

Forum:[Hack Forums](#)

Thread: [Countdown Timer](#)

Nakamoto. - May 27, 2015, 3:04 pm

To be honest I don't think I have ever seen a visual countdown timer, had to write this the other day for a tournament. Due to the lack of time to write it I borrowed the numbers from White Lights clock (thats a lot to type out). Anyway if anyone ever needs a timer here you go

Code @echo off

title Timer

color 0a

mode con cols=40 lines=8

setlocal enabledelayedexpansion

:starttime

set /p starttime="Enter time in Minutes: "

if %starttime% leq 0 (

echo Invalid input.

goto starttime

)

if %starttime% gtr 60 (

echo Invalid input.

goto starttime

)

:main

set mintime=%starttime%

set sectime=0

:mainloop

cls

call :display

timeout /t 1 >nul

if %sectime% geq 0 (

set /a sectime=%sectime%-1

)

if %sectime% lss 0 (

if %mintime% leq 0 (

goto endtime

) else (

set /a mintime=%mintime%-1

set sectime=59

)

)

goto mainloop

:endtime

cls

echo TIMES UP!

pause >nul

goto starttime

:display

set n1l1= 00&set n2l1=00000000&set n3l1=00000000&set n4l1=00 00&set
n5l1=00000000&set n6l1=00000000&set n7l1=00000000&set n8l1=00000000&set

```

n9l1=00000000&set n0l1=00000000&set nxl1=:::
  set n1l2= 00&set n2l2= 00&set n3l2= 00&set n4l2=00 00&set n5l2=00 &set
n6l2=00 &set n7l2= 00&set n8l2=00 00&set n9l2=00 00&set n0l2=00 00&set
nxl2=:::
  set n1l3= 00&set n2l3= 00&set n3l3= 00&set n4l3=00 00&set n5l3=00 &set
n6l3=00 &set n7l3= 00&set n8l3=00 00&set n9l3=00 00&set n0l3=00 00&set
nxl3=:::
  set n1l4= 00&set n2l4=00000000&set n3l4=00000000&set n4l4=00000000&set
n5l4=00000000&set n6l4=00000000&set n7l4= 00&set n8l4=00000000&set
n9l4=00000000&set n0l4=00 00&set nxl4=
  set n1l5= 00&set n2l5=00 &set n3l5= 00&set n4l5= 00&set n5l5= 00&set
n6l5=00 00&set n7l5= 00&set n8l5=00 00&set n9l5= 00&set n0l5=00 00&set
nxl5=:::
  set n1l6= 00&set n2l6=00 &set n3l6= 00&set n4l6= 00&set n5l6= 00&set
n6l6=00 00&set n7l6= 00&set n8l6=00 00&set n9l6= 00&set n0l6=00 00&set
nxl6=:::
  set n1l7= 00&set n2l7=00000000&set n3l7=00000000&set n4l7= 00&set
n5l7=00000000&set n6l7=00000000&set n7l7= 00&set n8l7=00000000&set
n9l7= 00&set n0l7=00000000&set nxl7=:::
  if %mintime% geq 10 (
    set min1=%mintime:~0,1%
  ) else (
    set min1=0
  )
  set min2=%mintime:~-1,1%
  if %sectime% lss 10 (
    set sec1=0
  ) else (
    set sec1=%sectime:~0,1%
  )
  set sec2=%sectime:~-1,1%
  for /l %%a in (1,1,7) do (
    call echo %%n!min1!%%a%% %%n!min2!%%a%% %%n!nxl%%a%% %%n!sec1!%%a%%
%%n!sec2!%%a%%
  )
goto :eof

```

Just in case its mutilated by the forum parsing [here](#) is a pastebin.

EDIT: Thanks to boosted fixing what I was to lazy to, it is now more accurate and does a full minute instead of losing 1 second off of every minute.

b00st3d - May 27, 2015, 3:17 pm

Looks good. Simple and effective just the way I like it.

Nakamoto. - May 29, 2015, 10:35 pm

(May 27, 2015 03:17 PM)b00st3d Wrote: [\[CITE\]](#) Looks good. Simple and effective just the way I like it.

Not sure how accurate it is, it may be losing a second from every minute so if you set if for an hour you may only be getting 59 minutes.

