|  |
| --- |
| **Department of Software Engineering**  **Mehran University of Engineering and Technology, Jamshoro** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Course: SWE121 – Object Oriented Programming** | | | |
| **Instructor** | Mr. Asmatullah | **Practical/Lab No.** | 10 |
| **Date** | 12-09-2022 | **CLOs** | CLO-3 |
| **Signature** |  | **Assessment Score** | 1 Marks |

|  |  |
| --- | --- |
| **Topic** | **Understanding I/O Fundamentals** |
| **Objectives** | * To Understand basic File I/O, and Streams. * To understand the Hierarchy of classes to deal with Input and Output streams. * [Reading Ordinary Text Files](https://www.caveofprogramming.com/java/java-file-reading-and-writing-files-in-java.html#readtext) * [Reading Binary Files](https://www.caveofprogramming.com/java/java-file-reading-and-writing-files-in-java.html#readbin) * [Writing Text Files](https://www.caveofprogramming.com/java/java-file-reading-and-writing-files-in-java.html#writetext) * [Writing JSON Files](https://www.caveofprogramming.com/java/java-file-reading-and-writing-files-in-java.html#writebin) |

|  |
| --- |
| **Lab Discussion: Theoretical concepts and Procedural steps** |

**Tools:**  Java Development Kit, Text Pad, Intellij, Eclipse

**Theory**

Outline

* Explain what are streams, input and output in Java. The file class, readers & writers. Typical uses of I/O streams. File reading & writing utilities.
* Use Input/output Streams in programs.

The java.io package contains nearly every class you might ever need to perform input and output (I/O) in Java. All these streams represent an input source and an output destination. The stream in the java.io package supports many data such as primitives, object, localized characters, etc.

## Stream

A stream can be defined as a sequence of data. There are two kinds of Streams −

* InPutStream − The InputStream is used to read data from a source.
* OutPutStream − The OutputStream is used for writing data to a destination.



Java provides strong but flexible support for I/O related to files and networks but this tutorial covers very basic functionality related to streams and I/O. We will see the most commonly used examples one by one −

### Byte Streams

Java byte streams are used to perform input and output of 8-bit bytes. Though there are many classes related to byte streams but the most frequently used classes are, FileInputStream and FileOutputStream. Following is an example

This is test for copy file.

As a next step, compile the above program and execute it, which will result in creating output.txt file with the same content as we have in input.txt. So let's put the above code in CopyFile.java file and do the following −

$javac CopyFile.java

$java CopyFile

Character Streams

Java Byte streams are used to perform input and output of 8-bit bytes, whereas Java Character streams are used to perform input and output for 16-bit unicode. Though there are many classes related to character streams but the most frequently used classes are, FileReader and FileWriter. Though internally FileReader uses FileInputStream and FileWriter uses FileOutputStream but here the major difference is that FileReader reads two bytes at a time and FileWriter writes two bytes at a time.

Standard Streams

All the programming languages provide support for standard I/O where the user's program can take input from a keyboard and then produce an output on the computer screen. If you are aware of C or C++ programming languages, then you must be aware of three standard devices STDIN, STDOUT and STDERR. Similarly, Java provides the following three standard streams −

* Standard Input − This is used to feed the data to user's program and usually a keyboard is used as standard input stream and represented as System.in.
* Standard Output − This is used to output the data produced by the user's program and usually a computer screen is used for standard output stream and represented as System.out.
* Standard Error − This is used to output the error data produced by the user's program and usually a computer screen is used for standard error stream and represented as System.err.

Following is a simple program, which creates InputStreamReader to read standard input stream until the user types a "q" −

* Reading and Writing Files
* As described earlier, a stream can be defined as a sequence of data. The InputStream is used to read data from a source and the OutputStream is used for writing data to a destination.
* Here is a hierarchy of classes to deal with Input and Output streams.



|  |
| --- |
| **Lab Tasks** |

1. Write a java program based on filing. Perform Create, Read and Write operations on JSON file.
2. Write a java program based on filing. Perform Create, Read and Write operations on text file using BufferedReader and BufferedWriter Class.
3. Write a program that opens a file in write mode (use file modes), takes name and roll no from the students writes it in the file, and closes the file.