

Maak GUIs met Python: Introductie in guizero

Installeer de guizero Python-bibliotheek en begin met het maken van je eigen GUI's.

Een grafische gebruikersinterface (GUI) is een manier om je Python-programma's gebruiksvriendelijker en spannender te maken.



Laura Sach

Laura leidt het A Level team van de Raspberry Pi Foundation en creëert zo middelen voor leerlingen om te leren over informatica.

@CodeBoom

MAKER



Martin O'Hanlon

Martin werkt in het leerteam van de Raspberry Pi Foundation, waar hij online cursussen, projecten en leermiddelen maakt.

@martinohanlon

MAKER

Je kunt verschillende componenten, genaamd 'widgets', aan je interface toevoegen, waardoor er veel verschillende manieren zijn om informatie in het programma in te voeren en als uitvoer weer te geven. Je kunt mensen op een knop laten drukken, een stukje tekst laten weergeven of zelfs een optie laten kiezen uit een menu. In deze serie zullen we gebruik maken van de *guizero*-bibliotheek, die is ontwikkeld met als doel beginners te helpen bij het gemakkelijk bouwen van GUI's. Python's standaard GUI-pakket heet *tkinter*, en is met Python op de meeste platformen al geïnstalleerd. De *guizero*-bibliotheek is een *wrapper* voor *tkinter* – dit betekent dat het een veel eenvoudiger manier biedt om Python's standaard GUI-bibliotheek te gebruiken.

01 Installeer guizero

Je moet de *guizero* Python-bibliotheek ([lawsie.github.io/guizero](https://github.com/lawsie/guizero)) installeren om de programma's in dit artikel te kunnen maken. Het is beschikbaar als Python-pakket, wat herbruikbare code is die je kunt downloaden, installeren en vervolgens kunt gebruiken in je programma's. Hoe je *guizero* installeert hangt af van je besturingssysteem en de rechten die je hebt om je computer te bedienen. Als je toegang hebt tot de commandoregel of de terminal, kun je het volgende commando gebruiken:

```
pip3 install guizero
```

Uitgebreide installatie-instructies voor *guizero* zijn beschikbaar op lawsie.github.io/guizero, inclusief opties voor het installeren wanneer je geen administratorrechten hebt op je computer en downloadbare installaties voor Windows.

02 Hallo Wereld

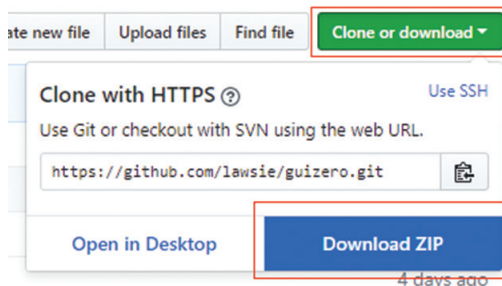
Nu je *guizero* hebt geïnstalleerd, laten we eens kijken of het werkt en een kleine 'hello-world' app schrijven, wat traditioneel is voor programmeurs om te doen als eerste programma bij het gebruik van een nieuwe tool of taal. Open de editor waar je je Python-code mee zult schrijven. Aan het begin van elk *guizero*-programma kies je de widgets die je nodig hebt uit de *guizero*-bibliotheek en importeer je ze. Je hoeft elke widget maar één keer te importeren en kunt hem vervolgens zo vaak als je wilt in je programma gebruiken. Zet deze code bovenaan de pagina om de App-widget te importeren:

```
from guizero import App
```

Alle *guizero*-projecten beginnen met een hoofdvenster dat een container-widget is die App heet. Aan het einde van elk *guizero*-programma moet je het programma vertellen dat de app die je net hebt gemaakt moet worden weergegeven. Voeg deze twee regels code toe onder de regel waar je de App-widget heeft geïmporteerd:

```
app = App(title="Hello world")
app.display()
```

Sla nu je code op en voer die uit. Je zou nu een GUI-venster moeten zien met de titel 'Hello world' (Figuur 1). Gefeliciteerd, je hebt zojuist je eerste *guizero*-app gemaakt!



