CSCI 2120:

Software Design & Development II

UNIT3: I/O management

io api

CharacterStream Classes

Overview

1. Introduction
2. Types of CharacterStream classes in Java

---

3. Reader Stream Classes
4. Hierarchy of Reader Stream Classes
5. Constructor of Reader Class
6. Reader Subclasses
7. Reader Methods in Java

---

8. Writer Stream Classes
9. Writer class declaration
10. Hierarchy of Writer class
11. Constructor of Writer Class
12. Writer Subclasses
13. Writer Methods in Java

Introduction

In the previous slides, we learned that the ByteStream classes provide sufficient functionality to handle 1 byte (i.e. 8 bits) of any type of I/O operation. But it is incompatible to work directly with 16-bit Unicode characters. To overcome this problem, Java added the CharacterStream classes to support characters.

CharacterStream classes in Java are used to read and write 16-bit Unicode characters. In other words, character stream classes are mainly used to read characters from the source and write them to the destination. They can perform operations based on characters, char arrays, and Strings.

Note:
Unicode is a 2-byte, 16-bit character set having 65,536 (i.e. 2^16) different possible characters. Only about 40,000 characters are used in practice, the rest are reserved for future purposes. It can handle most of the world’s living languages.

Types of CharacterStream classes in Java

Like byte stream classes, character stream classes also contain two kinds of classes. They are named as:

- Reader Stream classes
- Writer Stream classes

Let’s understand in detail reader stream classes and writer stream classes one by one.

Reader Stream Classes

Reader stream classes are used for reading characters from files.

Reader is an abstract superclass for all other subclasses such as BufferedReader, StringReader, CharArrayReader, etc. It implements the Closeable, AutoCloseable, and Readable interfaces.

The general syntax for the Reader class is as follows:

```
public abstract class Reader
       extends Object
       implements Readable, Closeable
```

Hierarchy of Reader Stream Classes

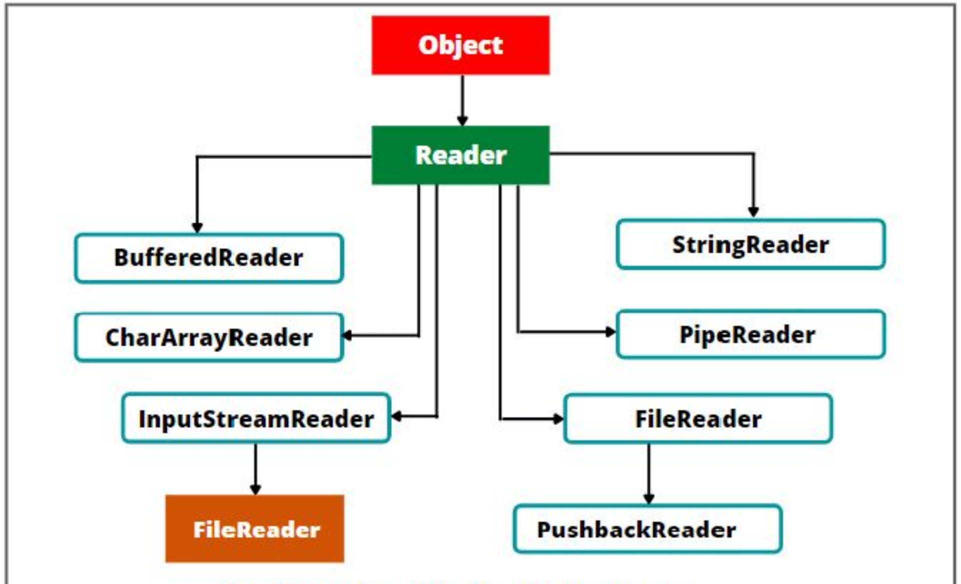

Fig: Hierarchy of Reader Stream Classes

The classes belonging to `Reader` stream classes are very similar functionality to Input Stream classes. The only difference is that Input Stream uses bytes, whereas `Reader` Stream classes use characters.

In fact, both byte and character stream classes use the same methods. Therefore, reader classes can perform all the operations implemented by input stream classes.

Reader Subclasses

`Reader` class in Java defines the following constructors with protected access modifiers. They are:

Constructor of Reader Class

Reader class in Java defines the following constructors with protected access modifiers:

a) protected Reader():
This form of constructor creates a new character-stream reader whose critical sections will synchronize on the reader itself.

b) protected Reader(Object lock):
This form of constructor creates a new character-stream reader whose critical sections will synchronize on the specified object.

Reader Subclasses

Reader Subclass	Description
BufferedReader	This class is used to read characters from the buffered input character stream.
CharArrayReader	This class is used to read characters from the char array or character array.
FileReader	This class is used to read characters (or contents) from a file.
FilterReader	This class is used to read characters from the underlying character input stream.
InputStreamReader	This class is used to translate (or convert) bytes to characters.
PipeReader	This class is used to read characters from the connected piped output stream.
StringReader	This class is used to read characters from a string.
PushBackReader	This class allows one or more characters to be returned to the input stream.

Reader Methods in Java

The Reader class defines the following methods that are as follows:

Reader Methods in Java

Method	Description
int read()	This method returns an integer representation of the next character present in the invoking input stream. It returns -1 when the end of the input is encountered.
int read(char buffer[ ])	This method is used to read up to buffer.length characters from the specified buffer. It returns the actual number of characters successfully read. It returns -1 when the end of the input is encountered.
int read(char buffer[ ], int loc, int numChars)	This method is used to read up to numChars characters from the buffer starting at the specified location. It returns the number of characters successfully read. –1 is returned when the end of the input is encountered.
void mark(int numChars)	This method is used to mark the current point in the input stream that will remain until numChars characters are read.

