



FILE OUTPUT

zyBook 8.4
PrintWriter API, FileOutputStream API

HOW DO WE WRITE TO A FILE?

We will use **PrintWriter** and **FileOutputStream** objects to write to a file.

Syntax:

```
PrintWriter name = new PrintWriter(new FileOutputStream("filename"));
```

Example:

```
String filename = "data.txt";  
FileOutputStream outputStream = new FileOutputStream(filename);  
PrintWriter outputA = new PrintWriter(outputStream);  
  
PrintWriter outputB = new PrintWriter(new FileOutputStream("data.txt"));
```

FileOutputStream constructor will **throw FileNotFoundException** if the file exists but is a directory rather than a regular file, does not exist but cannot be created, or cannot be opened for any other reason.

```
1 import java.io.FileNotFoundException;
2 import java.io.FileOutputStream;
3 import java.io.PrintWriter;
4
5 public class HelloWorldToFile {
6     public static void main(String[] args) {
7         String filename = "test-files/HelloWorld.txt";
8         FileOutputStream outputStream = null;
9         try {
10             outputStream = new FileOutputStream(filename);
11         } catch (FileNotFoundException e) {
12             System.out.println("Output Stream could not be created for " + filename);
13             System.exit(1);
14         }
15         PrintWriter output = new PrintWriter(outputStream);
16         output.println("Hello World!");
17         output.println("Have a great day!");
18         output.close();
19     }
20 }
```

```
$ javac -d bin -cp bin src/HelloWorldToFile.java
```

```
$ java -cp bin HelloWorldToFile
```

test-files/HelloWorld.txt:

```
Hello World!
Have a great day!
```

The image features a dark blue gradient background. In the corners, there are decorative white line art elements resembling circuit traces or neural network connections. These elements consist of thin lines that branch out and terminate in small circles, creating a technical or digital aesthetic.

What would happen if the output file already existed?

OVERWRITING OUTPUT FILE

- If the file does not exist, a new file will be created.
- If the file does exist, it will be **overwritten!**
 - Check to see if file exists before constructing `FileOutputStream`.
 - Consider options for output file that already exists
 - overwrite file
 - stop program
 - prompt user whether they would like to overwrite
 - prompt for a new output file

```
// Check to see if file exists before constructing FileOutputStream  
Path path = Path.of (filename);  
if (Files.exists(path)) {  
    // Consider options for output file that already exists (see above).  
    // Here we are stopping program.  
    System.exit(1);  
}
```

```
1 import java.io.FileNotFoundException;
2 import java.io.FileOutputStream;
3 import java.io.PrintWriter;
4 import java.nio.file.Files;
5 import java.nio.file.Path;
6
7 public class HelloWorldToFile {
8
9     public static void main(String[] args) {
10         String filename = "test-files/HelloWorld.txt";
11         Path path = Path.of (filename);
12         if (Files.exists(path)) {
13             System.out.println("Output file exists and was not overwritten.");
14             System.exit(1);
15         }
16
17         FileOutputStream outputStream = null;
18         try {
19             outputStream = new FileOutputStream(filename);
20         } catch (FileNotFoundException e) {
21             System.out.println("Output Stream could not be created for " + filename );
22             System.exit(1);
23         }
24         PrintWriter output = new PrintWriter(outputStream);
25
26         output.println("Hello World!");
27         output.println("Have a great day!");
28         output.close();
29     }
30 }
```

The background is a dark blue gradient. In the corners, there are white line-art illustrations of circuit boards or neural networks. These consist of vertical and horizontal lines of varying lengths, some ending in small circles, creating a symmetrical, abstract pattern.

More File Input and Output Example at:

<https://pages.github.ncsu.edu/engr-csc116-staff/Schmidt/assets/discussions/FileIO/>