

תרגיל - Task manager

הדגמה זו תציג לנו את מנהל המשימות (task manager). תוכנה זו היא התוכנה הבסיסית ביותר ב-Windows על מנת לנהל את התהליכים הרצים ולקבל מידע על המערכת.

פתיחת מנהל המשימות

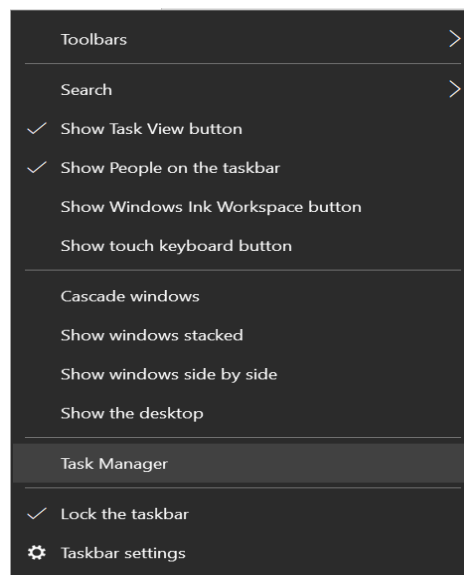
ישנן 3 דרכים לפתוח את מנהל המשימות:

1. על ידי לחיצת שלשיית המקשים ctrl+alt+delete ביחד, ולאחר מכן בחירה ב - Start Task Manager



2. על ידי לחיצה על שלשיית המקשים ctrl+shift+esc ביחד

3. על ידי לחיצה ימנית על שורת ההתחלה ובחירת "התחל מנהל המשימות" (start task manager)



משימה 1:

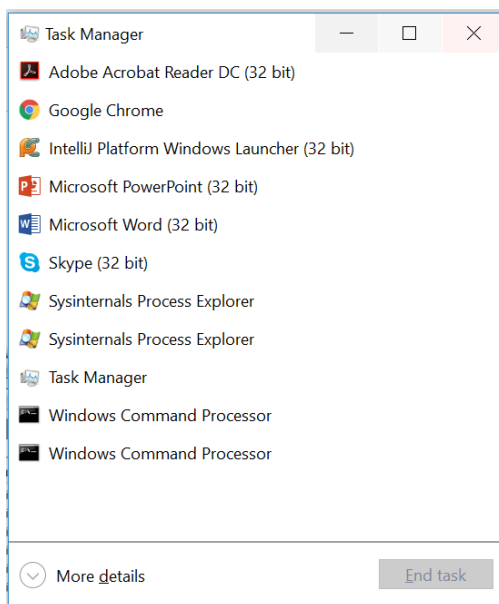
פיתחו את מנהל המשימות בכל אחת משלושת הדרכים.

למה ישנן מספר דרכים שונות לבצע את אותה הפעולה? הסיבה היא שפתיחת מנהל המשימות היא פעולה שקוראת הרבה פעמים, לכן כדאי שהיא תהיה הכי יעילה שאפשר. לכן קיימות שיטות 2 ו-3 (עבור מי שמשתמש בעכבר, ומי שמשתמש במקלדת). אז למה קיימת גם שיטה 1? הסיבה לכך היא כי שיטה 1 היא special escape sequence כלומר רצף פקודות מיוחד ש Windows מגדיר שאמור תמיד לעבוד (גם אם רוב התוכנות במחשב לא עובדות, לחיצה על `ctrl+alt+delete` אמור לעבוד כמו שצריך). המטרה היא ששיטה זו תהיה זמינה על מנת שהמשתמש יוכל להוציא את עצמו מכל תקלה שהיא, ומנהל המשימות הוא כלי חשוב בעבודה זו. לאחר שמתחילים את מנהל המשימות נסתכל על המסך הראשוני שהוא מראה, ונראה מה האפשרויות שיש בו. אנו נתמקד במסכי ה Applications, Processes, Performance -ונדון בהם.

הבנת מנהל המשימות

Fewer details

נתבונן על המצב הפשוט (fewer details).



