

Een grafisch interface voor de Raspberry Pi met tkinter.

Voor het maken van een grafische gebruikers omgevingen is er een tool set TCL/TK.

De implementatie voor Python 3.x is tkinter.

Om aan de tkinter workshop mee te kunnen doen moet je de RPi upgraden naar de laatste versie van Raspbian Jessie of Stretch.

Start de terminal en gebruik de volgende opdrachten.
sudo apt-get update
sudo apt-get upgrade

Een grafisch interface voor de Raspberry Pi met tkinter.

Wat kun je met tkinter doen?

- Een bedieningsvenster maken voor een programma.
- Het venster voorzien van bedieningselementen.
- De bedieningselementen koppelen aan een uit te voeren programma.
- De layout van het venster inrichten naar eigen smaak.

Een grafisch interface voor de Raspberry Pi met tkinter.

Widgets.

Elementen waaruit het user interface is opgebouwd.

Tkinter Widgets







Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

There are currently 15 types of widgets in Tkinter. We present these widgets as well as a brief description in the following table –

Een grafisch interface voor de Raspberry Pi met tkinter.

Operator	Description
Button ↗	The Button widget is used to display buttons in your application.
Canvas ↗	The Canvas widget is used to draw shapes, such as lines, ovals, polygons and rectangles, in your application.
Checkbutton ↗	The Checkbutton widget is used to display a number of options as checkboxes. The user can select multiple options at a time.
Entry ↗	The Entry widget is used to display a single-line text field for accepting values from a user.
Frame ↗	The Frame widget is used as a container widget to organize other widgets.

Een grafisch interface voor de Raspberry Pi met tkinter.

Label 	The Label widget is used to provide a single-line caption for other widgets. It can also contain images.
Listbox 	The Listbox widget is used to provide a list of options to a user.
Menubutton 	The Menubutton widget is used to display menus in your application.
Menu 	The Menu widget is used to provide various commands to a user. These commands are contained inside Menubutton.
Message 	The Message widget is used to display multiline text fields for accepting values from a user.
Radiobutton 	The Radiobutton widget is used to display a number of options as radio buttons. The user can select only one option at a time.

Een grafisch interface voor de Raspberry Pi met tkinter.

Scale ↗	The Scale widget is used to provide a slider widget.
Scrollbar ↗	The Scrollbar widget is used to add scrolling capability to various widgets, such as list boxes.
Text ↗	The Text widget is used to display text in multiple lines.
Toplevel ↗	The Toplevel widget is used to provide a separate window container.
Spinbox ↗	The Spinbox widget is a variant of the standard Tkinter Entry widget, which can be used to select from a fixed number of values.
PanedWindow ↗	A PanedWindow is a container widget that may contain any number of panes, arranged horizontally or vertically.
LabelFrame ↗	A labelframe is a simple container widget. Its primary purpose is to act as a spacer or container for complex window layouts.
tkMessageBox ↗	This module is used to display message boxes in your applications.

Een grafisch interface voor de Raspberry Pi met tkinter.

Standard attributes

Let us take a look at how some of their common attributes, such as sizes, colors and fonts are specified.

- [Dimensions](#) ↗
- [Colors](#) ↗
- [Fonts](#) ↗
- [Anchors](#) ↗
- [Relief styles](#) ↗
- [Bitmaps](#) ↗
- [Cursors](#) ↗

Een grafisch interface voor de Raspberry Pi met tkinter.

Een uitstekend referentie document is te vinden op:

<http://infohost.nmt.edu/tcc/help/pubs/tkinter/web/index.html>

Een serie van 15 tutorials op YouTube van Bucky Roberts leren je stap voor stap de ins en outs van het maken van een grafische gebruikers omgeving. De video's duren gemiddeld 5 tot 7 minuten en behandelen de onderwerpen in een logische volgorde.

https://www.youtube.com/watch?v=RJB1Ek2Ko_Y

Een grafisch interface voor de Raspberry Pi met tkinter.

Benodigdheden voor het volgen van de workshop.

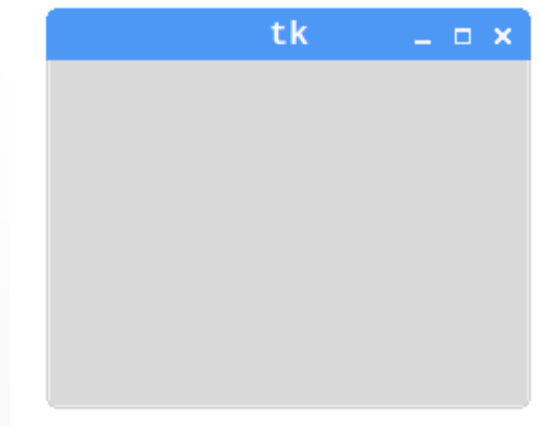
- Raspberry Pi 2 of 3 met de laatste update van Raspbian OS.
- Python 3.5 of 3.6 met de module tkinter geïnstalleerd.
- Thonny IDE 2.1.10 of Python IDLE 3.5 of 3.6
- Display, muis en toetsenbord.

Een grafisch interface voor de Raspberry Pi met tkinter.

