

## Kapitel 5: Använda biblioteksmodulen ti-plotlib

I denna tredje övning i kapitel 5 kommer du att lära dig hur man använder alternativen `disp` och `sleep` hos `ti_system`-biblioteket

1: Använda `disp`-instruktionen

I de första övningarna i detta kapitel fick du lära dig att importera datalistor och hur man arbetar med regressionsekvationer.

**Kommentar:** Instruktionen `4:var=recall_regEQ()` återkallar regressionsekvationen som är beräknad från två listor. Ekvationen behöver inte motsvara en linjär modell.

- Starta ett nytt skript och döp det till `KAP5OVN3`
- Importera modulen `ti_system`.
- Lägg till raden `disp clr()`
- Om du kör skriptet kommer du alltid att få en rensad skärm och prompten placeras högst upp på skärmen.

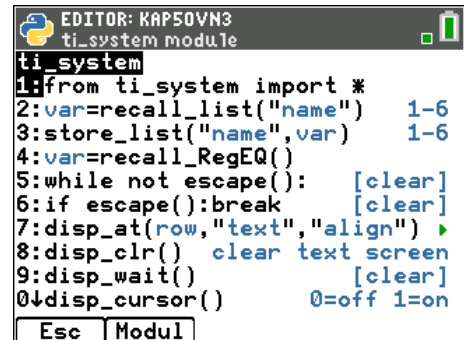
Fortsätt nu skriptet genom att infoga i en avgränsad loop och med instruktionen `7: disp_at(rad, "txt", "läge")` som du hittar i modulen `ti_system`. Den här satsen visar texten på `rad` i på skärmen genom att justera den till vänster, höger eller centrerad på raden.

Kör nu skriptet och observera hur det ser ut på skärmen!

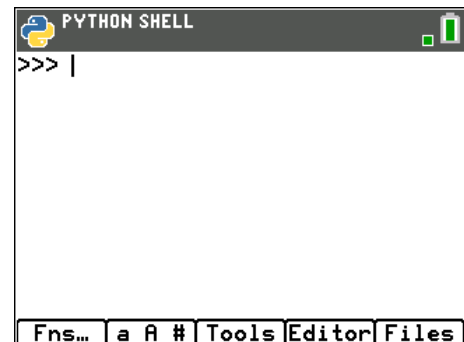
## Övning 3: Visning och timeout

## Syfte:

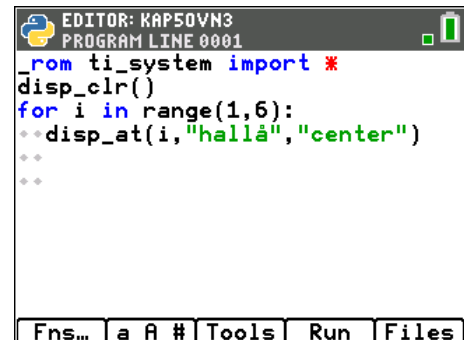
- Förstå hur instruktionerna `disp` och `sleep` fungerar
- Använda dessa instruktioner tillsammans med innehåll i andra bibliotek



```
EDITOR: KAP5OVN3
ti_system module
ti_system
1:from ti_system import *
2:var=recall_list("name") 1-6
3:store_list("name",var) 1-6
4:var=recall_RegEQ()
5:while not escape(): [clear]
6:if escape():break [clear]
7:disp_at(row,"text","align")
8:disp_clr() clear text screen
9:disp_wait() [clear]
0↓disp_cursor() 0=off 1=on
Esc Modul
```



```
PYTHON SHELL
>>> |
Fns... a A # Tools Editor Files
```



```
EDITOR: KAP5OVN3
PROGRAM LINE 0001
rom ti_system import *
disp_clr()
for i in range(1,6):
  disp_at(i,"hallå","center")
  
```



```
PYTHON SHELL
hallå
hallå
hallå
hallå
hallå>>> |
Fns... a A # Tools Editor Files
```

Redigera nu ditt skript enligt skärmen till höger. Du ska skapa tre strängvariabler **l**, **c**, **r** som därefter inkluderas i listan **p** (p står för position).

Vi ser att ordet "hallå" skrivs på rad 2, 3 och 4 *samtidigt* som det sker en förflyttning åt höger (vänster, mitten, höger). Man kan naturligtvis inte skriva något på rad 0. Det ger ett felmeddelande när man försöker köra skriptet.

**disp\_wait()**-satsen pausar skriptkörningen. Tryck på `enter` för att fortsätta exekveringen.

**Teacher tip:** Denna instruktion är intressant för att observera i steg-för-steg-körningen av ett skript inklusive en loop som föreslås här kan också ingå i en lista som en del av en mening. Till exempel "subjekt" "verb" "komplement".

Ta bort **statement disp\_wait()** och ersätt med viloläge (sekunder). Detta sista påstående introducerar en tidsfördröjning av den angivna varaktigheten.

