

# Integratie Externe Functionaliteiten Les 7 : Matplotlib grafieken stijlen

Met grafieken gaan we nu data visueel voorstellen. Hier komt nog iets extra bij kijken tegenover de tekstuele voorstelling waarmee we tot nu toe mee gewerkt hebben. Het stijlen van deze grafieken. Denk aan de lijn, de losse punten en de assen van de grafiek zelf.

We willen controle krijgen over de design van de data. Dit gaan we met behulp van code moeten aanpassen.

We beginnen met de lijnen. Er zijn 3 instellingen die we kunnen aanpassen. De soort lijn, de dikte en de kleur.

## Lijnen

### Lijnstijl

Als eerst bekijken we de soort lijn, **linestyle**. We geven deze parameter mee in de functie `plot()` na de array's met punten. We kunnen zowel **linestyle** als **ls** gebruiken.

De stijl is standaard weergegeven door een volle lijn, **solid**.

Er zijn 4 stijlen die we bij naam met een string, vergeet de quotes niet, kunnen invullen.

- |                  |    |                                  |
|------------------|----|----------------------------------|
| 1. Solid of -    | => | volle lijn                       |
| 2. dotted of :   | => | puntjes                          |
| 3. dashed of --  | => | streepjes                        |
| 4. dashdot of -. | => | afwisselend puntjes en streepjes |

Maak 2 array's met 4 y-punten.

Plot de array's en probeer alle lijnstijlen uit.

Als we onze smaak niet terugvinden tussen deze 4 of we een aanpassing nodig hebben, kunnen we ook specifiekere te werk gaan. Dan gebruiken we geen string, maar een tuple.

In de documentatie van matplotlib vinden we terug hoe:

[https://matplotlib.org/stable/gallery/lines\\_bars\\_and\\_markers/linestyles.html](https://matplotlib.org/stable/gallery/lines_bars_and_markers/linestyles.html)

## Kleur

We kunnen de kleur instellen met de parameter **color** of afgekort **c**.

Dit gaat op dezelfde manier te werk als bij webdesign.

Dit wil zeggen. Er zijn 140 kleuren die bij naam gekend zijn. Deze vullen we in met een string.

Een overzicht:

[https://www.w3schools.com/colors/colors\\_names.asp](https://www.w3schools.com/colors/colors_names.asp)

Verder kunnen we gebruik maken van hexadecimale code. Hiervoor moeten we een beetje digitale kleurenleer kennen.

Simpel uitgedrukt:

Elke pixel op een scherm bestaat uit 3 kleurenlampjes.

**Rood**, **Groen** en **Blauw**. Een code met 6 hexadecimale waarden zal in groepjes van 2 elk kleurenlampje voorstellen. Die bepaalt hoe fel dit lampje zal schijnen.

Het hexadecimale stelsel bestaat uit 16 waarden ipv 10: 0123456789abcdef. Waarbij 0 volledig uit betekent en f volledig open.

Deze code wordt steeds voorafgegaan door een #.

Bijvoorbeeld wit wordt dan: #ffffff. Alle kleuren open, samen maakt dit wit.

Zwart wordt dan #000000. Alle kleuren op nul.

#**ff6347** zal een samenstelling geven.

ff voor rood, 63 voor groen en 47 voor blauw.

Zo hebben we 256 mogelijkheden per kleur.  $256 \times 256 \times 256 = 16\,777\,216(!)$ . Daar vinden we dus zeker onze favoriete kleur in terug.

Volgende sites kunnen ons zeker helpen met een keuze:

<https://color.adobe.com/> of

[https://www.w3schools.com/colors/colors\\_picker.asp](https://www.w3schools.com/colors/colors_picker.asp)

Voeg bij de vorige grafiek de parameter voor kleur toe.  
Zet de grafiek in je favoriete hexadecimale kleur.

De hoofdkleuren voor print en scherm hebben nog eens een eigen afkorting met één letter.

Scherms:

r	=>	rood
g	=>	groen
b	=>	blauw

print:

y	=>	yellow
m	=>	magenta
c	=>	cyaan

zwart/wit

k	=>	zwart
w	=>	wit

## Lijndikte

Als laatste parameter kunnen we de lijndikte meegeven. Deze stellen we in met **linewidth** of **lw**.

De lijndikte wordt weergegeven in punten. Dit kan ook een kommagetal zijn. Deze wordt niet weergegeven als een float, maar ook als een string.

De standaard lijndikte is 1.5pt.

Pas tenslotte ook de lijndikte van onze grafiek aan.

# Markers

Punten kunnen we ook markeren op de grafiek. Dit gebruiken we om de verschillende punten duidelijker weer te geven, maar ook om losse punten te tonen zonder verbindingen.

## Markerstijl

Markers komen niet standaard voor op de grafiek. Er is dus geen default setting. Wel gaan we deze doorgaans als een bolletje voorstellen.

We gebruiken hiervoor de parameter **marker**. In een string stellen we dan in welk symbool we willen gebruiken. Voor een bolletje gebruiken we de letter o.

```
plt.plot(xpoints, ypoints, marker = 'o')
```

Merk op dat de lijn niet verdwenen is. De marker wordt dus toegevoegd en vervangt de lijn niet.