We beginnen de workshop met het basis venster.
Start de Thonny IDE of Python IDLE op.
Voer het onderstaande script in en voer het uit met F5

Het basis venster.

```
# Basis venster met tkinter  
  
from tkinter import *  
venster = Tk()  
venster.mainloop()
```



Een grafisch interface voor de Raspberry Pi met tkinter.

Het venster een titel geven en vaste afmetingen geven.

```
#Basis venster met tkinter

from tkinter import *
venster = Tk()
venster.title("tkinter basis venster")
venster.geometry("450x300")
venster.mainloop
```



Een grafisch interface voor de Raspberry Pi met tkinter.

De achtergrondkleur van het venster aanpassen.

```
# Achtergrondkleur van het venster aanpassen

from tkinter import *
venster = Tk()
venster.title("venster met achtergrondkleur")
venster.configure( background = "yellow")
venster.geometry("450x300+400+400")
venster.mainloop()
```



Met de `.configure` functie kunnen we ondermeer de achtergrondkleur aan passen.

Een grafisch interface voor de Raspberry Pi met tkinter.

Venster positie op het scherm, en een label.

```
#Basis venster met tkinter

from tkinter import *
venster = Tk()
venster.title("tkinter basis venster")
venster.geometry("450x300+400+400")
label1 = Label(venster, text="Dit is label 1")
label1.pack()
venster.mainloop()
```



Geometry: X x Y in pixels.
Origin is altijd in linker bovenhoek
van het scherm.

Offset t.o.v. linkerbovenhoek van
het scherm: +X en +Y
De functie .pack()
plaatst het label in het venster.

Een grafisch interface voor de Raspberry Pi met tkinter.

Label () en .pack() attributen.

#Basis venster met tkinter

```
from tkinter import *
venster = Tk()
venster.title("tkinter basis venster")
venster.geometry("450x300+400+400")
label1 = Label(venster, text="Dit is label 1", bg = "dark grey", fg = "white")
label2 = Label(venster, text = "Dit is label 2", bg = "yellow", fg = "red")
label1.pack(side = LEFT)
label2.pack()
venster.mainloop()
```

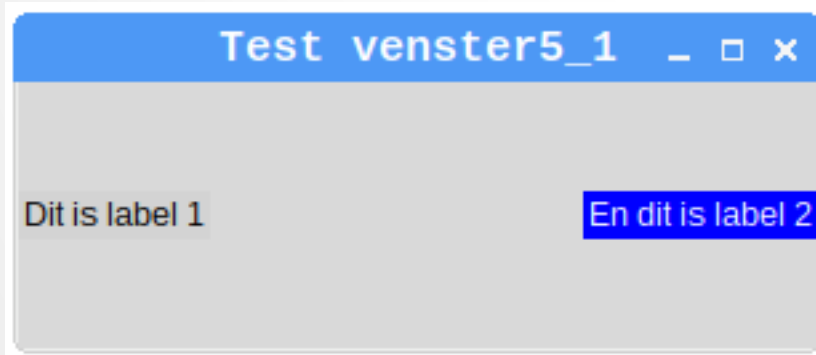


Het label widget kan met verschillende argumenten vorm worden gegeven. Naast de tekst zijn dat vorm, voor- en achtergrond kleuren. positie en reliëf.

Een grafisch interface voor de Raspberry Pi met tkinter.

**Label positioneren met attributen left, right, top en bottom.
De labels worden altijd halverwege het venster geplaatst.**

```
from tkinter import*
venster5=Tk()
venster5.title("Test venster5_1")
venster5.geometry("300x150+400+250")
label1=Label(venster5, text= "Dit is label 1", bg="light grey", fg="black")
label1.pack(side="left")
label2=Label(venster5, text= "En dit is label 2", bg="blue", fg= "white")
label2.pack(side="right")
venster5.mainloop()
```



Een grafisch interface voor de Raspberry Pi met tkinter.

Color Chart to be Used with Python and TKinter

The chart below contains the names recognized by TKinter, and the Hortsmann graphics library for identifying different colors. The python program that follows is used to generate the chart.