► Een alternatieve manier om *guizero* te installeren is door het zip-bestand van GitHub te downloaden.



▲ **Figuur 1** Je eerste guizero app

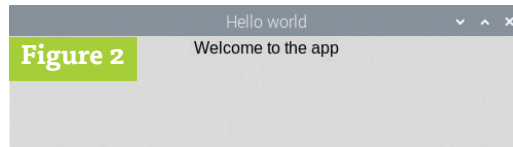
01-helloworld.py

DOWNLOAD DE VOLLEDIGE CODE:

> Language: **Python 3**

📄 magpi.cc/guizero

```
001. <from guizero import App, Text
002. app = App(title="Hello world")
003. message = Text(app, text="Welcome to the app")
004. app.display()
```



◀ **Figure 2** Add a text message

03 Widgets toevoegen

Widgets zijn de dingen die op de GUI verschijnen, zoals tekstvakken, knoppen, schuifregelaars en zelfs doodgewone stukjes tekst. Alle widgets komen tussen de regel code om de App te maken en de regel `app.display()`. Hier is de app die je net hebt gemaakt, maar in dit voorbeeld hebben we een tekst-widget toegevoegd:

```
from guizero import App, Text
app = App(title="Hello world")
message = Text(app, text="Welcome to the app")
app.display()
```

Is het je opgevallen dat er twee veranderingen zijn (**Figuur 2**)? Er is nu een extra regel code om de Text-widget toe te voegen, en we hebben ook Text toegevoegd aan de lijst van widgets om te importeren in de allereerste regel.

Laten we de code van de Text-widget wat beter bekijken:

```
message = Text(app, text="Welcome to the app")
```

Net als elke variabele in Python heeft een widget een naam nodig. Deze heet 'message'. Vervolgens geven we aan dat we willen dat dit een 'Text'-widget is. Tussen de haakjes staan enkele parameters om de Text-widget te vertellen hoe deze eruit moet zien. De eerste, 'app', vertelt de Text waar hij moet komen.

Alle widgets moeten in een container-widget zitten. Meestal zitten je widgets direct in een App, maar je zult later ontdekken dat er ook andere soorten container-widgets zijn waar je dingen in kunt zetten. Tot slot zeggen we dat de widget de tekst "Welcome to the app" moet bevatten.

Wanted-poster

01 Maak het spannender

Nu je een basis-GUI kunt maken, willen we het er wat spannender laten uitzien. Je kunt tekst in verschillende lettertypen, formaten en kleuren toevoegen, de achtergrondkleur veranderen en ook foto's toevoegen. Laten we, om dit allemaal te oefenen, een 'Wanted'-poster maken. Allereerst moet je beginnen met het maken van een app. Zet deze code in je editor om het meest basale app-venster te maken:

```
from guizero import App
app = App("Wanted!")
app.display()
```

Sla je code op en voer deze uit en je zou een app moeten zien die eruitziet als een gewoon grijs vierkant met de titel 'Wanted!' bovenin (**Figuur 3**).

02 Achtergrondkleuren

Laten we de achtergrond van de app een beetje anders maken. Traditioneel zien wanted-posters eruit alsof ze van perkament zijn gemaakt, dus laten we een lichtgele kleur toevoegen als achtergrond.

Zoek de coderegel waar je de app maakt. Voeg direct onder deze regel nog een regel code toe om de eigenschap `bg` van het venster aan te passen. `bg` is een afkorting voor 'background' en laat ons de kleur van de achtergrond veranderen. Nu moet je code er zo uitzien:

```
from guizero import App

app = App("Wanted!")
app.bg = "yellow"

app.display()
```

Je hebt nodig

- > Een computer (bijv. Raspberry Pi, Apple Mac, Windows- of Linux-pc)
- > Internetverbinding
- > Python 3 (python.org)
- > Een IDE (code editor), bijv: IDLE (geïnstalleerd met Python 3), Thonny (thonny.org), Mu (codewith.mu), PyCharm (jetbrains.com/pycharms)
- > De guizero Python-bibliotheek ([lawsie.github.io/guizero](https://github.com/lawsie/guizero))



► **Figuur 3** De basis-app

Dit wordt het bewerken van een eigenschap genoemd. In de code moet je de widget waar je het over hebt (app), de eigenschap die je wilt veranderen (bg) en de waarde die je wilt veranderen opgeven. Deze kleur (**Figuur 4**) is misschien iets te geel, laten we daarom de hex-code van een andere kleur geel opzoeken. Er zijn veel websites waar je kunt zoeken naar kleuren, je kunt bijvoorbeeld htmlcolorcodes.com proberen (**Figuur 5**).

