# Linux

###### מקומות שנחמד לחפש בהם באינטרנט

Askubuntu

Stackexchange

###### סיסמאות:

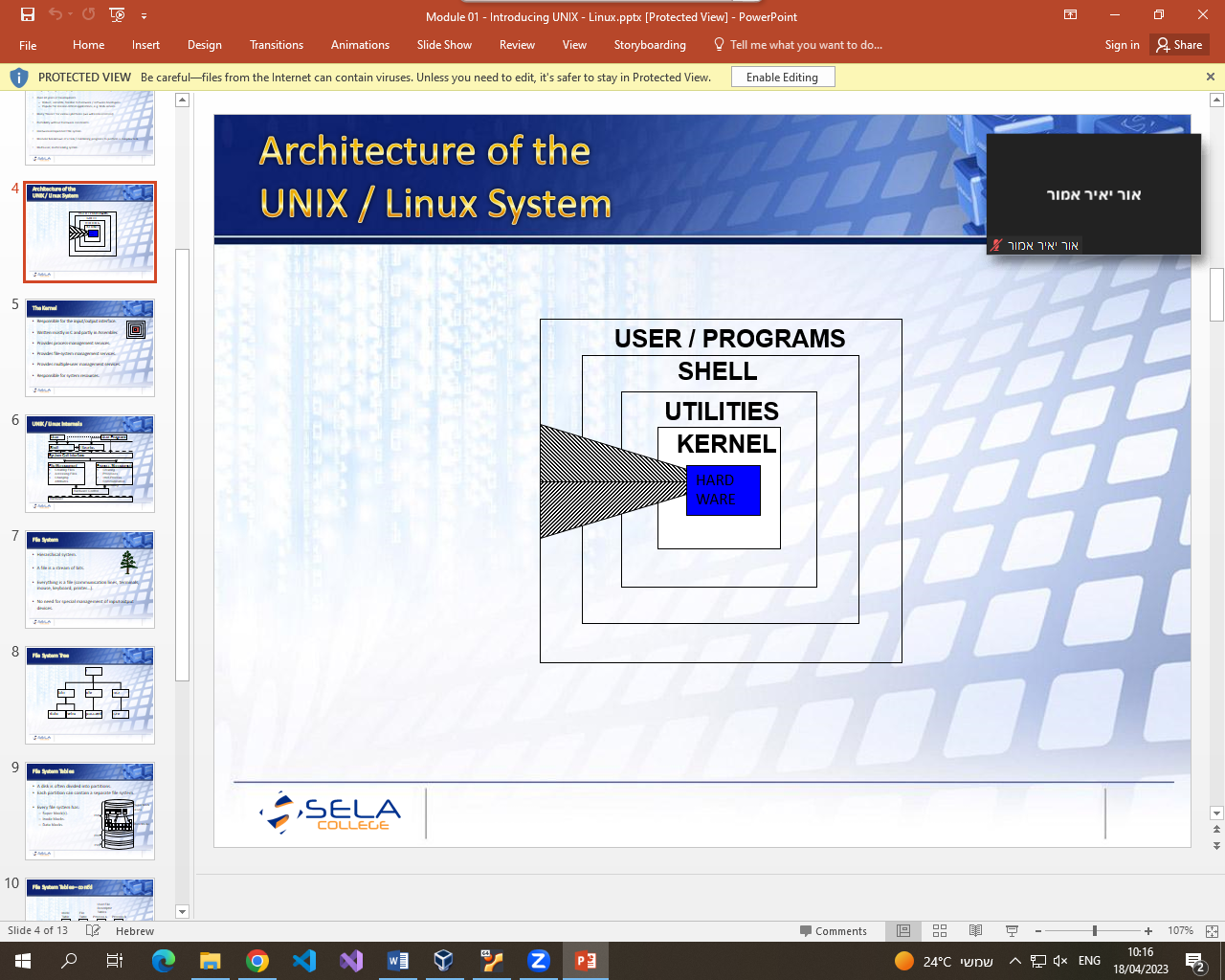
Avital: 1234

Root: ת.ז

## Introduction

It's an open source operation system   
it means that everyone can access the code and make changes   
its lighter and more reliable then windows

### System structure



#### Hard disk

###### Installing a new hard disk

Df –h = מראה את כל הדיסקים שמחוברים למחשב והניצול בנפח

Du –h=רשימה של כל הקבצים בתיקיה מסוימת והגודל שלהם (-h = humer)

* Sort –n = מסדר מהקטן לגדול לפי מספר (-r = reverse)
* Du –sh = summery מראה רק את המשקל של התיקיות (בלי רקורסיב)

ncdu= כלי שמראה לי הכל בצורה יותר גרפית מהגדול לקטן

Dd= ניגש ישירות להארד דיסק עצמו (block device)