Named colour chart												
snow	deep sky blue	gold	seashell3	SlateBlue2	LightBlue3	SpringGreen2	DarkGoldenrod1	brown4	pink3	purple1	gray26	gray64
ghost white	sky blue	light goldenrod	seashell4	SlateBlue3	LightBlue4	SpringGreen3	DarkGoldenrod2	salmon1	pink4	purple2	gray27	gray65
white smoke	light sky blue	goldenrod	AntiqueWhite1	SlateBlue4	LightCyan2	SpringGreen4	DarkGoldenrod3	salmon2	LightPink1	purple3	gray28	gray66
gainsboro	steel blue	dark goldenrod	AntiqueWhite2	RoyalBlue1	LightCyan3	green2	DarkGoldenrod4	salmon3	LightPink2	purple4	gray29	gray67
floral white	light steel blue	rosy brown	AntiqueWhite3	RoyalBlue2	LightCyan4	green3	RosyBrown1	salmon4	LightPink3	MediumPurple1	gray30	gray68
old lace	light blue	indian red	AntiqueWhite4	RoyalBlue3	PaleTurquoise1	green4	RosyBrown2	LightSalmon2	LightPink4	MediumPurple2	gray31	gray69
linen	powder blue	saddle brown	bisque2	RoyalBlue4	PaleTurquoise2	chartreuse2	RosyBrown3	LightSalmon3	PaleVioletRed1	MediumPurple3	gray32	gray70
antique white	pale turquoise	sandy brown	bisque3	blue	PaleTurquoise3	chartreuse3	RosyBrown4	LightSalmon4	PaleVioletRed2	MediumPurple4	gray33	gray71
papaya whip	dark turquoise	dark salmon	bisque4	blue2	PaleTurquoise4	chartreuse4	IndianRed1	orange2	PaleVioletRed3	thistle1	gray34	gray72
blanched almond	medium turquoise	salmon	PeachPuff2	DodgerBlue2	CadetBlue1	OliveDrab1	IndianRed2	orange3	PaleVioletRed4	thistle2	gray35	gray73
bisque	turquoise	light salmon	PeachPuff3	DodgerBlue3	CadetBlue2	OliveDrab2	IndianRed3	orange4	maroon1	thistle3	gray36	gray74
peach puff	cyan	orange	PeachPuff4	DodgerBlue4	CadetBlue3	OliveDrab3	IndianRed4	DarkOrange1	maroon2	thistle4	gray37	gray75
navajo white	light cyan	dark orange	NavajoWhite2	SteelBlue1	CadetBlue4	DarkOliveGreen1	sienna1	DarkOrange2	maroon3		gray38	gray76
lemon chiffon	cadet blue	coral	NavajoWhite3	SteelBlue2	turquoise1	DarkOliveGreen2	sienna2	DarkOrange3	maroon4		gray39	gray77
mint cream	medium aquamarine	light coral	NavajoWhite4	SteelBlue3	turquoise2	DarkOliveGreen3	sienna3	DarkOrange4	VioletRed1		gray40	gray78
azure	aquamarine	tomato	LemonChiffon2	SteelBlue4	turquoise3	DarkOliveGreen4	sienna4	coral1	VioletRed2		gray42	gray79
alice blue	dark green	orange red	LemonChiffon3	DeepSkyBlue2	turquoise4	khaki1	burlywood1	coral2	VioletRed3		gray43	gray80
lavender	dark olive green	red	LemonChiffon4	DeepSkyBlue3	cyan2	khaki2	burlywood2	coral3	VioletRed4		gray44	gray81
lavender blush	dark sea green	hot pink	cornsilk2	DeepSkyBlue4	cyan3	khaki3	burlywood3	coral4	magenta2		gray45	gray82
misty rose	sea green	deep pink	cornsilk3	SkyBlue1	cyan4	khaki4	burlywood4	tomato2	magenta3		gray46	gray83
dark slate gray	medium sea green	pink	cornsilk4	SkyBlue2	DarkSlateGray1	LightGoldenrod1	wheat1	tomato3	magenta4		gray47	gray84
dim gray	light sea green	light pink	ivory2	SkyBlue3	DarkSlateGray2	LightGoldenrod2	wheat2	tomato4	orchid1		gray48	gray85
slate gray	pale green	pale violet red	ivory3	SkyBlue4	DarkSlateGray3	LightGoldenrod3	wheat3	OrangeRed2	orchid2		gray49	gray86
light slate gray	spring green	maroon	ivory4	LightSkyBlue1	DarkSlateGray4	LightGoldenrod4	wheat4	OrangeRed3	orchid3		gray50	gray87
gray	lawn green	medium violet red	honeydew2	LightSkyBlue2	aquamarine2	LightYellow2	tan1	OrangeRed4	orchid4		gray51	gray88
light grey	medium spring green	violet red	honeydew3	LightSkyBlue3	aquamarine4	LightYellow3	tan2	red2	plum1		gray52	gray89
medium blue	green yellow	medium orchid	honeydew4	LightSkyBlue4	DarkSeaGreen1	LightYellow4	tan4	red3	plum2		gray53	gray90
blue	lime green	dark orchid	LavenderBlush2	SlateGray1	DarkSeaGreen2	yellow2	chocolate1	red4	plum3		gray54	gray91
cornflower blue	yellow green	dark violet	LavenderBlush3	SlateGray2	DarkSeaGreen3	yellow3	chocolate2	DeepPink2	plum4		gray55	gray92
dark slate blue	forest green	blue violet	LavenderBlush4	SlateGray3	DarkSeaGreen4	yellow4	chocolate3	DeepPink3	MediumOrchid1		gray56	gray93
slate blue	olive drab	purple	MistyRose2	SlateGray4	SeaGreen1	gold2	firebrick1	DeepPink4	MediumOrchid2		gray57	gray94
medium slate blue	dark khaki	medium purple	MistyRose3	LightSteelBlue1	SeaGreen2	gold3	firebrick2	HotPink1	MediumOrchid3		gray58	gray95
light slate blue	khaki	thistle	MistyRose4	LightSteelBlue2	SeaGreen3	gold4	firebrick3	HotPink2	MediumOrchid4		gray59	gray97
medium sea blue	pale goldenrod	snow2	azure2	LightSteelBlue3	PaleGreen1	goldenrod1	firebrick4	HotPink3	DarkOrchid1		gray60	gray98
royal blue	light goldenrod yellow	snow3	azure3	LightSteelBlue4	PaleGreen2	goldenrod2	brown1	HotPink4	DarkOrchid2		gray61	gray99
blue2	light yellow	snow4	azure4	LightBlue1	PaleGreen3	goldenrod3	brown2	pink1	DarkOrchid3		gray62	gray62
dodger blue	yellow	seashell2	SlateBlue1	LightBlue2	PaleGreen4	goldenrod4	brown3	pink2	DarkOrchid4		gray63	gray63

