



In deze les gebruiken we de TI-Innovator Hub in een Python programma.

Doelen :

- Het verkennen van de ti_hub module.
- Het toepassen van enkele basisvaardigheden.

In de ti_hub module vinden we de opdrachten om vanuit Python met de TI-Innovator Hub te communiceren.

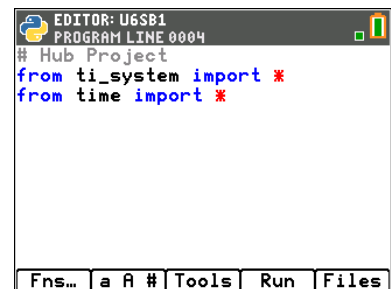
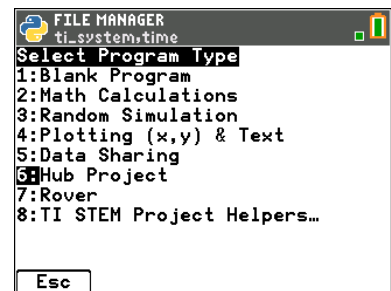
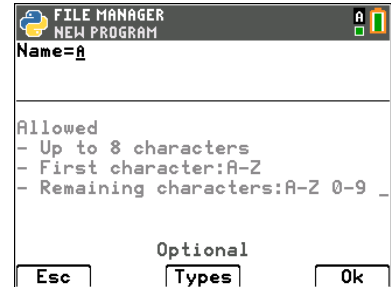
We gaan een programma maken dat de gekleurde led in de Hub laat branden en dat de ingebouwde luidspreker gebruikt.

Meestal heb je bij programma's die de Hub gebruiken extra opdrachten nodig die in andere modules zitten. Daarom is het handig om die modules alvast in het programma in te voegen.

Een makkelijke manier hiervoor is bij het maken van een nieuw programma te kiezen voor een bepaald type.

Open een nieuw programma en kies bij het scherm waar je de naam moet intypen voor het type **Hub Project** met F3(Types).

Je ziet dan in de editor dat er twee modules zijn geladen.



Tip voor de docent: Je kunt de modules die nu zijn geladen ook altijd nog achteraf toevoegen.

Als je de ingebouwde led van de Hub wilt gebruiken moet je de opdrachten hiervoor apart toevoegen. Dat kan met het ti_hub menu.

Kies » en dan 6:ti_hub...

De ingebouwde led hoort bij de Built-in devices.
Selecteer deze en kies dan voor Color (de gekleurde led).


In de editor staat nu dat deze is geïmporteerd.
Dit betekent dat de opdracht is toegevoegd aan het menu.

Als je nu op » drukt zie je dat er een nieuwe regel aan het menu is toegevoegd.
De led kun je aansturen met deze nieuwe menu optie.



```
EDITOR: U6SB1
ti_hub module
Import Commands Ports Advanced
1: Hub Built-in devices...
2: Input devices...
3: Output devices...

Esc
```

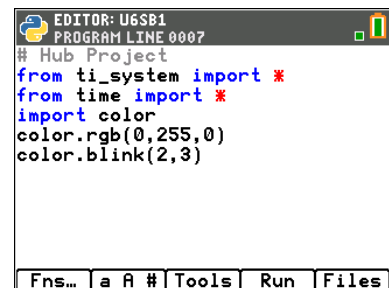


```
EDITOR: U6SB1
Modul
1: math...
2: random...
3: time...
4: ti_system...
5: ti_plotlib...
6: ti_hub...
7: ti_rover...
8: Color... <Hub Output>

Esc
```

We geven de led een kleur en laten deze drie seconden knipperen met een frequentie van twee keer per seconde.

Let erop dat de Hub is aangesloten.



```
EDITOR: U6SB1
PROGRAM LINE 0007
# Hub Project
from ti_system import *
from time import *
import color
color.rgb(0,255,0)
color.blink(2,3)

Fns... a A # Tools Run Files
```

We gaan ook geluid toevoegen.
Ga weer naar het ti_hub menu en daarin weer de Built-in devices en kies nu voor **sound**.

In het modules-menu (via ») staat nu weer een nieuwe optie: Sound.
Kies hieruit **tone(freq, time)**.

Laat nu 2 seconden een toon van 800 Hz horen.

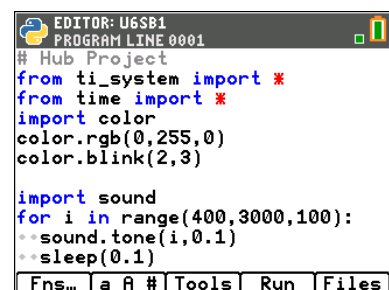


```
EDITOR: U6SB1
PROGRAM LINE 0001
# Hub Project
from ti_system import *
from time import *
import color
color.rgb(0,255,0)
color.blink(2,3)

import sound
sound.tone(800,2)

Fns... a A # Tools Run Files
```

Als laatste passen we ons programma aan door de toonhoogte te variëren.
Maak een loop die de frequentie laat lopen van 400 tot 2000 Hz met stappen van 100 Hz waarbij elke toon 0.1 seconde duurt.
Belangrijk hierbij is dat de tijd tussen twee opeenvolgende tonen ook minstens 0.1 seconde is per toon.
Je kunt hiervoor de opdracht **sleep()** gebruiken die in de time module staat.



```
EDITOR: U6SB1
PROGRAM LINE 0001
# Hub Project
from ti_system import *
from time import *
import color
color.rgb(0,255,0)
color.blink(2,3)

import sound
for i in range(400,3000,100):
    sound.tone(i,0.1)
    sleep(0.1)

Fns... a A # Tools Run Files
```

