

Versuchen Sie immer, Ihren Code zu kommentieren!

Aufgabe 1

Schreiben Sie ein Programm, das ein Verzeichnis namens "TestFolder" auf dem Desktop erstellt. Innerhalb dieses Verzeichnisses ist eine Textdatei namens "example.txt" zu erstellen, in die der Text **"Hello, World!"** geschrieben wird. Nach dem Erstellen der Datei ist zu überprüfen, ob die Datei erfolgreich erstellt wurde, und das Ergebnis soll in der Konsole ausgegeben werden (true, false). Es sollten die Klassen **Path**, **Directory** und **File** verwendet werden. Um die Existenz der Datei zu prüfen, ist die Methode **File.Exists()** zu nutzen.

Aufgabe 2

In dieser Aufgabe ist der Inhalt der zuvor erstellten Datei "example.txt" auszulesen und in der Konsole auszugeben. Anschließend soll eine neue Zeile mit dem Text **"This is a new line."** am Ende der Datei hinzugefügt werden. Der aktualisierte Inhalt der Datei ist erneut auszulesen und auszugeben. Für das Lesen des Inhalts kann **File.ReadAllText()** verwendet werden, während das Anhängen einer Zeile mit **File.AppendAllText()** erfolgen sollte.

Aufgabe 3

Zuerst ist ein Basispfad für einen Ordner namens "MyFiles" auf dem Desktop zu erstellen. Mithilfe von **Path.Combine()** sind drei Dateipfade für die Dateien "document1.txt", "document2.txt" und "document3.txt" zu erstellen. Anschließend sind die Dateinamen und Erweiterungen dieser drei Dateipfade in der Konsole auszugeben. Die Methoden **Path.GetFileName()** und **Path.GetExtension()** können verwendet werden, um den Dateinamen und die Dateierweiterung zu extrahieren.

Aufgabe 4

Die Datei "example.txt", die in einer vorherigen Aufgabe erstellt wurde, soll in ein neues Verzeichnis namens "Backup" auf dem Desktop kopiert werden. Schließlich ist die Originaldatei im Verzeichnis "TestFolder" zu löschen, nachdem überprüft wurde, dass sie erfolgreich verschoben wurde.

Aufgabe 5

Ein Verzeichnis namens "Logs" ist auf dem Desktop zu erstellen, wobei vorher zu prüfen ist, ob es bereits existiert. Nach dem Erstellen des Verzeichnisses sollen darin mehrere leere Textdateien angelegt werden, wie z.B. "log1.txt", "log2.txt" und "log3.txt". Schließlich sollen alle Dateien im "Logs"-Verzeichnis aufgelistet und ihre Namen in der Konsole ausgegeben werden.

Aufgabe 6

Es soll eine rekursive Funktion implementiert werden, die alle Dateien und Verzeichnisse auf dem Desktop des aktuellen Benutzers auflistet. Jedes Verzeichnis und jede Datei soll dabei mit einer Einrückung pro neuer Unterverzeichnisebene angezeigt werden. Der Pfad zum Desktop kann über `Environment.GetFolderPath(Environment.SpecialFolder.Desktop)` ermittelt werden. Für diese Aufgabe sind `Directory.GetDirectories()` und `Directory.GetFiles()` zu verwenden, um die Verzeichnisstruktur rekursiv zu durchlaufen.

C:\Users\<username>\Desktop

 C:\Users\<username>\Desktop\Dir1

 C:\Users\<username>\Desktop\Dir1\test1.txt

 C:\Users\<username>\Desktop\Dir1\test2.txt

 C:\Users\<username>\Desktop\Dir2

 C:\Users\<username>\Desktop\Dir2\test3.txt

 C:\Users\<username>\Desktop\Dir2\text4.txt