C# File Handling Exercises

Exercise 1: Creating and Writing to a Text File

1. **Objective**: Learn to create a file and write simple text to it.
2. **Steps**:
3. In a new console application, use the `System.IO` namespace.
4. Use `File.WriteAllText` to create a file named "example.txt" and write the text "Hello, World!" to it.
5. Run the program and check that "example.txt" contains the text "Hello, World!".
6. **Expected Outcome**: A file named "example.txt" is created with "Hello, World!" as its content.

Exercise 2: Reading from a Text File

1. **Objective**: Read and display text from a file.
2. **Steps**:
3. Ensure "example.txt" exists with some content.
4. Use `File.ReadAllText` to read the contents of "example.txt".
5. Print the contents to the console.
6. **Expected Outcome**: The console displays the content of "example.txt".

Exercise 3: Writing Multiple Lines to a File

1. **Objective**: Write multiple lines to a text file.
2. **Steps**:
3. Create a `string[]` array with several strings (e.g., "Line 1", "Line 2", "Line 3").
4. Use `File.WriteAllLines` to write these lines to a new file called "multilines.txt".
5. Verify the file to ensure each string appears on a new line.
6. **Expected Outcome**: A file named "multilines.txt" is created with each string on a separate line.

Exercise 4: Appending Text to a File

1. **Objective**: Add text to an existing file without overwriting it.
2. **Steps**:
3. Use `File.AppendAllText` to add the line "Additional Line" to "example.txt".

4. Run the program and check "example.txt" to see the new line appended.
5. **Expected Outcome**: "example.txt" now contains the original text plus "Additional Line" at the end.

Exercise 5: Checking if a File Exists

1. **Objective**: Check if a file exists before performing operations.
2. **Steps**:
3. Use `File.Exists` to check if "example.txt" exists.
4. If it does, read and print its content; otherwise, display a message like "File does not exist."
5. **Expected Outcome**: The program either reads and displays "example.txt" or prints a message saying it does not exist.

Exercise 6: Deleting a File

1. **Objective**: Safely delete a file.
2. **Steps**:
3. Use `File.Exists` to check if "example.txt" exists.
4. If it does, use `File.Delete` to delete it.
5. Run the program and check that "example.txt" no longer exists.
6. **Expected Outcome**: "example.txt" is deleted if it existed.

Exercise 7: Reading a File Line by Line

1. **Objective**: Read each line in a file individually.
2. **Steps**:
3. Create a file named "sample.txt" with multiple lines of text.
4. Use `File.ReadLines` to read the file line by line.
5. Print each line to the console.
6. **Expected Outcome**: Each line from "sample.txt" is printed on a new line in the console.

Exercise 8: Using FileStream to Write Binary Data

1. **Objective**: Learn to work with binary files.
2. **Steps**:
3. Use `FileStream` to create and open a file named "binaryfile.dat" for writing.

4. Use `BinaryWriter` with the `FileStream` to write some integer and string data.
5. Close the file and verify its creation.
6. **Expected Outcome**: "binaryfile.dat" is created with binary data that can be opened later with a binary reader.

Exercise 9: Reading Binary Data with FileStream

1. **Objective**: Read and interpret binary data from a file.
2. **Steps**:
3. Use `FileStream` and `BinaryReader` to open and read the data from "binaryfile.dat".
4. Display the read integer and string data on the console.
5. Verify the data matches what was written in the previous exercise.
6. **Expected Outcome**: The console displays the integer and string data stored in "binaryfile.dat".

Exercise 10: File Copy and Move Operations

1. **Objective**: Learn to copy and move files.
2. **Steps**:
3. Create a file named "original.txt" with some sample text.
4. Use `File.Copy` to make a copy named "copy.txt".
5. Use `File.Move` to move "copy.txt" to a new location, renaming it to "moved.txt".
6. Verify that "copy.txt" is moved and renamed as "moved.txt".
7. **Expected Outcome**: The file is successfully copied and moved to a new location with the new name.