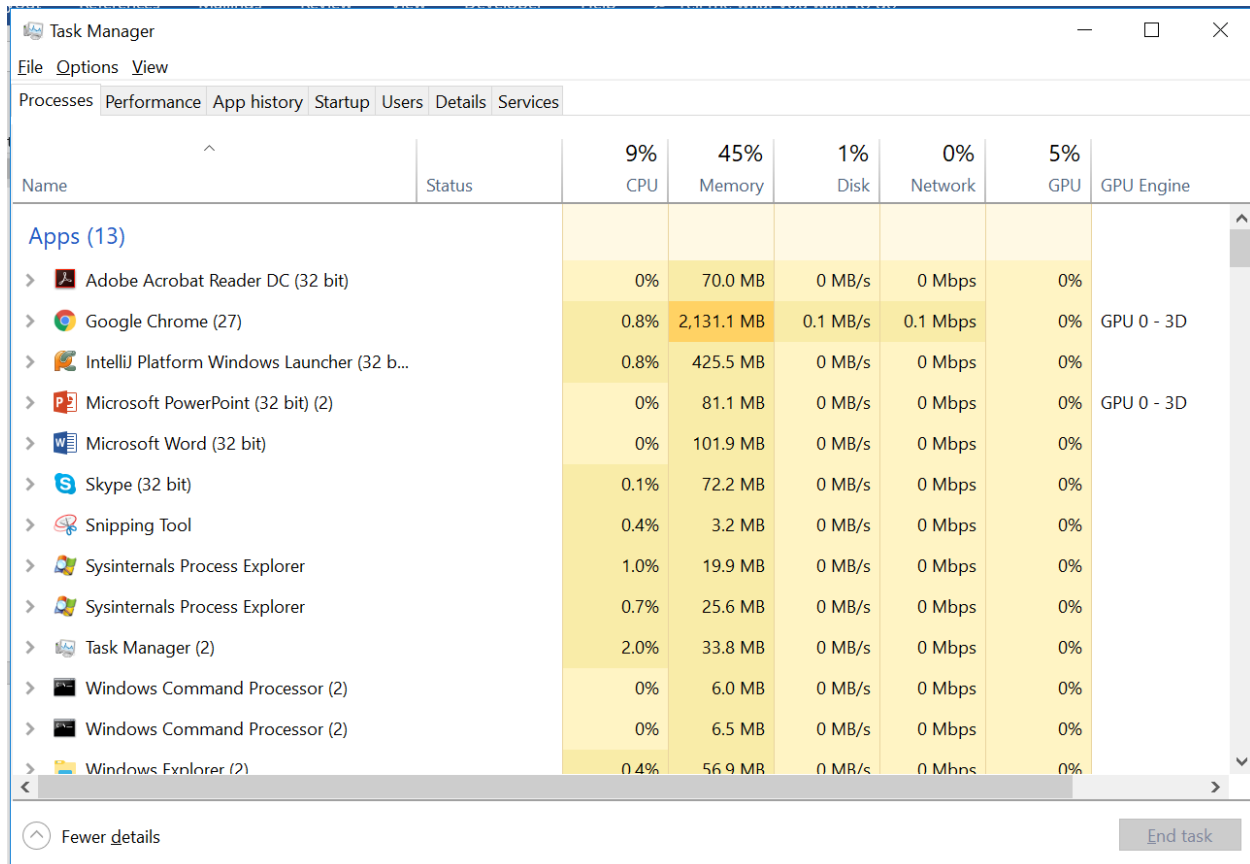
מסך זה מציג את התוכנות שרצות על המחשב. המסך לא מציג את כל התוכנות, אלא רק חלק. **שאלה:** האם אתם מסוגלים להבין אילו תוכנות מוצגות פה ואילו אינן? **תשובה:** המסך מציג את התוכנות שיש להם חלון שניתן לראות (ההגדרה האמתית יותר מורכבת, אבל זה קרוב מספיק לעכשיו).

משימה 2 : פתחו מחשבון (`calc.exe`) ובדקו שהתוכנה מופיעה ברשימה. נסו את כל הפעולות שאפשר לעשות איתו בעזרת המקש הימני כמו "Bring To Front", "Switch To" ובסוף "end task".

