```
// joi/9/copy/Copy1.java
//
//
// Copyright 2003 Bill Campbell and Ethan Bolker

import java.io.*;

/**
 * Simple read-a-char, write-a-char loop to exercise file I/O.
 *
 * Usage: java Copy1 inputfile outputfile
 */

public class Copy1
{
    private static final int EOF = -1;  // end of file character rep.

    /**
     * All work is done here.
     *
     * @param args names of the input file and output file.
     */

    public static void main( String[] args )
    {
        FileReader inStream  = null;
        FileWriter outStream = null;
        int ch;

        try {
            // open the files
            inStream  = new FileReader( args[0] );
            outStream = new FileWriter( args[1] );

            // copy
            while ((ch = inStream.read()) != EOF) {
                outStream.write( ch );
            }
        }
        catch (IndexOutOfBoundsException e) {
            System.err.println(
                "usage: java Copy1 sourcefile targetfile" );
        }
        catch (FileNotFoundException e) {
            System.err.println( e );  // rely on e’s toString()
        }
        catch (IOException e) {
            System.err.println( e );
        }
        finally { // close the files
            try {
                if (inStream != null) {
                    inStream.close();
                }
            }
            catch (Exception e) {
```

```
                System.err.println("Unable to close input stream.");
            }
            try {
                if (outStream != null) {
                    outStream.close();
                }
            }
            catch (Exception e) {
                System.err.println("Unable to close output stream.");
            }
        }
    }
}
```

```java
// joi/9/copy/Copy2.java
//
//
// Copyright 2003 Bill Campbell and Ethan Bolker

import java.io.*;

/**
 * Simple read-a-line write-a-line loop to exercise file I/O.
 *
 * Usage: java Copy2 inputfile outputfile
 */

public class Copy2
{
    /**
     * All work is done here.
     *
     * @param args names of the input file and output file.
     */

    public static void main( String[] args )
    {
        BufferedReader inStream  = null;
        BufferedWriter outStream = null;
        String line;

        try {
            // open the files
            inStream =  new BufferedReader(new FileReader(args[0]));
            outStream = new BufferedWriter(new FileWriter(args[1]));

            // copy
            while ((line = inStream.readLine()) != null) {
                outStream.write( line );
                outStream.newLine();
            }
        }
        catch (IndexOutOfBoundsException e) {
            System.err.println(
                "usage: java Copy2 sourcefile targetfile" );
        }
        catch (FileNotFoundException e) {
            System.err.println( e );  // rely on e’s toString()
        }
        catch (IOException e) {
            System.err.println( e );
        }
        finally { // close the files
            try {
                if (inStream != null) {
                    inStream.close();
                }
            }
            catch (Exception e) {
                System.err.println("Unable to close input stream.");
```

```java
            }
            try {
                if (outStream != null) {
                    outStream.close();
                }
            }
            catch (Exception e) {
                System.err.println("Unable to close output stream.");
            }
        }
    }
}
```