

# Chapter 10

## File Input and Output

### OBJECTIVES

After you have read and studied this chapter, you should be able to

- Include a `FileDialog` object in your program to let the user specify a file.
- Write bytes to a file and read them back from the file using `FileOutputStream` and `FileInputStream`.
- Write values of primitive data types to a file and read them back from the file using `DataOutputStream` and `DataInputStream`.
- Write text data to a file and read them back from the file using `PrintWriter` and `BufferedReader`.
- Write objects to a file and read them back from the file using `ObjectOutputStream` and `ObjectInputStream`.
- Write exception-handling routines using the `try-catch` block..

FIGURE 10.1 Directory structure used for the examples in this section. We assume the Windows 95 environment.

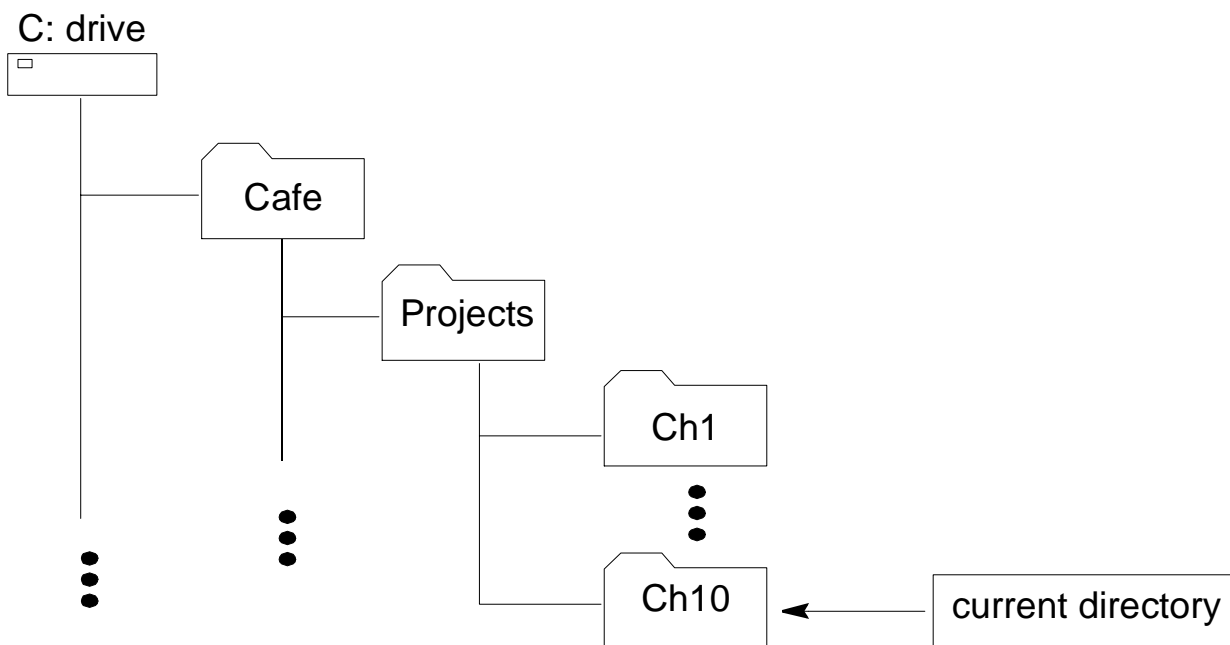


FIGURE 10.2 A **FileDialog** object appearing on the screen in the **LOAD** mode.

