

**Welcome  
to My  
Presentation!**





Presentation on CSE 331

Lecture: 35

**Presented by**

Syeda Nowshin Ibnat

ID: 17183103020

Intake: 39 (section:1)

Dept. of Computer Science and Engineering





# List of Topics

**01**

File Input/Output: Exploring java.io

**02**

FileInputStream

**03**

FileOutputStream

**04**

ByteArrayInputStream

**05**

ByteArrayOutputStream



1

# File Input/Output: Exploring java.io

## What is File?

- A file is a named location that can be used to store related information.
- For example, **main.java** is a Java file that contains information about the Java program.

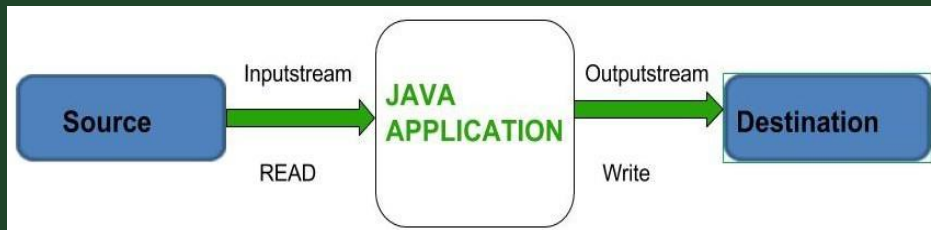
## Using File in java.io

The File class of the java.io package is used to perform various operations on files and directories(collection of files and subdirectories).

## What is File I/O?

File Input → Data Read

File Output → Data Write



## What is java.io?

The java.io package contains nearly every class you might ever need to perform input and output (I/O) in Java.

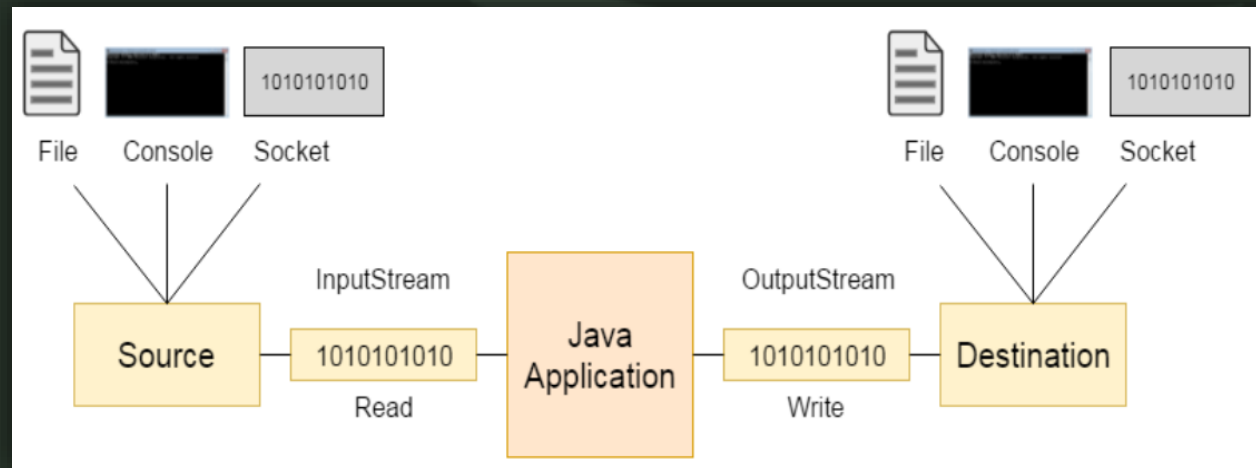


# What is Stream?

- A stream can be defined as a sequence of data. The stream in the java.io package supports many data such as primitives, object etc.

