



Unit 5 : Het gebruik van de ti_system module

Oefenblad 1: Werken met tekst

In deze les maken we kennis met de ti_system module waarmee teksten op het scherm afgedrukt kunnen worden.

Doelen :

- Werken met display functies.
- Programma stoppen met **clear**.
- modulo rekenen met de % - deling

De ti_system module maakt het ook mogelijk teksten op vaste posities op het scherm te plaatsen.

We gaan een programma maken dat om je naam vraagt en daarna die naam op het scherm laat bewegen totdat we een toets indrukken.

Als we het programma straks starten willen met een leeg scherm beginnen.

De opdracht daarvoor, **disp_clr()** kun vinden in het ti_system menu.

Vervolgens moet het programma om een naam vragen.

We gebruiken de variabele "naam" voor de naam.

Om die in te voeren terwijl het programma loopt is er de opdracht **input()**.

Die kun je vinden met **F1(Fns...)** en dan bij **I/O**.

Hiernaast zie je hoe je deze opdracht kunt gebruiken om de ingetypte tekst toe te kennen aan de variabele.

```
EDITOR: USSB3
PROGRAM LINE 0002
from ti_system import *
```

```
EDITOR: USSB3
PROGRAM LINE 0006
from ti_system import *
disp_clr()
print()
naam=input(" Wat is je naam: ")
```

We maken nu een while-loop die telkens eerst het scherm leeg maakt en dan de naam een regel lager weer afdrukt. Het regelnummer noemen we r.

We laten de loop net zolang lopen totdat we de **clear** toets indrukken.

In de ti_sytem module staat hiervoor een opdracht: **while not escape()**.

```
EDITOR: USSB3
PROGRAM LINE 0008
from ti_system import *
disp_clr()
print()
naam=input(" Wat is je naam: ")

r=1
while not escape():
  **
  -
```

Het scherm heeft 11 regels dus telkens als r = 11 moet er weer opnieuw bij regel 1 begonnen worden.

Hiervoor gebruiken we de modulo-deling (in Python is het teken hiervoor: %)

Zo geldt bijvoorbeeld $25\%11 = 3$ (je haalt een heel aantal keren 11 van de 25 af en kijkt wat je dan overhoudt).

Hiernaast staat het volledige programma.

De opdracht **disp_at()** drukt de tekst af op de aangegeven regel en lijnt die uit op de opgegeven manier (links, midden of rechts).

Ook deze opdracht staat in de ti_system module.

```
EDITOR: USSB3
PROGRAM LINE 0001
from ti_system import *
disp_clr()
print()
naam=input(" Wat is je naam: ")

r=1
while not escape():
  **disp_clr()
  **disp_at(r,"Hallo "+naam,"cente
  r")
  **r=r%11+1
```



De ti-system module heeft nog een andere nuttige opdracht namelijk **sleep()**. Deze opdracht zorgt ervoor dat het programma het opgegeven aantal seconden pauzeert.

In het programma hiernaast wordt telkens 0.2 seconden gewacht voordat het verder gaat wat ervoor zorgt dat het tempo van de lus trager wordt.

(De opdracht **sleep()** vind je in ti_system helemaal onderaan, net zolang naar beneden scrollen totdat je er bent).

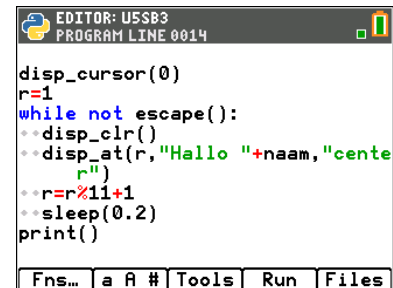
Om het nog een beetje mooier te maken kun je ook de cursor nog verbergen. Dat kan met **disp_cursor(0)**, ook in het ti_system menu.



```
EDITOR: USSB3
PROGRAM LINE 0014

r=1
while not escape():
  **disp_clr()
  **disp_at(r,"Hallo "+naam,"cente
  r")
  **r=r%11+1
  **sleep(0.2)
  print()

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



```
EDITOR: USSB3
PROGRAM LINE 0014

disp_cursor(0)
r=1
while not escape():
  **disp_clr()
  **disp_at(r,"Hallo "+naam,"cente
  r")
  **r=r%11+1
  **sleep(0.2)
  print()

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