1. No. Data in memory is only temporary working data. A file is data stored on a storage device with a name and location, so data in memory is not called a file until it is saved on such a device.

2. import java.io.*;

3. The error is that the backslash in the path is not escaped. It should be:

```
  File textFile = new File("c:\\inventory.txt");
```

4a. The catch statement (used with try) is used to write an exception handler.

4b.

```
try {
    File myFile = new File("someFile.txt");
    // code that tries to create or open the file
}
catch (IOException e) {
    System.out.println("Error: the specified file name cannot be used to create the file.");
    System.out.println(e.getMessage());
}
```

5a. The stream for displaying error messages is System.err.

5b. These error messages are displayed in the console (command window), unless they are redirected somewhere else.

6a. A file stream keeps track of the current position in the file where the next byte or character will be read or written.

6b. The characters '\r' and '\n' together make up a line terminator.

7. The FileWriter and PrintWriter classes are used together to write data to a file.

8. totalBalance += Double.parseDouble(accountBalance);

9. Object serialization is converting an object into a byte stream so it can be stored in a file or sent over a network. Object deserialization is converting that byte stream back into the original object in memory.

10. The class must implement the java.io.Serializable interface.