Reader Methods in Java

Method	Description
void reset()	The reset() method is used to reset the input pointer to the preceding set mark.
long skip(long numChars)	The skip() method is used to skip the specified numChars characters from the input stream and returns the number of characters actually skipped.
boolean ready()	This method returns a true value if the next request of input is ready (i.e. not wait). Otherwise, it returns false.
void close()	This method is used to close the input stream. If the program attempts to read the input further, it generates IOException.
boolean markSupported( )	This method returns boolean value true if mark( )/reset( ) are supported on this stream. All the methods provided by the Reader class (except for markSupported( )) will throw an IOException on error conditions.

Writer Stream Classes

`Writer` stream classes are used to write characters to a file. In other words, They are used to perform all output operations on files.

Writer stream classes are similar to output stream classes with only one difference that output stream classes use bytes to write while writer stream classes use characters to write.

In fact, both output stream and writer stream classes use the same methods.

Writer class declaration

`Writer` class is an abstract class that acts as a superclass for all the other writer stream classes. It implements the `Appendable`, `Closeable,` and `Flushable` interfaces.

The general syntax for Writer class is given below:

```
public abstract class Writer
       extends Object
       implements Appendable, Closeable, Flushable
```

Hierarchy of Writer class

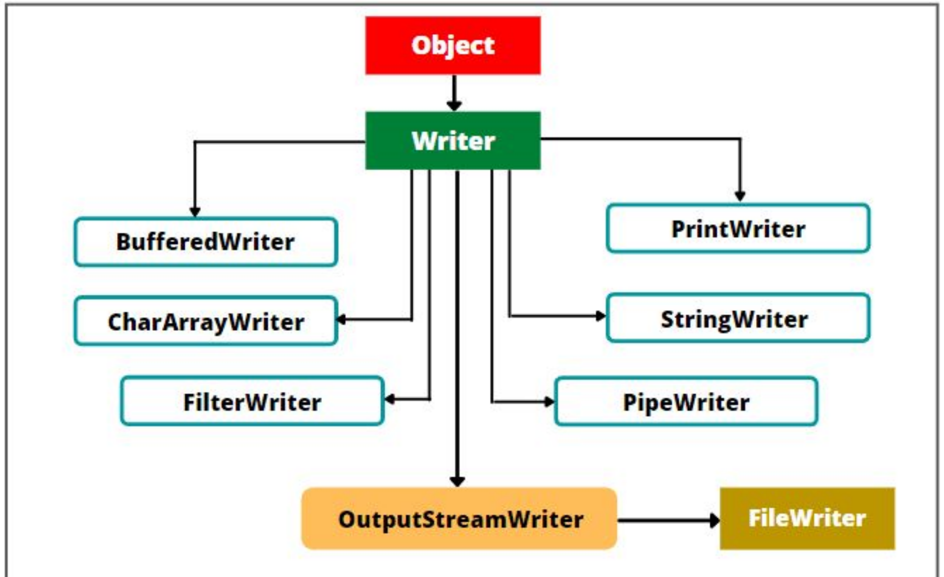

Fig: Hierarchy of Writer Stream Classes

The `Writer` class is designed to write 16-bit Unicode characters to the output stream. Since the `Writer` class is an abstract class, it cannot be instantiated.

Therefore, the subclasses of the `Writer` class are used to write the characters onto the output stream.

The hierarchy diagram of subclasses of the `Writer` class is shown in the below figure.

Constructor of Writer Class

Writer class in Java defines the following constructors with protected access modifiers that are as follows:

1. protected Writer(): This form of constructor constructs a new character-stream writer whose critical sections will synchronize on the writer itself.

2. protected Writer(Object lock): This form of constructor constructs a new character-stream writer whose critical sections will synchronize on the given object.

Writer Subclasses

Let’s understand the subclasses of writer class in a brief description.

Writer Subclasses

Writer Subclass	Description
BufferedWriter	This class is used to write characters to the buffered output character stream.
FileWriter	This output stream class writes characters to the file.
CharArrayWriter	This output stream class writes the characters to the character array.
OutputStreamWriter	This output stream class translates or converts from bytes to characters.
PipedWriter	This class writes the characters to the piped output stream.
StringWriter	This output stream class writes the characters to the string.
PrintWriter	This output stream class contains print() and println().
FilterWriter	This class is used to write characters on the underlying character output stream.

Writer Methods in Java

To write the characters to the output stream, Writer stream class in Java defines the following methods that are as follows:

Note:

All the methods defined by writer stream class throw an IOException if any errors occur.

Writer Methods in Java

Method	Description
void write()	The write() method is used to write the data to the invoking output stream.
void write(int ch)	This method is used to write a single character to the invoking output stream.
void write(char buffer[ ])	This method is used to write a complete array of characters to the invoking output stream.
void write(char buffer [ ], int loc, int numChars)	This method is used to write a subrange of numChars characters from the character array, starting at the specified location to the output stream.
void write(String str)	This method is used to write str to the invoking output stream.

Writer Methods in Java

Method	Description
void write(String str, int loc, int numChars)	This method writes a subrange of numChars characters from the string str, starting at the specified location.
void close ()	This method is used to close the output stream. It will produce an IOException if an attempt is made to write to the output stream after closing the stream.
void flush ()	This method flushes the output stream and writes the waiting buffered characters.
Writer append(char ch)	The append() method appends character ch to the end of the invoking output stream. It returns a reference to the invoking output stream.
Writer append(CharSequence chars)	This method appends chars to the end of the invoking output stream. It returns a reference to the invoking output stream.

END