Als we de lijn weg willen, stellen we de **linestyle** in met een lege string.

Als we alleen met punten werken is dit meestal om data op een speciale manier weer te geven dus gebruiken we niet **plot**, maar **scatter**. Hier later meer over.

Alle soorten markers vinden we terug in de Matplotlib documentatie:

[https://matplotlib.org/stable/api/markers\\_api.html](https://matplotlib.org/stable/api/markers_api.html)

Let wel op er is een verschil tussen de integers en de strings met getallen. We kunnen wel beide gebruiken.

## Kleur

Als we de parameter **color** meegeven zien we dat heel onze grafiek, inclusief de markers, deze ingestelde kleur zal krijgen.

Maar wat als we de markers harder willen laten opvallen?

Dan hebben we 2 parameters die specifiek de kleur van de marker aanpassen.

Een parameter voor de *vulkleur*, **markerfacecolor** of **mfc**.

En een parameter voor de *randkleur*, **markeredgecolor** of **mec**.

De waarde van deze parameters stellen we in zoals de parameter **kleur**. Met kleurnaam, hexadecimaal of afkorting.

Als we de hele bol willen aanpassen moeten we de rand- en vulkleur op dezelfde waarde instellen.

Pas de kleur van de markeringen aan.

Gebruik een verschillende kleur voor de rand- en de vulkleur.

## Markergrootte

De markeringen kunnen ook van groter of kleiner worden weergegeven.

Dit met de parameter **markersize** of **ms**.

We stellen een getal in een string in. De standaardwaarde is 6.

Als we de randdikte willen aanpassen is er de parameter **markeredgewidth** of **mew**. Deze rand zal naar binnen groeien. Ze maakt de markering dus niet groter.

Maak giga markeringen met een dikkere rand.

## Shorthand stijlen

Als we geen complexe stijl instellingen hebben maken we gebruik van **format** of **fmt**. In deze parameter geven we 3 instellingen tegelijk.

We benoemen de parameter **NIET**.

De **markerstijl**, **lijnstijl** en **kleur**.

De 3 waarden stellen we dan in aan de hand van de afkorting. In een string, zonder spaties of komma's tussen de waarden.

Onze keuzes worden dus sterk beperkt.

Stijl de grafiek aan de hand van shorthand.

Een groene grafiek met driehoekjes en een streepjeslijn.

## Labels en titels

Onze grafiek heeft nu handige kleuren en markeringen. Toch blijven de assen onduidelijk als we alleen met getallen als data werken. We kunnen dit verduidelijken door er labels aan toe te voegen.

We gebruiken hiervoor geen parameters, maar extra functies.

Er is er één voor elke as: **xlabel()** en **ylabel()**.

Tussen de haakjes zetten we dan een string met de tekst die moet verschijnen.

Deze functie moet worden toegevoegd voor de functie **show()** die de grafiek zal tonen.

Met de functie **title()** voegen we op eenzelfde manier een titel toe aan de grafiek.

Voeg labels voor de x- en y-as en een titel toe.

Om de tekst duidelijker te maken kunnen we ook hier stijl aan geven.

We stellen de stijl samen in een dictionary en passen die nadien toe met de parameter **fontdict**.

De belangrijkste eigenschappen zijn:

family	=>	lettertype, een string
color	=>	kleur, een string
size	=>	lettergrootte, een integer

een dictionary ziet er dan zo uit:

```
font1 = {'family':'serif','color':'blue','size':20}
```

Een beetje typografie:

We onderscheiden een aantal grote font-types:

serif	=>	schreeflettertype
sans-serif	=>	schreefloos
cursive	=>	<i>handgeschreven</i>
monospace	=>	computerachtig
fantasy	=>	specialekes

Als we 1 van deze types instellen zal de computer zelf een lettertype van die soort kiezen.

We kunnen echter ook specifiek een lettertype bepalen. Het default lettertype is *DejaVu Sans*.

Let wel op, het lettertype moet wel geïnstalleerd zijn op de pc.

Er zijn verschillende sites waar we gratis lettertypes kunnen downloaden om onze lettertypebibliotheek aan te vullen.

**Maak 2 fontstijlen aan. Pas die respectievelijk toe op de titel en de assen.**

Om tenslotte de positie van de titel en labels te bepalen gebruiken we de parameter **loc**.

Standaard staat deze op **center**, maar met **left** en **right** verplaatsen we die.

## Raster of grid

De functie `grid()` voegt rasterlijnen toe aan de grafiek. We hoeven hier niets mee te geven tussen de haakjes voor een standaard raster.

Met de parameter **axis** kunnen we slechts 1 as van het raster tonen. We kiezen dan tussen de waarden **x** en **y** in een string.

Zijn we niet tevreden met de grijze stijl van het raster? Dan gebruiken we dezelfde parameters als bij de lijnen van onze grafiek om deze aan te passen.

Benader volgende grafiek en maak er zelf nog eentje naar keuze:

