**FILES**

**CONCEPTS LEARNED**

1. Streams.
2. File creation.
3. Reading and writing to files.
4. File Input stream.
5. File Output stream.
6. Buffered input stream.
7. Buffered Ouptut stream.
8. Program 1.
9. Program 2.
10. Program 3.

**STREAM**

A stream can be defined as a sequence of data.

InutStream − The InputStream is used to read data from a source.

OutPutStream − The OutputStream is used for writing data to a destination.

## **BYTE STREAM**

Java byte streams are used to perform input and output of 8-bit bytes.

FileInputStream and FileOutputStream comes under byte stream since they read and write byte by byte.

**CHARACTER STREAM**

Character streams are used to perform input and output for 16-bit unicode.(i.e) they read and write character by character.

FileReader and FileWriter comes under character stream.

# **FILE CREATION**

We create a file object using File() method as follows,

File f = new File("C:/java/hello");

# **READING AND WRITING FILES**

The InputStream is used to read data from a source and the OutputStream is used for writing data to a destination.

## **FILE INPUT STREAM**

Just use new FileInputStream to create FileInputStream object and use this object to read the file.

With the help of this FileInputStream object any helper methods inside FileInputStream can be used.

File f = new File("C:/java/hello");

InputStream f = new FileInputStream(f);

## **FILE OUTPUT STREAM**

Similarly, just use new FileOutputStream to create FileOutputStream object and use this object to write into a file.

With the help of this FileOutputStream object any helper methods inside FileOutputStream can be used.

## **BUFFERED INPUT STREAM**

This is normally used in cases where a file needs to accessed line by line.

## **BUFFERED OUTPUT STREAM**

Similarly used in cases where files are written line by line.

While using buffered input and output stream objects object.flush() is done.

# **PROGRAM 1**

Create 2 files, test1.txt and test2.txt , write content into them separately, copy content of both files in a new file newtest.txt and delete the files test1.txt and test2.txt.

## **SOLUTION**

# **PROGRAM 2**

Read contents of a file test.txt and write it in another file sample.txt with the condition: Alternate characters capitalized.

Do exception handling and create a custom reader/writer to implement the scenario.

## **SOLUTION**

# **PROGRAM 3**

Read in an xml file , parse it and convert to JSON

## **SOLUTION**