

## Finally in Exception Handling

```
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

class ExceptionHandlingBasic {
    public static void main(String []args){
        System.out.println("In Main method .....");
        new ExceptionHandlingBasic().printLogs();
        System.out.println("Existing Main Method .....");
    }
    public void printLogs(){
        System.out.println("loading file .....");
        new ExceptionHandlingBasic().openFile();
        System.out.println("finished reading file .....");
    }
    public void openFile(){
        FileReader reader = null; // try has it's own defined context need to
        // define it global to make it visible.
        try {
            reader = new FileReader("/home/neha/java.info/testFile"); //
            // constructor can throw FileNotFoundException
            int i=0;
            while(i != -1){
                //reader.read() may throw IOException
                i = reader.read();
                System.out.print((char) i );
            }
            System.out.println("--- File End ---");
        } catch (FileNotFoundException e) {
            System.out.println(e.getMessage());
            e.printStackTrace();
        } catch (IOException e) {
            System.out.println(e.getMessage());
            e.printStackTrace();
        }finally{
            System.out.println("Inside finally clause.....");
            try{
                if(reader!=null){
                    reader.close();
                }
            }catch (IOException e) {
```

```

        System.out.println(e.getMessage());
        e.printStackTrace();
    }
    System.out.println("Existing finally clause.....");
}
}
}

```

You can attach a finally-clause to a try-catch block. The code inside the finally clause will always be executed, even if an exception is thrown from within the try or catch block. If your code has a return statement inside the try or catch block, the code inside the finally-block will get executed before returning from the method.

Whether an exception is thrown or not inside the try or catch block the code inside the finally-block is executed. The example above shows how the file reader is always closed, regardless of the program flow inside the try or catch block.