

C# Essentials 2D arrays

Koen Bloemen



**DE HOGESCHOOL
MET HET NETWERK**

Elfde-Liniestraat 24, 3500 Hasselt, www.pxl.be





- 2D arrays

2D arrays

- **1D array**

- is 1 rij
- ['a']['b']['c']['d']['e']['f']

- **2D array**

- tabel van rijen en kolommen.
- ['a']['b']['c']
- ['d']['e']['f']

- Bepaal indices?

- 1D: a (0de element), b (1), c (2), d (3), e (4), f (5)
- 2D: a (0de rij, 0de kolom), b (0, 1), c (0, 2), d (1, 0), e (1, 1), f (1, 2)

2D arrays

- Declaratie

```
int[,] getallen = new int[4, 2]; // 4 rijen en 2 kolommen  
int[,] getallekes = new int[2, 4]; // 2 rijen en 4 kolommen
```

- Initialisatie

```
int[,] getallen = new int[4, 2] { { 8, 6 }, { 9, 1 }, { 7, 8 }, { 5, 6 } };  
int[,] getallen = new int[,] { { 8, 6 }, { 9, 1 }, { 7, 8 }, { 5, 6 } };  
int[,] getallen = { { 8, 6 }, { 9, 1 }, { 7, 8 }, { 5, 6 } };
```

- Waarom gaat deze niet?

```
int[,] getallen = { { 8, 6, 4 }, { 9, 1 }, { 7, 8, 2, 1 }, { 5, 6 }, { 1 } };
```

// Aantal kolommen moet telkens gelijk blijven.

2D arrays

- Loopen

```
int aantalRijen = getallen.GetLength(0); // 0 geeft aan: rijen
int aantalKolommen = getallen.GetLength(1); // 1 geeft aan: kolommen
int totaalAantalElementen = getallen.Length;
for (int r = 0; r < aantalRijen; r++)
{
    for (int k = 0; k < aantalKolommen; k++)
    {
        getallen[r, k] = r + k;
    }
}
```

Hulpfuncties arrays

- Herinner Clear() om een array leeg te maken

```
// 1D Array leegmaken
float[] getallen = { 1.1f, 1.2f, 1.3f, 1.4f, 1.5f, 1.6f, 1.7f };
int startIndex = 2;
int length = 4;
Array.Clear(getallen, startIndex, length);
// { 1.1f, 1.2f, 0.0f, 0.0f, 0.0f, 0.0f, 1.7f }

// Hoe volledig leegmaken?
int startIndex = 0;
int length = getallen.Length;
Array.Clear(getallen, startIndex, length);
```

Hulpfuncties arrays

- Clear(): Array leegmaken

```
// 2D Array leegmaken
float[,] getallen = { { 1.1f, 1.2f }, { 1.3f, 1.4f }, { 1.5f, 1.6f }, { 1.7f, 1.8f } };
int startIndex = 3;
int length = 4;
Array.Clear(getallen, startIndex, length);
foreach (float getal in getallen)
{
    Console.WriteLine(getal);
}
// Wat is het resultaat?
{ {1.1f, 1.2f}, {1.3f, 0.0f}, {0.0f, 0.0f}, {0.0f, 1.8f} }
```