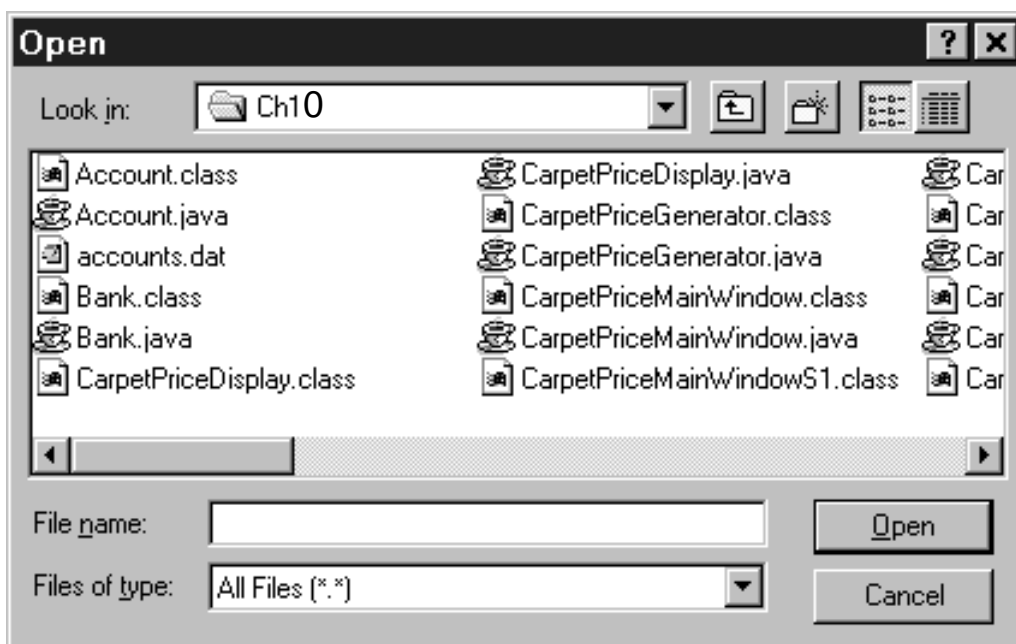


FIGURE 10.3 A **FileBox** object opened in the **SAVE** mode.

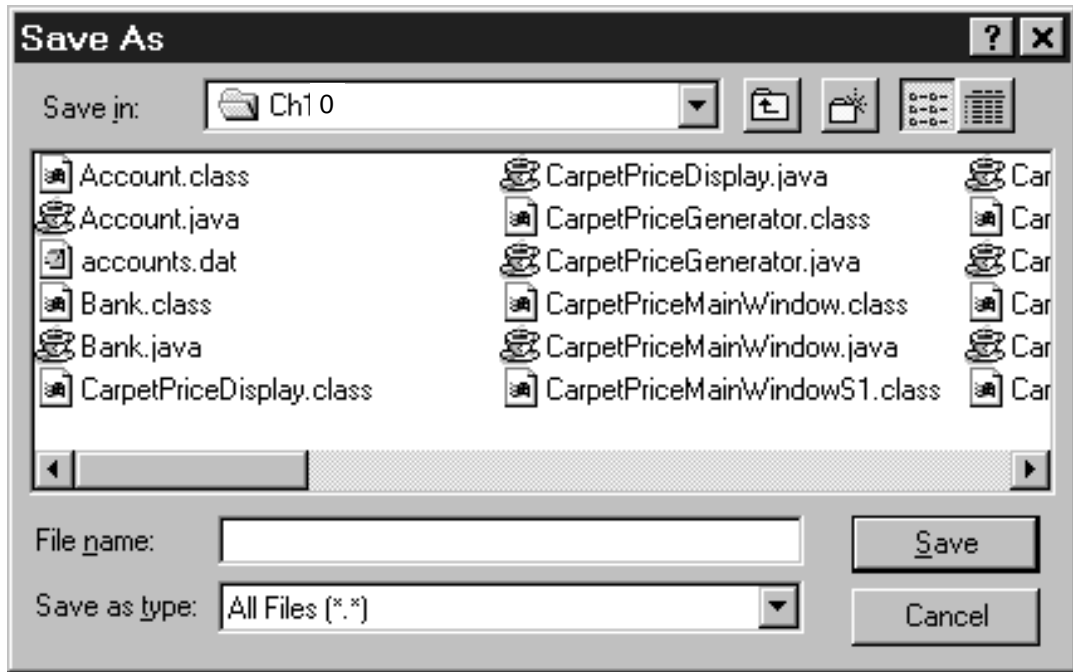


FIGURE 10.4 A message dialog warning of a file overwrite.

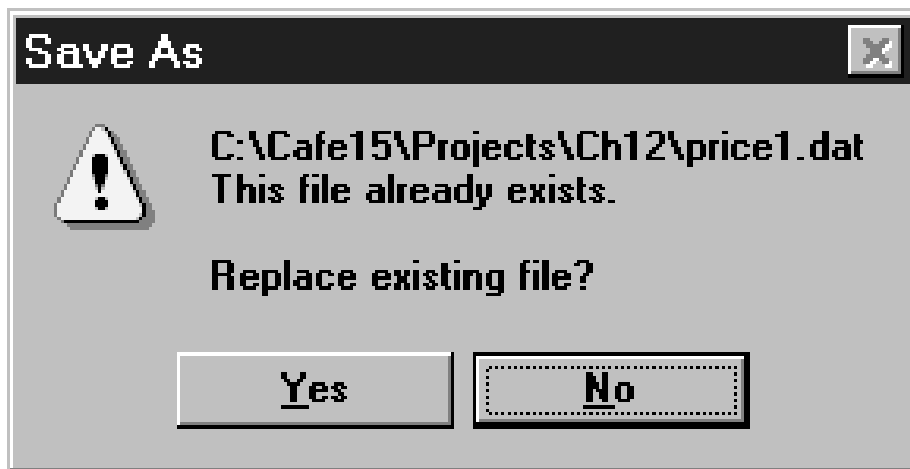


FIGURE 10.5 A diagram showing how the three objects **outFile**, **outFileStream**, and **outDataStream** are related.

```
File        outFile = new File("sample2.data");
FileOutputStream outFileStream = new FileOutputStream(outFile);
DataOutputStream outDataStream
                = new DataOutputStream(outFileStream);
```