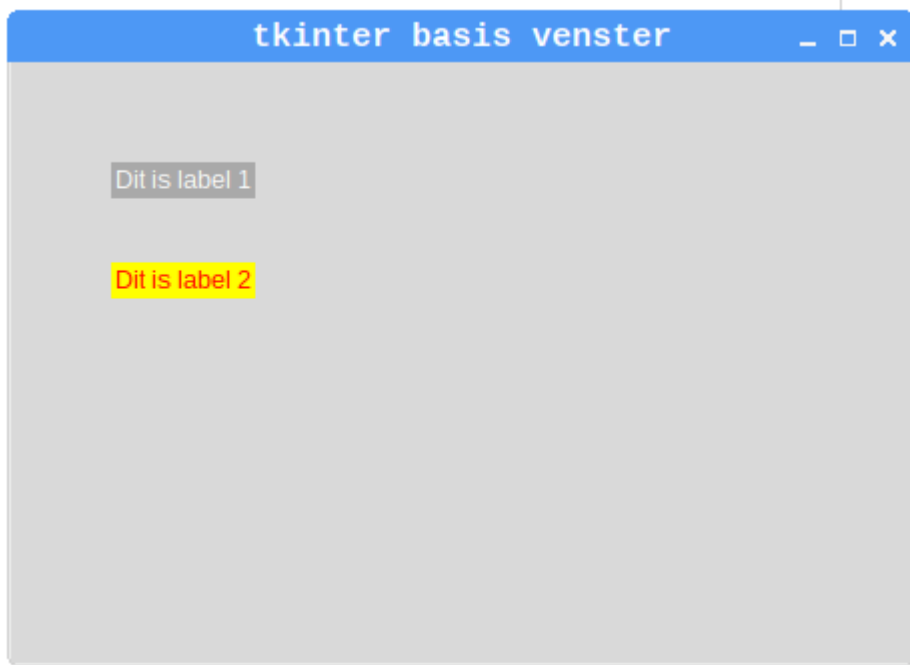
Kleurenkaart met de namen van de kleuren die in tkinter gebruikt kunnen worden.

Een grafisch interface voor de Raspberry Pi met tkinter.

De `.place()` functie maakt nauwkeurig plaatsen van widgets mogelijk.

#Basis venster met tkinter

```
from tkinter import *
venster = Tk()
venster.title("tkinter basis venster")
venster.geometry("450x300+400+400")
label1 = Label(venster, text="Dit is label 1", bg = "dark grey", fg = "white")
label2 = Label(venster, text = "Dit is label 2", bg = "yellow", fg = "red")
label1.place( x = 50, y = 50)
label2.place ( x = 50, y = 100)
venster.mainloop()
```



De `place` functie geeft meer mogelijkheden om je widgets te positioneren in het venster.

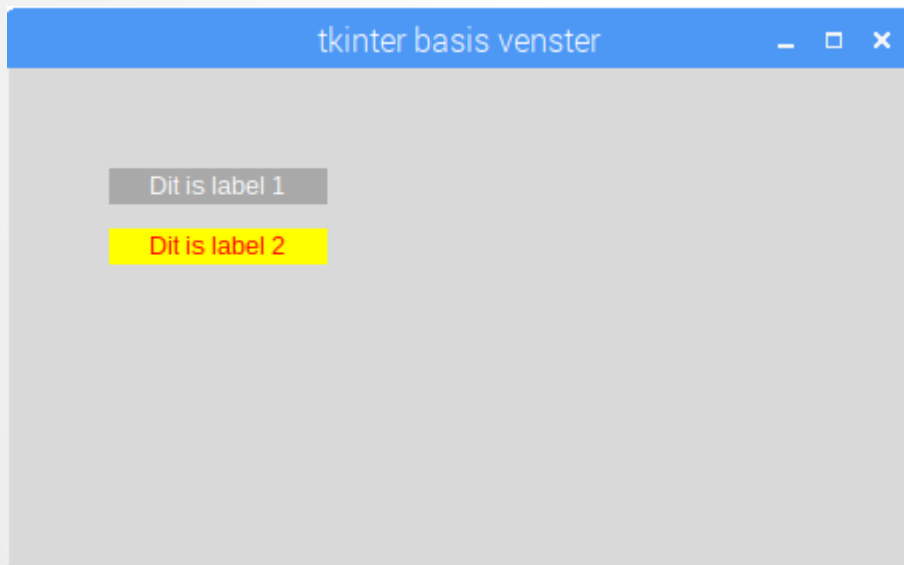
Je kunt d.m.v. `x` en `y` coördinaten de juiste positie bepalen t.o.v. de linkerbovenhoek van het venster.