There are two kinds of Streams –

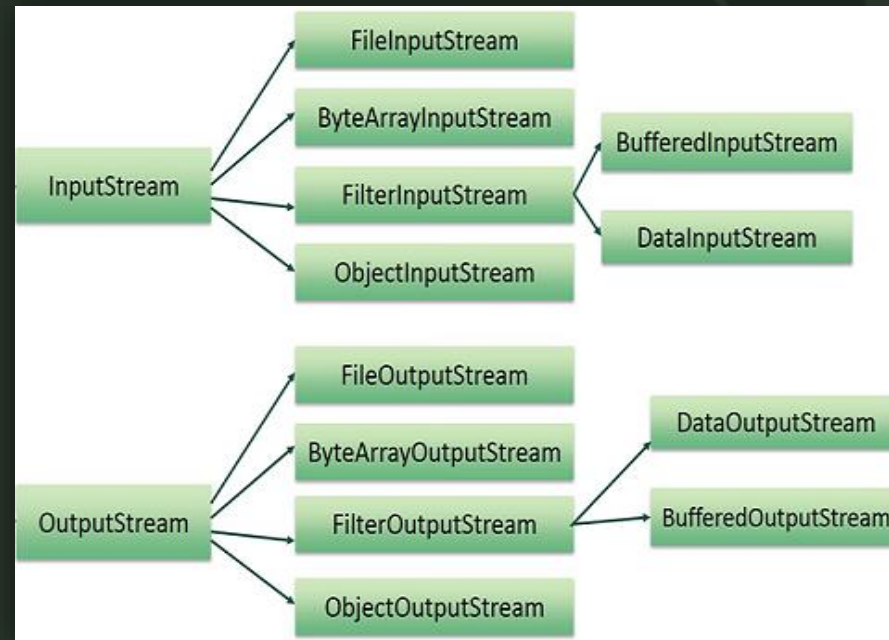
- **InPutStream** – The InputStream is used to read data from a source.
- **OutPutStream** – The OutputStream is used for writing data to a destination.





## What is Byte Stream?

- Byte stream is used to read and write a single byte (8 bits) of data, whereas Java **Character** streams are used to perform input and output for 16-bit unicode.
- Though there are many classes related to byte streams but the most frequently used classes are, **FileInputStream** and **FileOutputStream**.





2

FileInputStream





The **FileInputStream** class creates an **InputStream** that you can use to read bytes from a file.

Input:

```
import java.io.FileInputStream;

public class Main {

    public static void main(String args[]) {

        try {
            FileInputStream input = new FileInputStream("input.txt");

            System.out.println("Data in the file: ");

            // Reads the first byte
            int i = input.read();

            while(i != -1) {
                System.out.print((char)i);

                // Reads next byte from the file
                i = input.read();
            }
            input.close();
        }

        catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
Data in the file:
This is a line of text inside the file.
```



3

FileOutputStream



The `FileOutputStream` class provides implementations for different methods present in the `OutputStream` class.

Input:

```
import java.io.FileOutputStream;

public class Main {
    public static void main(String[] args) {

        String data = "This is a line of text inside the file.";

        try {
            FileOutputStream output = new FileOutputStream("output.txt");

            byte[] array = data.getBytes();

            // Writes byte to the file
            output.write(array);

            output.close();
        }

        catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
This is a line of text inside the file.
```



4

ByteArrayInputStream



The `ByteArrayInputStream` class of the `java.io` package can be used to read an array of input data (in bytes).

Input:

```
import java.io.ByteArrayInputStream;

public class Main {
    public static void main(String[] args) {

        // Creates an array of byte
        byte[] array = {1, 2, 3, 4};

        try {
            ByteArrayInputStream input = new ByteArrayInputStream(array);

            System.out.print("The bytes read from the input stream: ");

            for(int i= 0; i < array.length; i++) {

                // Reads the bytes
                int data = input.read();
                System.out.print(data + ", ");
            }
            input.close();
        }

        catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
The bytes read from the input stream: 1, 2, 3, 4,
```



5

ByteArrayOutputStream



The `ByteArrayOutputStream` class of the `java.io` package can be used to write an array of output data (in bytes).

Input:

```
import java.io.ByteArrayOutputStream;

class Main {
    public static void main(String[] args) {

        String data = "This is a line of text inside the string.";

        try {
            // Creates an output stream
            ByteArrayOutputStream out = new ByteArrayOutputStream();
            byte[] array = data.getBytes();

            // Writes data to the output stream
            out.write(array);

            // Retrieves data from the output stream in string format
            String streamData = out.toString();
            System.out.println("Output stream: " + streamData);

            out.close();
        }

        catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
Output stream: This is a line of text inside the string.
```



## Reference

### Websites:

1) Programiz

<https://www.programiz.com/>

2) Tutorialspoint

<https://www.tutorialspoint.com/java/index.htm>

3) JavaTpoint

<https://www.javatpoint.com/java-tutorial>

### Book:

The Complete reference by Herbert Schildt, 9th edition



**Thank You!**