolean ready()throws IOException	Determines if the stream is ready to be run.

9.5.2 Writer Stream Classes:

- Like output stream classes, the writer stream classes are designed to perform all output operations on files.
- Only difference is that while output stream classes are designed to write bytes, the writer stream are designed to write character.
- The Writer class is an abstract class which acts as a base class for all the other writer stream classes.
- This base class provides support for all output operations by defining methods that are identical to those in Output stream class.

Methods	Description
Abstract write(char c) throws IOException	Writes a character to output stream.
Abstract write (char []b) throws IOException	Writes all characters in the array b.
Abstract write(String s) throws IOException	Writes string.
Abstract write(char c[],int n,int len) throws IOException	Writes m character from array b starting from n th character. The no. of characters will be equal to the len.
Void close()throws IOException	Closes the output stream and releases any resources associated with the stream.
Void flush()throws IOException	Flushes the output stream. The buffered data is written to OutputStream.