Een grafisch interface voor de Raspberry Pi met tkinter.

De afmetingen van een label instellen

```
from tkinter import *
venster = Tk()
venster.title("tkinter basis venster")
venster.geometry("450x250+400+400")
label1 = Label(venster, text="Dit is label 1", bg = "dark grey", fg = "white", width = 15)
label2 = Label(venster, text = "Dit is label 2", bg = "yellow", fg = "red", width = 15)
label1.place( x = 50, y = 50)
label2.place ( x = 50, y = 80)

venster.mainloop()
```



Met het width argument kun je de breedte van een label instellen in een aantal Monospace karakters.

Een grafisch interface voor de Raspberry Pi met tkinter.

Widgets plaatsen met de .grid functie.

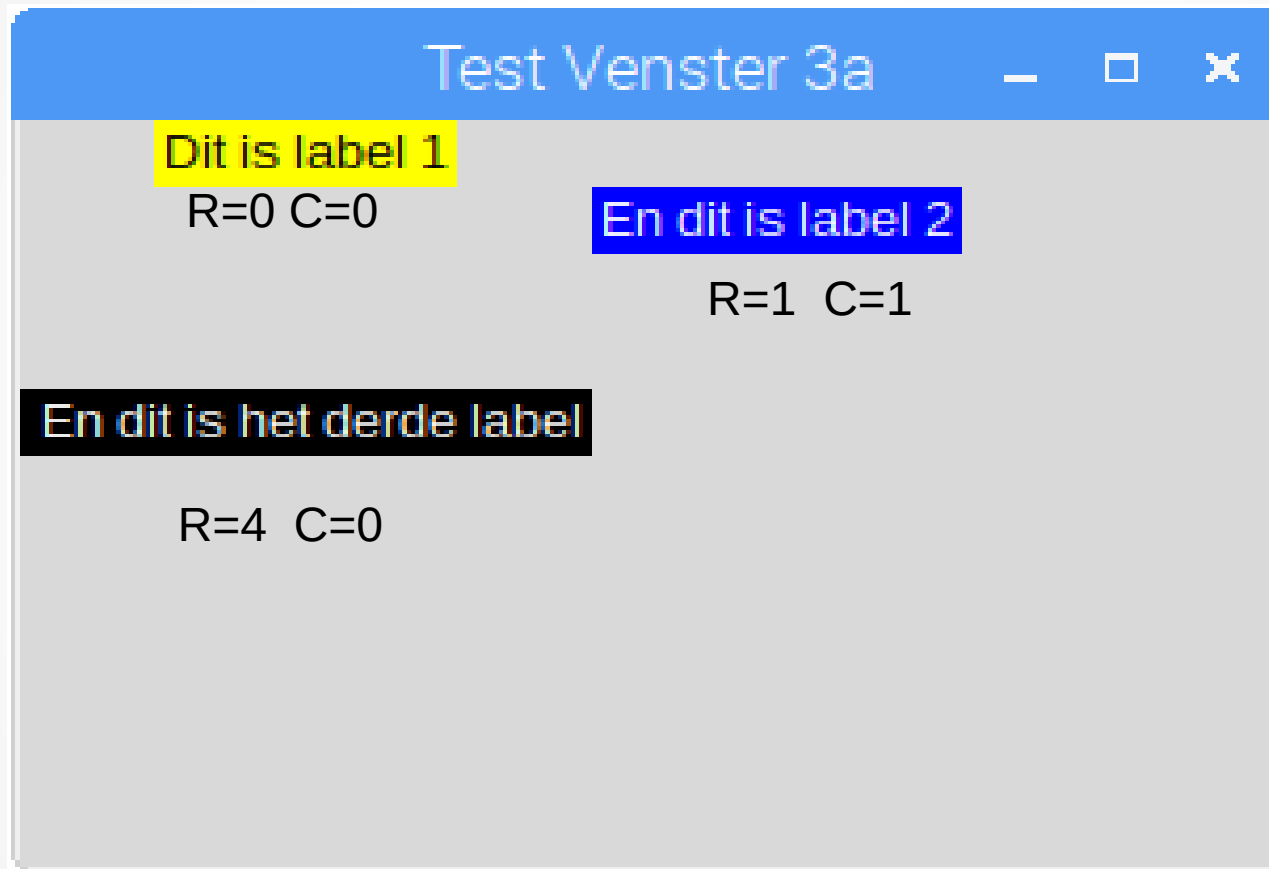
```
# een voorbeeld van een andere plaatsing met het .grid commando

from tkinter import*
venster5=Tk()
venster5.title("Test Venster 3a")
venster5.geometry("300x200+200+150")
label1=Label(venster5,text= "Dit is label 1",bg="yellow",fg="black")
label1.grid(row=0, column=0)
label2=Label(venster5,text= "En dit is label 2",bg="blue", fg= "white")
label2.grid(row=1, column=1)
label3=Label(venster5, text= " ")
label3.grid ( row= 2, columnspan =1)
label4=Label(venster5, text= " ")
label4.grid (row = 3, columnspan =1)
label5=Label(venster5,text= " En dit is het derde label", bg="black", fg= "white")
label5.grid(row=4, column=0)

venster5.mainloop

# de column breedte wordt aan gepast aan het label met de langste tekst
```

Een grafisch interface voor de Raspberry Pi met tkinter.