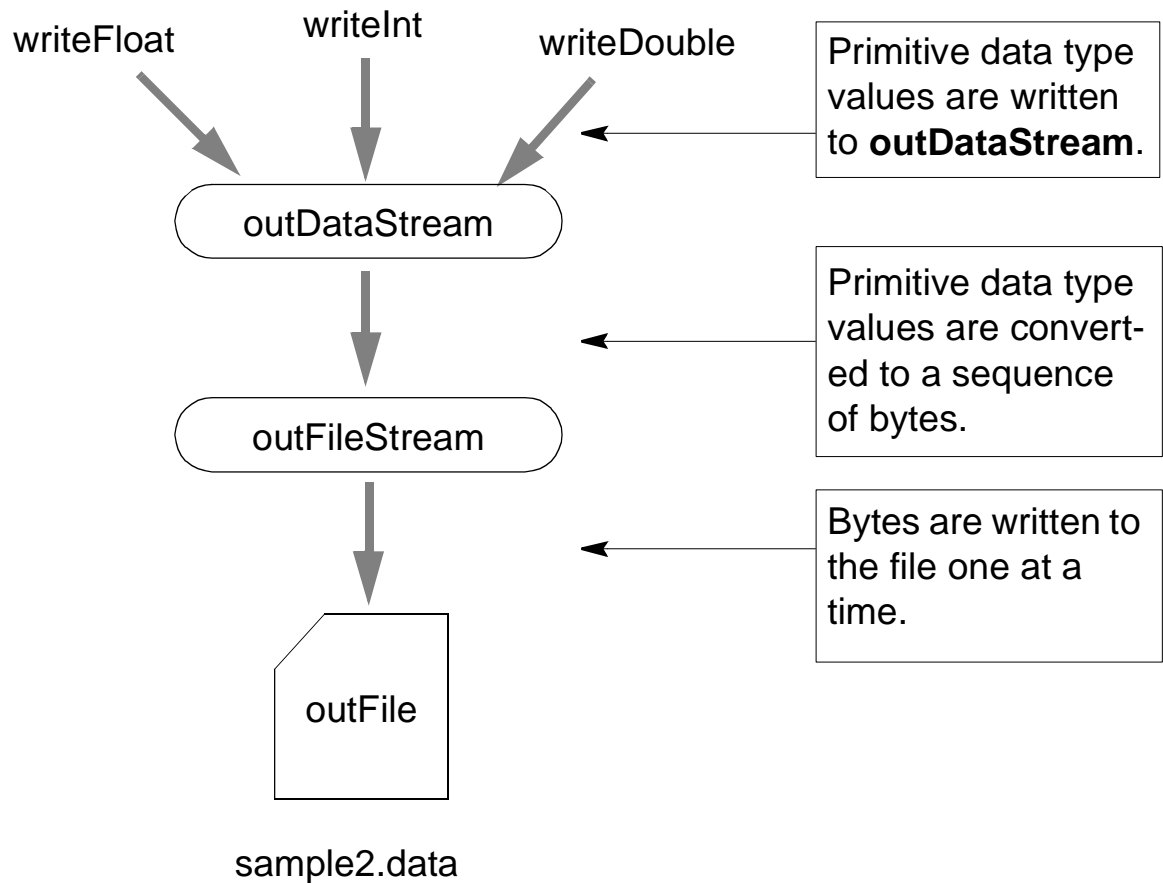


FIGURE 10.6 A diagram showing how the three objects **inFile**, **inFileStream**, and **inDataStream** are related.

```
File        inFile        = new File("sample2.data");
FileInputStream inFileStream = new FileInputStream(inFile);
DataInputStream inDataStream
                = new DataInputStream(inFileStream);
```