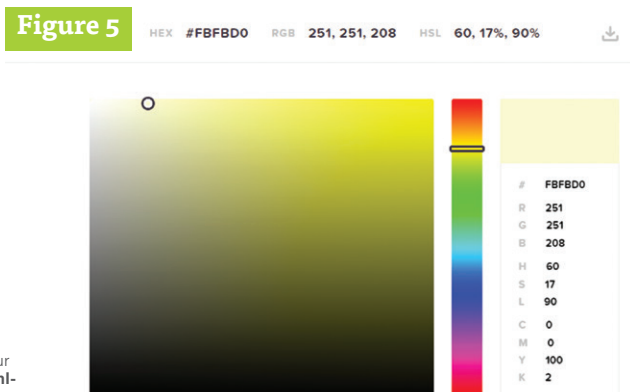
Wanneer je de gewenste kleur hebt geselecteerd, zie je de code ervan op de site verschijnen als hexadecimaal (in dit geval #FBFBDD) of als RGB (251, 251, 208). Je kunt beide formaten gebruiken voor het instellen van kleuren in guizero; je kunt bijvoorbeeld de code voor het geel maken van je achtergrond verwijderen en dan een van deze opties in je programma proberen:

```
app.bg = "#FBFBDD"
app.bg = (251, 251, 208)
```

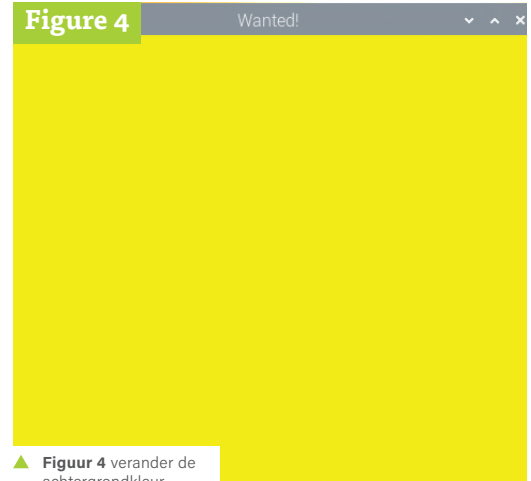
03 Tekst toevoegen

Je app zou er ongeveer zoals **Figuur 6** moeten uitzien. Laten we nu wat tekst toevoegen aan de GUI. We beginnen met het toevoegen van de tekst die alle goede wanted-posters nodig hebben - het woord 'Wanted'!

Zoek eerst de regel code die je al hebt, waar je de App hebt geïmporteerd.



► **Figuur 5** Een kleur selecteren op htmlcolorcodes.com



▲ **Figuur 4** verander de achtergrondkleur

```
from guizero import App
```

Je moet Text importeren om een stuk tekst te kunnen maken, dus voeg het toe aan het einde van de lijst. Nu ziet de regel er zo uit:

```
from guizero import App, Text
```

Telkens wanneer je een nieuw type widget wilt gebruiken, voeg je de naam toe aan het einde van de lijst. Het is niet nodig om hele nieuwe regels code toe te voegen: houd het gewoon op een lijst in één regel, zodat je programma niet te ingewikkeld wordt. Nu je tekst kunt gebruiken, willen we ook een stukje tekst toevoegen. Vergeet niet dat alle widgets op de GUI moeten worden toegevoegd tussen de regel code

