

Aufgabe – Filter-Operationen

Es sei folgendes Array gegeben:

```
int[] numbers = { 5, 4, 1, 3, 9, 8, 6, 7, 2, 0, 22, 12, 16, 18, 11, 19, 13 };
```

Ermitteln Sie mittels LINQ-Ausdrücken die folgenden Informationen:

1. Alle Zahlen echt kleiner als 7
2. Alle geraden Zahlen
3. Alle einstelligen ungeraden Zahlen
4. Alle geraden Zahlen ab dem 6 Element (einschließlich) im Array

Es sei folgendes Array gegeben:

```
string[] numbers =  
{ "zero", "one", "two", "three", "four", "five", "six", "seven", "eight",  
  "nine", "ten", "eleven", "twelve", "thirteen", "fourteen" };
```

Ermitteln Sie mittels LINQ-Ausdrücken die folgenden Informationen:

1. Alle „Zahlen“ die drei Zeichen lang sind
2. Alle „Zahlen“ die ein „o“ enthalten
3. Alle „Zahlen“ die auf „teen“ enden
4. Die Großbuchstabendarstellung aller „Zahlen“ die auf „teen“ enden
5. Alle „Zahlen“ die „four“ enthalten

Aufgabe – Aggregationen

Es sei folgendes Array gegeben:

```
int[] numbers = { 5, 4, 1, 3, 9, 8, 6, 7, 2, 0, 22, 12, 16, 18, 11, 19, 13 };
```

Ermitteln Sie mittels LINQ-Ausdrücken die folgenden Informationen:

1. Die Summe aller Werte im Array
2. Die kleinste Zahl
3. Die größte Zahl
4. Den Durchschnittswert
5. Die kleinste gerade Zahl
6. Die größte ungerade Zahl
7. Die Summe aller geraden Zahlen
8. Den Durchschnittswert aller ungeraden Zahlen