Således kommer det föreslagna ordet att visas i dess position som motsvarar värdet på **p [i]** -listan varannan sekund.

```
EDITOR: KAP50VN3
PROGRAM LINE 0001
from ti_system import *
disp_clr()
l="left"
c="center"
r="right"
p=[l,c,r]
i=0
for i in range(0,3):
    disp_at(i+2,"hallå",str(p[i]))
```

```
PYTHON SHELL
hallå          hallå          hallå>
>> |
```

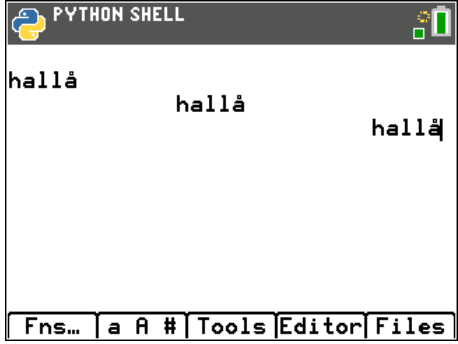
```
EDITOR: KAP50VN3
PROGRAM LINE 0010
from ti_system import *
disp_clr()
l="left"
c="center"
r="right"
p=[l,c,r]
i=0
for i in range(0,3):
    disp_at(i+2,"hallå",str(p[i]))

disp_wait()_
```

```
EDITOR: KAP50VN3
PROGRAM LINE 0010
from ti_system import *
disp_clr()
l="left"
c="center"
r="right"
p=[l,c,r]
i=0
for i in range(0,3):
    disp_at(i+2,"hallå",str(p[i]))

sleep(2)
```

Instruktionen **disp\_cursor (1)** visar markören som en vertikal linje i slutet av det ord som ska visas. **disp\_cursor (0)** kommer inte att visa det.



```

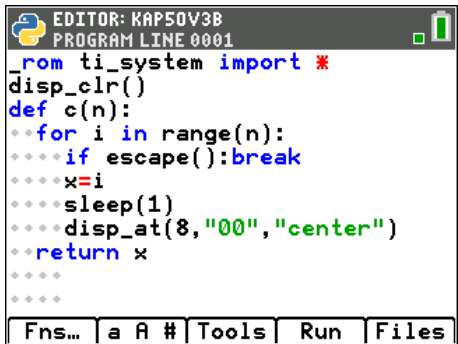
PYTHON SHELL
hallå          hallå          hallå
  
```

**2: escape()-instruktioner**

- Skapa ett nytt skript och döp det till KAP5OVN3B
- Infoga instruktionen disp\_clr() för att rensa skärmen.
- Definiera en funktion vars argument är ett heltal.

I en avgränsad loop kommer instruktionen **if escape (): break** att stoppa körningen av skriptet om du trycker på `clear`.

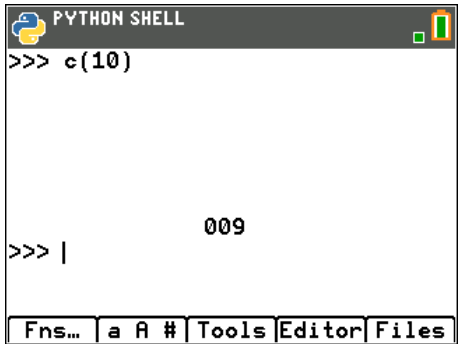
- Annars lagras i-värdet för loopsteget i en variabel **x**.
- Med en paus på 1 sekund kan du se texten "00" på skärmen.



```

EDITOR: KAP5OVN3B
PROGRAM LINE 0001
from ti_system import *
disp_clr()
def c(n):
    for i in range(n):
        if escape():break
        x=i
        sleep(1)
        disp_at(8,"00","center")
    return x
  
```

När du kör skriptet, kommer du att se skärmen flimrar lite. Syftet med det här skriptet är att visa när du anropar c(n)-funktionen för att orsaka den motsatta visningen.



```

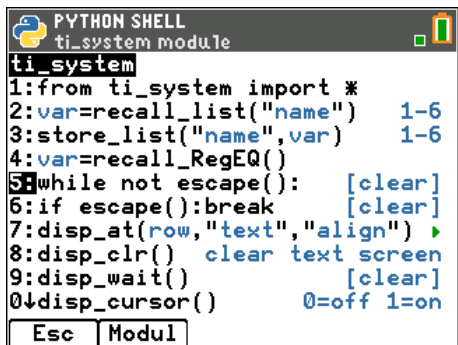
PYTHON SHELL
>>> c(10)

          009

>>> |
  
```

Driften av **satsen while not escape()** är identiskt när det gäller den tangent som ska tryckas in men här fortsätter skriptet att köras tills tangenten `clear` trycks in.

**Lärarkommentar:** Instruktionerna här kommer att vara mycket användbara när man implementerar skript tillsammans med *TI-Innovator* och *TI-Rover*.



```

PYTHON SHELL
ti_system module
ti_system
1:from ti_system import *
2:var=recall_list("name") 1-6
3:store_list("name",var) 1-6
4:var=recall_RegEQ()
5:while not escape(): [clear]
6:if escape():break [clear]
7:disp_at(row,"text","align") >
8:disp_clr() clear text screen
9:disp_wait() [clear]
0↓disp_cursor() 0=off 1=on
Esc Modul
  
```