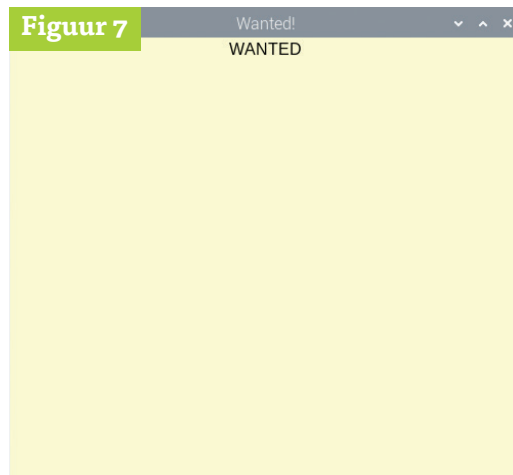
” bg is een afkorting voor ‘background’ en laat ons de kleur van de achtergrond veranderen ”

Lees de documentatie

Je vraagt je misschien af hoe je te weten komt welke eigenschappen een bepaalde widget heeft die je kunt veranderen. Zelfs als je een beginnend programmeur bent, is het de moeite waard om te leren hoe je documentatie moet lezen, omdat het je de volledige kracht van guizero en alle andere bibliotheken die je tegenkomt laat gebruiken. De guizero documentatie is te vinden op lawsie.github.io/guizero. Daar klik je op het menu Widgets en op de widget die je wilt veranderen, scroll naar beneden tot je bij de sectie Properties komt. Als je bijvoorbeeld de widget 'Text' selecteert dan zie je hier alle eigenschappen van een stuk Text die je eventueel kunt wijzigen. De documentatie bevat ook vaak nuttige stukjes code die je laten zien hoe je een bepaalde eigenschap of methode kunt gebruiken, dus wees niet bang om een kijkje te nemen – je weet nooit wat je kunt leren!



▲ **Figuur 6** Lichte achtergrond



◀ **Figuur 7** De tekst is te klein

waar je de App maakt en de regel code waar je deze weergeeft. Je code moet er nu zo uitzien:

```
from guizero import App, Text

app = App("Wanted!")
app.bg = "#FBFBD0"

wanted_text = Text(app, "WANTED")

app.display()
```

Hier is `wanted_text` de naam van het stuk tekst. Dit is zodat we er later in de code over kunnen praten – denk aan de naam van een persoon. (Je zou je stuk tekst zelfs Dave kunnen noemen als je dat wilt – voor de computer maakt het niet uit).

Tussen de haakjes hebben we twee dingen. Het tweede, "WANTED", is eenvoudig omdat het de tekst is die we op het scherm willen weergeven. Het eerste is de container die dit stukje tekst bestuurt, die zijn 'master' wordt genoemd. In dit geval zeggen we dat deze tekst door de app moet worden bestuurd. Wanneer je voor het eerst start met het maken van GUI's, zullen de meeste van je widgets de app als hun 'master' hebben, maar er zijn andere containers die widgets kunnen opslaan en die je later zult leren kennen.

04 Wijzig grootte en kleur van tekst

Oh oh, deze tekst is vrij klein (Figuur 7). Laten we de eigenschap `text_size` op precies dezelfde manier veranderen als toen we de achtergrondkleur van de app veranderden. Vergeet niet dat je drie dingen moest specificeren:

1. De naam van de widget
2. De eigenschap die veranderd moet worden
3. De nieuwe waarde die hij moet krijgen

Dus, in dit geval geef je de widget (`wanted_size`), de te wijzigen eigenschap (`text_size`) en de nieuwe waarde (`50`) op. Voeg een nieuwe regel code toe direct onder

de regel waar je de tekst hebt aangemaakt, om de eigenschap te wijzigen.

```
wanted_text = Text(app, "WANTED")
wanted_text.text_size = 50
```

Je hebt nu een grotere tekst op je poster (Figuur 8). Kijk of je nu het lettertype van deze tekst kunt veranderen in iets anders. Welke lettertypes beschikbaar zijn hangt af van welk besturingssysteem je gebruikt, dus hier zijn enkele suggesties:

- ▶ Times New Roman
- ▶ Verdana
- ▶ Courier
- ▶ Impact

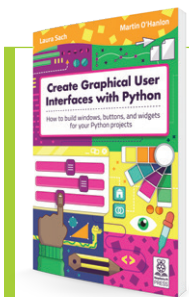
Geen enkele 'wanted'-poster zou compleet zijn zonder foto, dus laten we er een toevoegen. Onze poster is voor de kat van de auteur, omdat zij altijd aan dingen krabt waar ze niet aan mag krabben.

