**Quiz: Einführung in Tasks, async und await in C#**

**Teil 1: Theorie (Multiple Choice / Richtig-Falsch)**

**1. Was passiert bei synchroner Programmierung, wenn eine lang laufende Operation ausgeführt wird?**  
a) Der Hauptthread kann andere Aufgaben ausführen.  
b) Der Hauptthread wird blockiert, bis die Operation abgeschlossen ist.  
c) Die Operation wird automatisch im Hintergrund ausgeführt.

**2. Welche Aussage über asynchrone Programmierung ist korrekt?**  
a) Asynchronität blockiert den Hauptthread.  
b) Asynchronität ermöglicht paralleles Ausführen von Aufgaben.  
c) Async-Methoden laufen immer in einem neuen Thread.

**3. Richtig oder Falsch:** Ein Task kann einen Rückgabewert enthalten.

**4. Welche Rückgabewerte sind für async-Methoden erlaubt?**  
a) void  
b) Task  
c) Task  
d) alle oben genannten

**5. await:**  
a) Blockiert den aktuellen Thread, bis der Task fertig ist.  
b) Wartet auf den Task, ohne den Thread zu blockieren.  
c) Startet automatisch einen neuen Task.

**Teil 2: Codeverständnis**

**6. Was wird ausgegeben?**

static async Task BeispielAsync()

{

Console.WriteLine("Start");

await Task.Delay(2000);

Console.WriteLine("Ende");

}

a) Start → 2 Sekunden Pause → Ende  
b) Start → Ende → 2 Sekunden Pause  
c) Keine Ausgabe

**7. Was passiert, wenn Task.Run() verwendet wird?**  
a) Die Aufgabe wird im gleichen Thread ausgeführt.  
b) Die Aufgabe wird in einem separaten Thread oder Hintergrundtask ausgeführt.  
c) Der Code wird sofort blockiert, bis die Aufgabe fertig ist.

**8. Welche Methode sollte verwendet werden, um mehrere Tasks gleichzeitig abzuwarten?**  
a) Task.Wait()  
b) Task.WhenAll()  
c) Thread.Sleep()

**9. Welche Aussage zum Frühstücks-Beispiel ist korrekt?**  
a) Synchron dauert es nur so lang wie die längste Aufgabe.  
b) Synchron dauert es so lang wie die Summe aller Aufgaben.  
c) Asynchron dauert es länger als synchron.

**Teil 3: Kurze Aufgaben / Verständnis**

**10. Sie wollen eine asynchrone Methode schreiben, die 5 Sekunden wartet. Wie könnte die Methode aussehen?**

**11. Sie haben zwei Tasks: einer dauert 5 Sekunden, der andere 2 Sekunden. Wie lange dauert die asynchrone parallele Ausführung ungefähr?**  
a) 2 Sekunden  
b) 5 Sekunden  
c) 7 Sekunden

**12. Erklären Sie in einem Satz den Unterschied zwischen Task und Task.**

**13. Welche C#-Klasse wird verwendet, um die Zeit einer Operation zu messen?**

**Teil 4: Kurze Rechenaufgabe**

**14. Wenn ein Webserver 36.615.980 Seitenaufrufe in 2.592.000 Sekunden verarbeitet, wie viele Seitenaufrufe pro Sekunde sind das?**

**15. Sie starten drei Aufgaben parallel:**

* Kaffee kochen (5 min)
* Eier braten (7 min)
* Toast toasten (3 min)

Wie lange dauert das Frühstück asynchron?

**Antworten**

1. b) Der Hauptthread wird blockiert, bis die Operation abgeschlossen ist.
2. b) Asynchronität ermöglicht paralleles Ausführen von Aufgaben.
3. Richtig
4. d) alle oben genannten
5. b) Wartet auf den Task, ohne den Thread zu blockieren.
6. a) Start → 2 Sekunden Pause → Ende
7. b) Die Aufgabe wird in einem separaten Thread oder Hintergrundtask ausgeführt.
8. b) Task.WhenAll()
9. b) Synchron dauert es so lang wie die Summe aller Aufgaben.

static async Task Warte5SekundenAsync()  
{  
await Task.Delay(5000);  
}

11. b) 5 Sekunden

12. Task führt eine Aufgabe ohne Rückgabewert aus, Task<T> liefert einen Wert zurück.

13. Stopwatch

14. 36.615.980 ÷ 2.592.000 ≈ 14,1 Seitenaufrufe/Sekunde

15. 7 Minuten, da die längste Aufgabe die Dauer bestimmt

---

Wenn du willst, kann ich daraus direkt ein \*\*fertig formatiertes PDF-Layout\*\* erstellen, z. B. mit \*\*Fettschrift, nummerierten Abschnitten und gut lesbarer Struktur\*\*, sodass es direkt in Word oder LibreOffice importiert werden kann.

Willst du, dass ich das mache?