Ls dev/sd\*= מראה את כל הדיסקים והמחיצות במערכת

/sbin/mkfs.ext4 /dev/sdc= יוצר מערכת ניהול תיקיות לדיסק מסויים

* -L /name= נותן לייבל לנפח שניתן להשתמש בו במקום השם של המערכת לדיסק

Mount = יוצר קישור בין הנפח הפיזי של הדיסק לבין נתיב מסוים בעץ התיקיות

#### Umount = מבטל את הקישור אבל המידע עדיין קיים

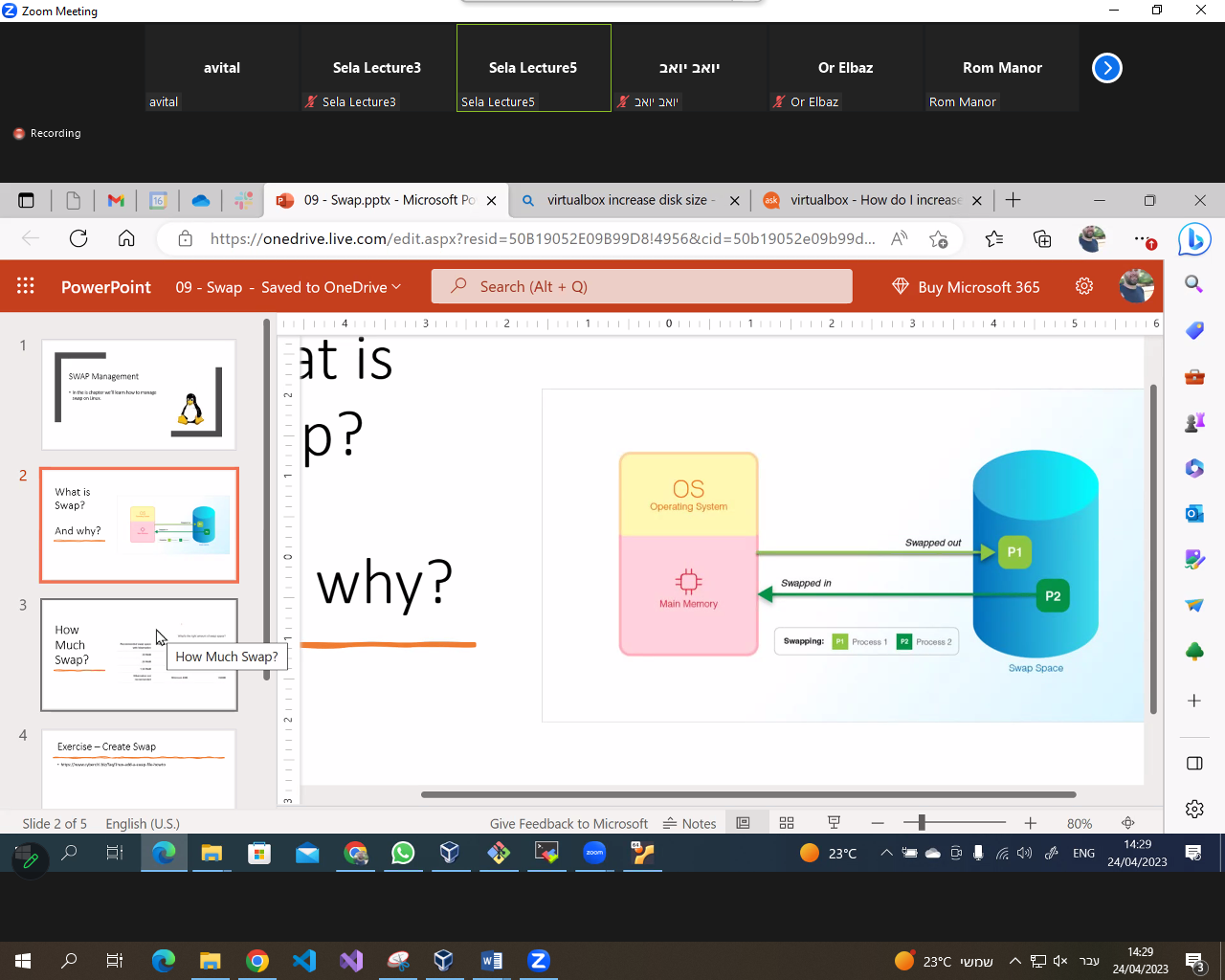
**יש איזה קובץ שצריך לערוך ולהוסיף אליו את הפעולה של המאונט**

Mount -a= עושה מאונט לכל הדיסקים- משמש כבדיקה לראות שהכל יעלה בהפעלת המחשב

#### Mnt = דברים זמניים

#### SWAP

משאילים חלק מה storage (הארד דיסק) לטובת הזיכרון (ראם)



Free –h

מראה כמה זיכרון פנוי יש לי וכמה בשימוש  
וגם מראה כמה swap יש לי וכמה בשימוש  
בדרך כלל ה swap יהיה שווה ל memory או יותר

#### i-node

### A logical partition of the hard disk that keeps information about the data.

Every object that is being created gets an   
i-node number.