[More details](#)

נעבור למצב הסימניות בעזרת לחיצה על [more details](#).

סימניות processes ו details



Name	Status	9% CPU	45% Memory	1% Disk	0% Network	5% GPU	GPU Engine
Apps (13)							
Adobe Acrobat Reader DC (32 bit)		0%	70.0 MB	0 MB/s	0 Mbps	0%	
Google Chrome (27)		0.8%	2,131.1 MB	0.1 MB/s	0.1 Mbps	0%	GPU 0 - 3D
IntelliJ Platform Windows Launcher (32 b...		0.8%	425.5 MB	0 MB/s	0 Mbps	0%	
Microsoft PowerPoint (32 bit) (2)		0%	81.1 MB	0 MB/s	0 Mbps	0%	GPU 0 - 3D
Microsoft Word (32 bit)		0%	101.9 MB	0 MB/s	0 Mbps	0%	
Skype (32 bit)		0.1%	72.2 MB	0 MB/s	0 Mbps	0%	
Snipping Tool		0.4%	3.2 MB	0 MB/s	0 Mbps	0%	
Sysinternals Process Explorer		1.0%	19.9 MB	0 MB/s	0 Mbps	0%	
Sysinternals Process Explorer		0.7%	25.6 MB	0 MB/s	0 Mbps	0%	
Task Manager (2)		2.0%	33.8 MB	0 MB/s	0 Mbps	0%	
Windows Command Processor (2)		0%	6.0 MB	0 MB/s	0 Mbps	0%	
Windows Command Processor (2)		0%	6.5 MB	0 MB/s	0 Mbps	0%	
Windows Explorer (2)		0.4%	56.9 MB	0 MB/s	0 Mbps	0%	

משימה 3 פתחו את מסך התהליכים. מצאו במסך את calc. נסו למצוא את המיקום של קובץ ההרצה במחשב ומי החברה שיצרה אותו. מה מזהה התהליך של calc?

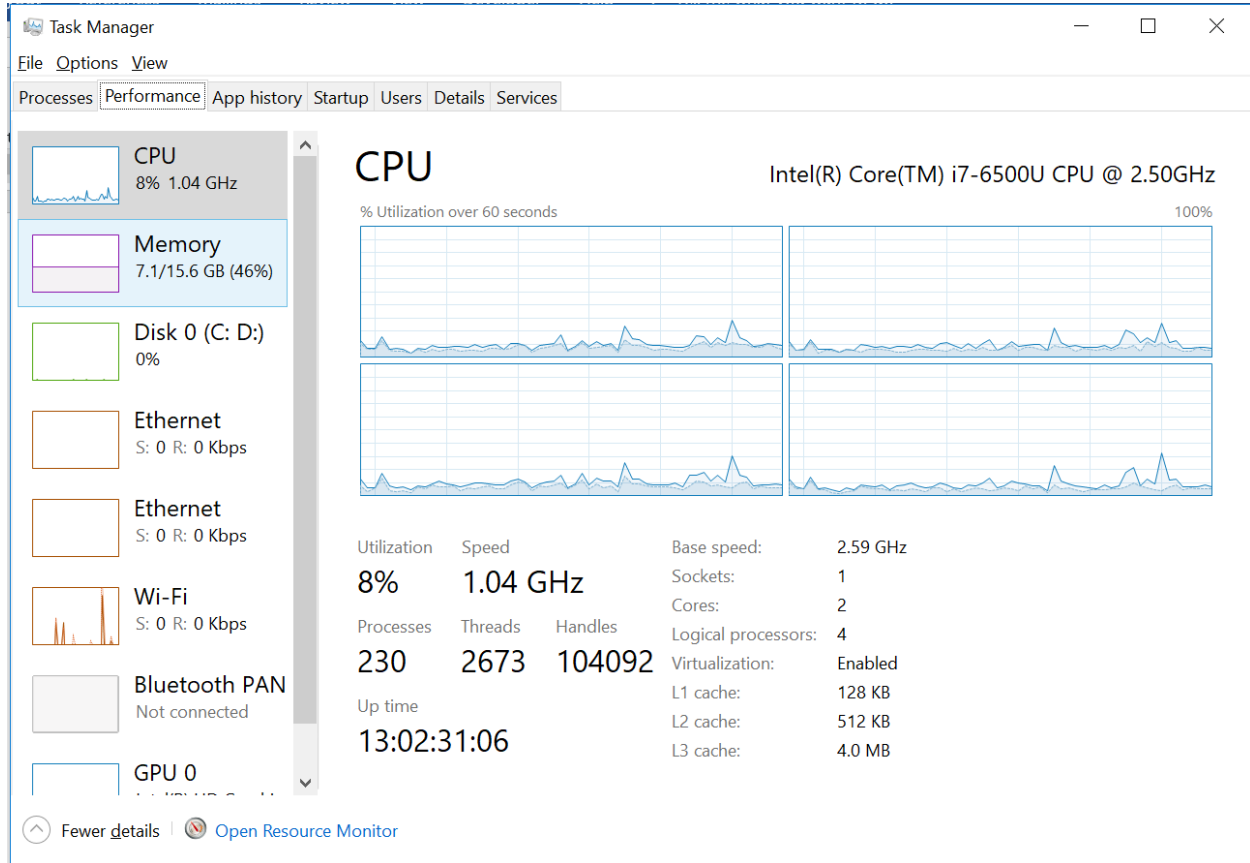
• PID – מזהה ייחודי שיש למערכת ההפעלה לכל תהליך.