Advertisement - May 30, 2015, 4:32 am

(May 29, 2015 10:35 PM)RenRioku™ Wrote: [\[CITE\]](#) Not sure how accurate it is, it may be losing a second from every minute so if you set if for an hour you may only be getting 59 minutes.

Didn't really read the code but if you could you could use a real timer, not bat no offence and use it

for hour and if your guess is right use some sort of

```
Code set /a SecondsVariableThing=%SecondsVariableThing% - 1
```

Nakamoto. - May 30, 2015, 5:27 am

(May 30, 2015 04:32 AM)RekkiNoobsAndGrabbinBoobs Wrote: [\[CITE\]](#) Didn't really read the code but if you could you could use a real timer, not bat no offence and use it for hour and if your guess is right use some sort of

```
Code set /a SecondsVariableThing=%SecondsVariableThing% - 1
```

I could but the purpose I wrote it for it isn't required to be super accurate, so I see no sense for me to sit for an hour to test it.

b00st3d - May 30, 2015, 7:14 am

I didn't see it at first but this should be set at 60 rather than 59 because it immediately changes from 0 to 59 then waits a sec to change to 58. You could also check if %sectime% LSS 0 and leave the below line the same.

```
Code set /a mintime=%mintime%-1
      set sectime=59
```

Crez - May 30, 2015, 7:31 am

Looks really awesome :) Sadly I had to exchange the unicode chars because my Windows already supports Unicode in Batch scripts, that's why I just saw a bunch of weird chars at the first time starting it. Great job man! I would +rep you if I could :-/

(English is not my native english, that's why most posts contain some mistakes. Thank you for understanding.)

Nakamoto. - June 1, 2015, 12:56 pm

(May 30, 2015 07:14 AM)b00st3d Wrote: [\[CITE\]](#) I didn't see it at first but this should be set at 60 rather than 59 because it immediately changes from 0 to 59 then waits a sec to change to 58. You could also check if %sectime% LSS 0 and leave the below line the same.

```
Code set /a mintime=%mintime%-1
      set sectime=59
```

Changing the LEQ to LSS doesn't work, it break the script and I don't like seeing a 60 there for a second, doesn't seem clocky.

b00st3d - June 1, 2015, 1:21 pm

(Jun 1, 2015 12:56 PM)RenRioku™ Wrote: [\[CITE\]](#) Changing the LEQ to LSS doesn't work, it break the script and I don't like seeing a 60 there for a second, doesn't seem clocky.

It does work. You just need to change if %sectime% gtr 0 to GEQ. see below for an updated mainloop

```
Code :mainloop
      cls
      call :display
      timeout /t 1 >nul
      if %sectime% geq 0 (
        set /a sectime=%sectime%-1
      )
      if %sectime% lss 0 (
        if %mintime% leq 0 (
```

```

        goto endtime
    ) else (
        set /a mintime=%mintime%-1
        set sectime=59
    )
    )
    goto mainloop

```

Nakamoto. - June 1, 2015, 1:31 pm

(Jun 1, 2015 01:21 PM)b00st3d Wrote: [\[CITE\]](#) It does work. You just need to change if %sectime% gtr 0 to GEQ. see below for an updated mainloop

Code :mainloop

```

    cls
    call :display
    timeout /t 1 >nul
    if %sectime% geq 0 (
        set /a sectime=%sectime%-1
    )
    if %sectime% lss 0 (
        if %mintime% leq 0 (
            goto endtime
        ) else (
            set /a mintime=%mintime%-1
            set sectime=59
        )
    )
    )
    goto mainloop

```

Didn't cross my mind to change both but I guess it does work lol. On a side note although I tried to make it idiot proof I didn't think to cover every base of someone that might try to break it like myself.

if you enter a number followed by a letter it sets the mintime to the number then starts the sec time at 0 goes to 59 and goes a whole minute before mintime changes at all.

Example 4a makes it read 4:00 a second later its 4:59, no real reason to try and fix it, not a pressing matter.

Austerity - June 1, 2015, 8:17 pm

Very nice work, this works perfectly.

xxv2012 - August 6, 2015, 12:10 pm

very good timer thanks for sharing ^_^