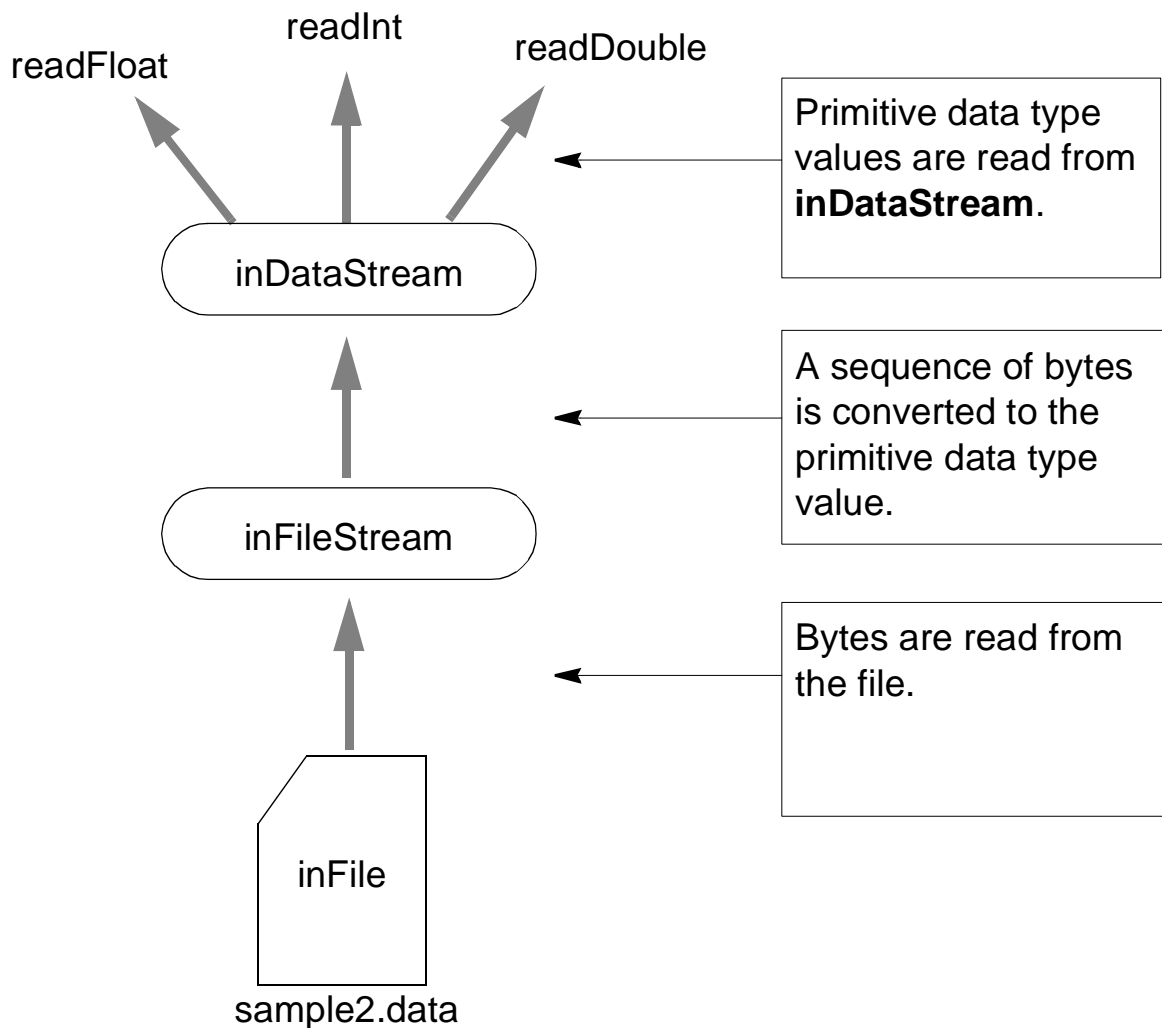
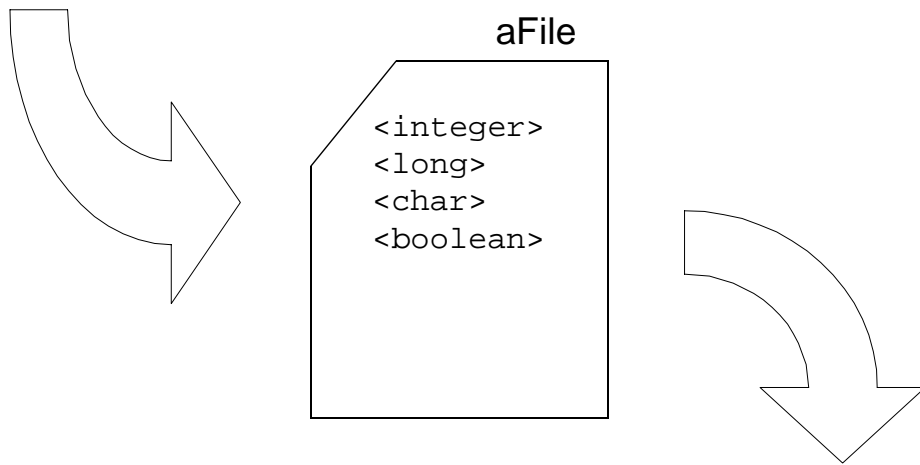


FIGURE 10.7 The order of write and read operations must match to read the stored data back correctly.

```
outStream.writeInteger(...);
outStream.writeLong(...);
outStream.writeChar(...);
outStream.writeBoolean(...);
```



```
inStream.readInteger(...);
inStream.readLong(...);
inStream.readChar(...);
inStream.readBoolean(...);
```