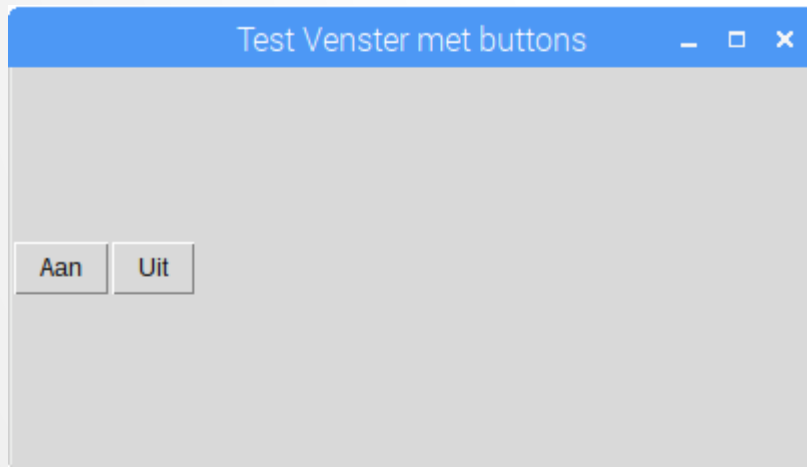


Lege labels 3 en 4 zijn niet zichtbaar
Maar worden gebruikt om verticale ruimte
Te creeren tussen label2 en label3.

Een grafisch interface voor de Raspberry Pi met tkinter.

Buttons

```
from tkinter import*
venster5 = Tk()
venster5.title("Test Venster met buttons")
venster5.geometry("400x200+200+150")
b1 = Button(venster5, text = "Aan")
b1.pack(side= "left")
b2 = Button(venster5, text = "Uit")
b2.pack(side = "left")
venster5.mainloop
```



We hebben twee toetsen gemaakt, b1 en b2. Met de .pack functie zijn ze gepositioneerd midden links van het venster.

Een grafisch interface voor de Raspberry Pi met tkinter.

```
from tkinter import*
venster5 = Tk()
venster5.title("Test Venster met buttons")
venster5.geometry("400x200+200+150")
fr = Frame(venster5)
fr.pack()
lbl=Label(venster5, text="Het label komt bovenaan")
lbl.pack()
b1 = Button(venster5, text = "Aan")
b1.pack(side= "left", padx = 20)
b2 = Button(venster5, text = "Uit")
b2.pack(side = "left", padx = 20)
venster5.mainloop
```



Met padx en pady argumenten
Maken we ruimte om een widget.
Standaard is dat in pixels. Zie de
reference voor andere maat systemen

Een grafisch interface voor de Raspberry Pi met tkinter.

```
from tkinter import*
venster5 = Tk()
venster5.title("Test Venster met buttons")
venster5.geometry("400x200+200+150")
lbl=Label(venster5, text="Het label komt bovenaan")
lbl.pack()
b1 = Button(venster5,text = "Aan", relief = RAISED, borderwidth = 3)
b1.pack(side= "left", padx = 20)
b2 = Button(venster5,text = "Uit", relief = RAISED, borderwidth = 3)
b2.pack(side = "left",padx = 20)
venster5.mainloop
```



Buttons voorzien van relief en waarbij de randdikte 3 is ingesteld.

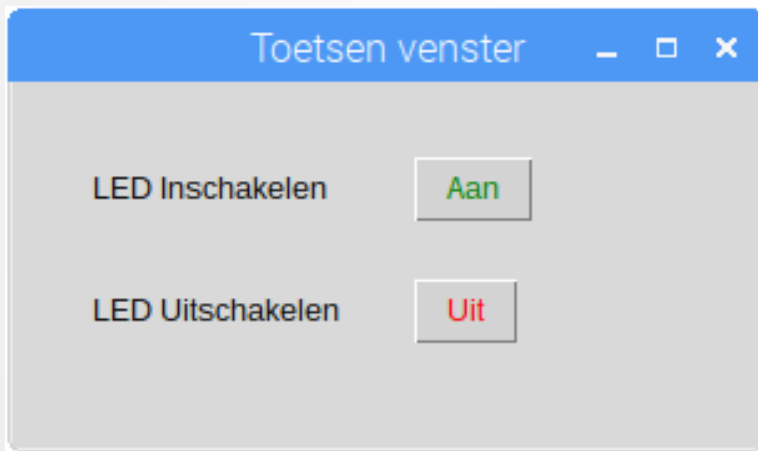
Een grafisch interface voor de Raspberry Pi met tkinter.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup (12, GPIO.OUT)

def led_aan():
    GPIO.output(12,1)

def led_uit():
    GPIO.output(12,0)