Sla een kopie van de foto die je wilt gebruiken op in dezelfde map als je GUI-programma. Je kunt afbeeldingen in andere mappen gebruiken, maar als je dat doet zul je het pad naar de afbeelding moeten aangeven, dus het is een stuk makkelijker om ze in het begin gewoon in dezelfde map op te slaan.

Hopelijk ben je nu gewend aan het toevoegen van widgets. Vergeet niet dat ze altijd aan het begin van het programma moeten worden geïmporteerd, en vervolgens komt de widget die wordt aangemaakt met een sprekende naam onder de regel met code waarmee je de App maakt, maar boven de afsluitende regel



▲ **Figuur 8** Grotere tekst



Create Graphical User Interfaces with Python

Voor meer tutorials over het maken van je eigen GUI's met guizero, kun je terecht in ons nieuwe boek, *Create Graphical User Interfaces with Python*. De 156 pagina's staan vol met essentiële informatie en een reeks spannende projecten. magpi.cc/pythongui

▶ **Figuur 8** Grotere tekst

► **Figuur 9** De voltooide poster



met `app.display()`.
Voeg 'Picture' toe aan de lijst van de te importeren widgets aan het begin van het programma.

02-wanted.py

> Language: **Python 3**

DOWNLOAD DE VOLLEDIGE CODE:

 magpi.cc/guizero-code

```
001. from guizero import App, Text, Picture
002.
003. app = App("Wanted!")
004. app.bg = "#FBFBF0"
005.
006. wanted_text = Text(app, "WANTED")
007. wanted_text.text_size = 50
008. wanted_text.font = "Times New Roman"
009.
010. cat = Picture(app, image="tabitha.png")
011. app.display()
```

Beeldmanipulatie

Omdat guizero een bibliotheek is voor beginners en we de installatie zo eenvoudig mogelijk wilden maken, zitten er geen verfijnde beeldbewerkingsfuncties in, omdat hiervoor een extra bibliotheek nodig is die 'pillow' heet. Je kunt altijd niet-geanimeerde GIF-plaatjes op elk platform gebruiken, en PNG-plaatjes op alle platforms behalve Mac, dus als je niet zeker weet of je de extra beeldbewerkingsfuncties hebt geïnstalleerd, houd het dan op die twee bestandstypes.

```
from guizero import App, Text, Picture
```

Maak nu een Picture-widget met twee parameters: de app en de bestandsnaam van de afbeelding. Dit is de code die we hebben gebruikt omdat onze foto **tabitha.png** heet.

```
cat = Picture(app, image="tabitha.png")
```

Voer je code (die moet lijken op 02-wanted.py) opnieuw uit en je zou de afbeelding onder je tekst moeten zien (Figuur 9). Nu is het aan jou om je nieuwe GUI-vaardigheden te gebruiken en de poster vorm te geven zoals je maar wilt. ◀

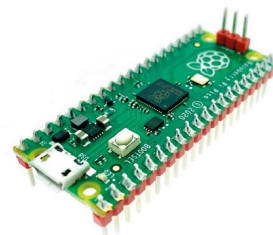
210419-03

Noot van de redactie: dit artikel verscheen oorspronkelijk in MagPi tijdschrift. Elektor publiceert de Nederlandse, Franse en Duitse edities van MagPi.

WORD ABONNEE EN ONTVANG

GRATIS Raspberry Pi + Headers

IEDERE 2 MAANDEN HET LAATSTE RASPBERRY PI NIEUWS EN DE ALLERLEUKSTE PROJECTEN!



Uw voordelen:

- Gratis Raspberry Pi + Headers
- Goedkoper dan zes losse nummers
- Iedere editie in de brievenbus; je hoeft de deur niet uit
- Elk nummer ook digitaal beschikbaar (PDF)
- Je leest MagPi al voordat het blad in de winkel ligt



ABONNEER NU OP WWW.MAGPI.NL