מצאו את התהליכים החשובים הבאים (התיאור של כל אחד מהם הוא פשטני מאוד, הוא מובא פה רק כדי שיהיה קצת היגיון כשנראה אותם בהמשך) (***יש לאפשר את העמודה process name**):

- System - התהליך שמייצג את ה "מערכת הפעלה עצמה" בעצם דברים שקשורים למערכת הפעלה אבל לא לאף תהליך (קורים לחלוטין ב kernel)
- Windows Explorer – הנווט של windows, מה שאנחנו בעצם רואים בו את כל התיקיות והמסך עבודה שלנו.
- Calc – המחשבון
- svchost – תוכנה כללית של windows שמכילה services רבים.
- winlogon – מסך ה login.
- lsass – אחראי על ניהול המשתמשים.
- csrss – התהליך הראשון שנוצר עבור כל משתמש.
- smss – התהליך שאחראי ליצור את ה csrss של כל משתמש.

- System idle process - התהליך שמייצג את הזמן שהמערכת לא עושה כלום.

סימנית Performance



מסך זה מציג סיכום של ביצועי המחשב באופן כללי. מה מסמל כל פריט במסך?
משימה 4 פתחו את מסך הביצועים של מנהל התהליכים והסבירו את השינויים שרואים בו בהתאם לפעולות הבאות ב python (מומלץ לשים את מנהל המשימות ברקע ככה שאפשר לראות אותו, ואת ה-python לידו במסך).

א. הקצו רשימה גדולה מאוד $for i in range(10**9)$ [יכול להיות שצריך לשחק טיפה עם הגודל המדויק של המערך בשביל אפקט טוב] מה קרה לגודל הזיכרון שהתוכנה צורכת?
 ב. מחקו את הרשימה הגדולה שיצרנו באמצעות $del(a)$ יכול להיות שכדי שהזיכרון ירד צריך לקרוא ל- gc (garbage collector), עושים זאת ע"י

```
Import gc
gc.collect()
```

מה קרה לצריכת הזיכרון עכשיו?
 ג. כתבו את הסקריפט הקצר הבא:

```
while True:
    pass
```

מה השתנה הפעם? (רמז: הסתכלו בעמודת ה cpu).
 ד. הפסיקו את הלולאה האינסופית (מספיק בחלון של ה python ללחוץ ctrl+c), מה קרה עכשיו?

ה. החליפו בין תצוגה כללית לתצוגה של CPU לוגי (עימדו על גרף ה CPU ולחצו על הכפתור הימני).
ו. הוסיפו את גרף ה kernel.

קרדיט:
עומר ברק
תומר גלון