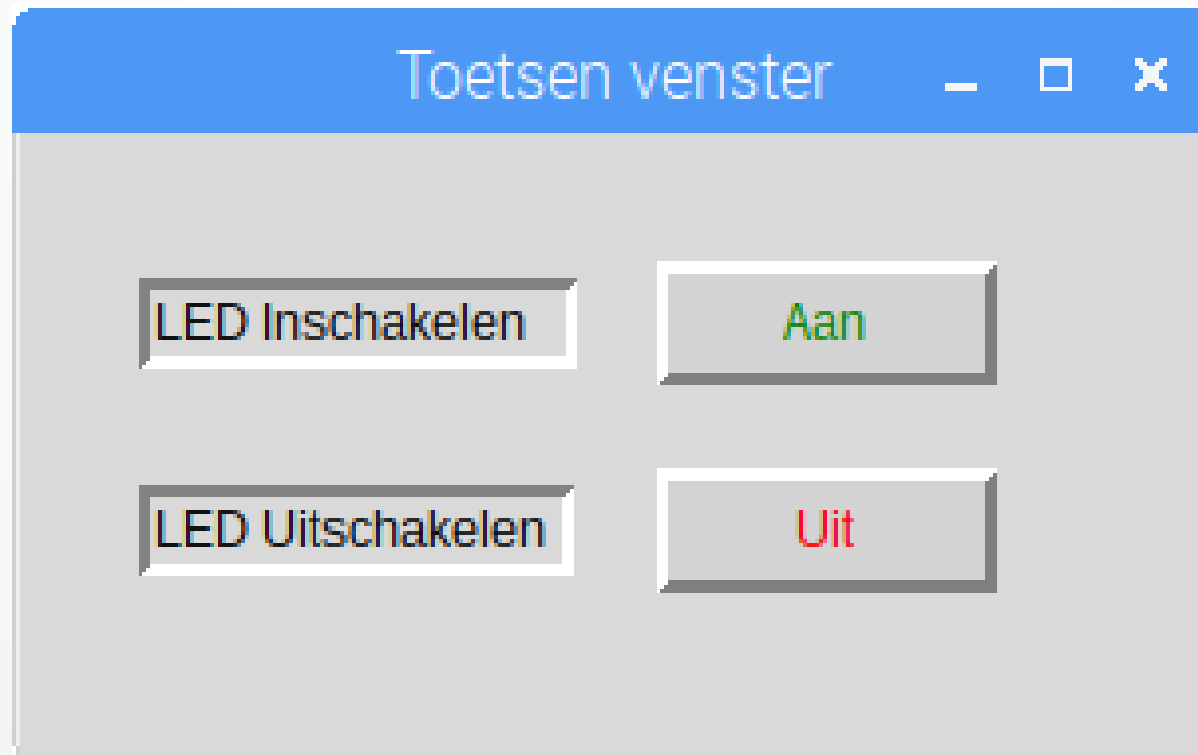
from tkinter import *
venster = Tk()
venster.title("Toetsen venster")
venster.geometry("300x150+400+350")
label1 = Label(text = "LED Inschakelen").place( x = 30, y = 35)
label2 = Label(text = "LED Uitschakelen ").place( x = 30, y = 85)
button_aan = Button(text = "Aan", bg ="light grey",fg = "green",command = led_aan).place( x = 160, y = 30)
button_uit = Button(text = "Uit", bg ="light grey",fg = "red",command = led_uit).place( x = 160, y = 80)
venster.mainloop
```



Een venster met labels die de functie van de toetsen aangeeft.

Een grafisch interface voor de Raspberry Pi met tkinter.

Een venster waarbij voor de labels een verdiept relief is ingesteld en voor de toetsen een verhoogd relief. Aan de toetsen is een eenvoudige functie gekoppeld. Door de **Aan** toets wordt een led ingeschakeld, de **Uit** toets doet het tegenovergestelde. Het script staat op de volgende dia.



Een grafisch interface voor de Raspberry Pi met tkinter.

Led met serieweerstand van 330 ohm aansluiten op pin 12 en 6 van de GPIO.

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup (12, GPIO.OUT)

def led_aan():
    GPIO.output(12,1)

def led_uit():
    GPIO.output(12,0)

from tkinter import *
venster = Tk()
venster.title("Toetsen venster")
venster.geometry("300x150+400+350")
label1 = Label(text = "LED Inschakelen  ", relief = SUNKEN, borderwidth = 3).place( x = 30, y = 35)
label2 = Label(text = "LED Uitschakelen ", relief = SUNKEN, borderwidth = 3).place( x = 30, y = 85)

button_aan = Button(text = "Aan", bg ="light grey",fg = "green",width = 8,relief = RAISED,
borderwidth = 3,command = led_aan).place( x = 160, y = 30)

button_uit = Button(text = "Uit", bg ="light grey",fg = "red",width = 8,relief = RAISED,
borderwidth = 3,command = led_uit).place( x = 160, y = 80)
venster.mainloop
```

Een grafisch interface voor de Raspberry Pi met tkinter.

GPIO besturing met twee leds als indicators



Een grafisch interface voor de Raspberry Pi met tkinter.

```
#Besturing rood - groen led
```

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings (False)
GPIO.setup(12, GPIO.OUT)
GPIO.setup(16, GPIO.OUT)
```

```
def rood_a():
    GPIO.output(16, 1)
```

```
def rood_u():
    GPIO.output(16, 0)
```

```
def groen_a():
    GPIO.output(12, 1)
```

```
def groen_u():
    GPIO.output(12, 0)
```

```
from tkinter import *
```

```
venster = Tk()
```

```
venster.title ("Led besturing Rood/Groen")
```

```
venster.geometry("350x200+300+300")
```

```
button_rood_a = Button( text = "Rood Aan",bg ="light grey",fg = "Red",width = 10, relief= RAISED, borderwidth = 4, command = rood_a)
```

```
button_rood_a.place(x = 50 ,y = 40)
```

```
button_rood_u = Button( text = "Rood Uit",bg ="light grey", fg ="Red",width = 10, relief = RAISED, borderwidth = 4, command = rood_u)
```

```
button_rood_u.place( x = 50, y = 120)
```

```
button_groen_a = Button( text = "Groen Aan", bg ="light grey", fg = "Green",width = 10, relief = RAISED, borderwidth = 4, command = groen_a)
```

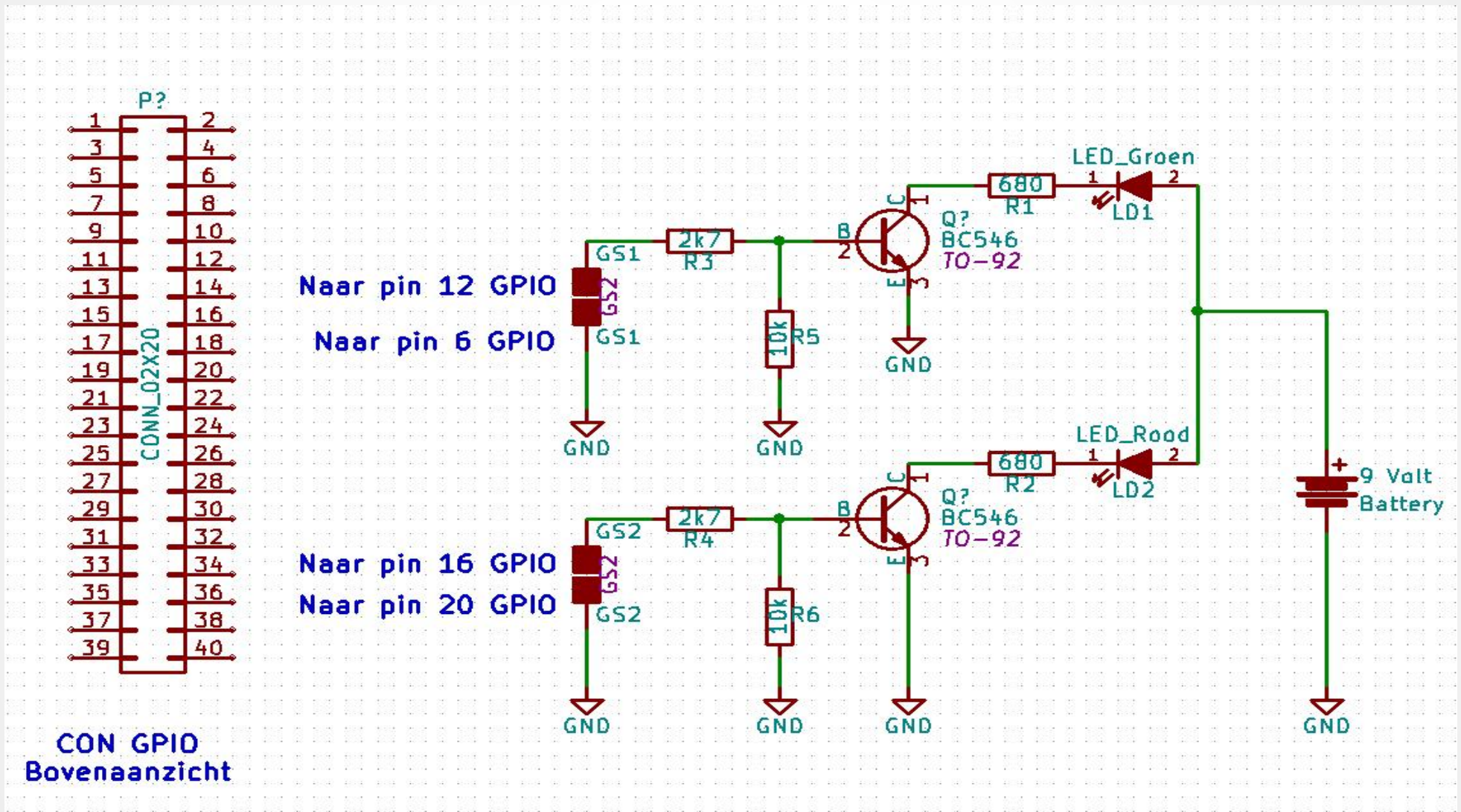
```
button_groen_a.place( x = 200, y = 40)
```

```
button_groen_u = Button( text = "Groen Uit", bg ="light grey", fg = "Green",width = 10, relief = RAISED, borderwidth = 4, command = groen_u)
```

```
button_groen_u.place( x = 200, y = 120)
```

```
venster.mainloop
```

Een grafisch interface voor de Raspberry Pi met tkinter.



Led indicator circuit.