##### Links

Leads me to the relevant file

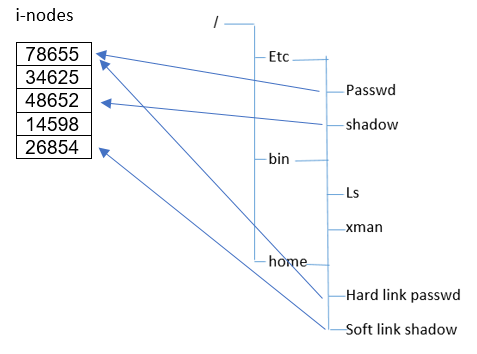
###### Hard link

Creates a link based on the i-node number. He will have the same i-node number as the file

###### Soft link

Creates a connection to the path of the file. Will have a different i-node number from the file. Also called a symbolic link, more common.

Creation: ln –s "/path" "link name"



## Ubuntu

* אנחנו עובדים עם מערכת ניהול קבצים בשם ext.
* אנחנו משתמשים ב shell שנקרא bash. הוא גרסה מחודשת של sh.

### File system

/etc/shadow= קובץ סיסמאות מוצפן

/etc/passwd= קובץ משתמשים

/etc/group= קובץ קבוצות

#### Processes

Ps – command to show the processes running right now

* -ef – shows all the processes exists on the computer

Pid – process id

Ppid – parent process id

Init – the very first process, the father of all the processes

Kill- will make a process to stop working

Kill -9 – no process can resist this command

#### Jobs

### Terminal

**Linux is Case Sensitive!**

דברים שכדאי לדעת

* נתיב יחסי: יחסית לאן שאני עכשיו –לא יתחיל ב " /" - אתחיל ממה שאני רואה בתוך התיקיה
* נתיב אבסולוטי: לא משנה איפה אני נמצאת עכשיו (לא צריך cd) תמיד יתחיל ב " /"  
  מקביל ל "C:" ב windows
* ניתן לקצר תהליכים באמצעות " \* "לצורך תפיסת כל הקבצים תחת תיקיה מסוימת
* אין דבר כזה גיבוי בלי שחזור שעובד!
* כל פקודה שאריץ שמורה בתור קובץ ב /usr/bin
* יוזר רוט לא צריך סיסמאות כדי להיכנס ליוזרים רגילים
* קבצים שמתחילים ב "." הם קבצים נסתרים (לא נראה אותם ב ls)

#### Commands

|  |  |
| --- | --- |
| **Command** | **Action** |
| man | מדריך על השימוש בפקודה |
| whoami | מציג את שם המשתמש |
| Id | פרטים על המשתמש |
| pwd | מראה את הנתיב המלא שאני נמצאת בו |
| Ls | מציג את כל התיקיות והקבצים בנתיב |
| - | flag |
| touch | יוצר קובץ גנרי ריק |
| mkdir | יצירת תיקיה |
| rmdir | מחיקת תיקיה |
| rm | מחיקת קובץ |
| tree | מציג את עץ התיקיות בנתיב |
| cp | העתק הדבק |
| mv | גזור הדבק |
| cd "path" | מעבר בין נתיבים |
| .. | תיקיה אחת למעלה |
| . | התיקיה שאני נמצאת בה |
| cd / cd ~ | חזרה לתיקיית הבית של היוזר |
| clear / ctrl + l | מנקה את הטרמינל |
| ./"file\_name" | הרצה של קובץ |
| nano | עורך בסיסי |
| חץ למעלה | מחזיר לי את כל הפקודות שעשיתי |
| cat | מדפיס את התוכן של הקובץ |
| more | מראה באחוזים כמה נשאר |
| less | פותח במסך נפרד גדול יותר |
| head | מדפיס את השורות הראשונות |
| tail | מדפיס את השורות האחרונות |
| su - "user\_name" | מעבר למשתמש אחר |
| su - | מעבר למשתמש root |
| sudo | ביצוע פעולה באמצעות משתמש root |
| passwd | מתן סיסמה ליוזר (צריך להיות בתוך היוזר) |
| usermod | הוספת יוזר לקבוצה |
| exit / ctrl + d | חזרה לאן שהייתי |
| apt install " Package\_name" | פקודה להתקנה |
| tar | פקודת דחיסה/כיווץ |
| grep "word" "path" | מראה רק את השורות שמכילות מילה מסוימת |
| witch | מראה את הנתיב שבו נמצאת הפקודה |
| $ | קורא למשתנה |
| " " | הופך את התוכן לסטרינג אחד (כשיש $) |
| ' ' | מציג את הסטרינג בדיוק כמו שהוא כתוב(כשיש $) |
| path | משתנה שמכיל את הנתיבים לכל הפקודות |
| useradd | פקודה להוספת משתמש |
| wc | ספירת מילים |
| wc -l | ספירת שורות |
| Groupadd | יצירת קבוצה |
| Groups | מראה את כל הקבוצות שהיוזר נמצא בהן |
| > | זורק את ה output על המסך לתוך הקובץ |
| >> | מוסיף את ה output לקובץ שכבר קיים |
| Nano ."file\_name" | גישה ועריכה של קובץ נסתר |
| sort | מסדר את הפלט מהגדול לקטן |
| awk |  |
| sed |  |
| chown | שינוי בעלות על קובץ או תיקיה |
| trap | ביצוע פעולה נוספת לפני ביצוע הפעולה הניתנת |

#### Link to [Additional commands](https://www.geeksforgeeks.org/basic-shell-commands-in-linux/)

#### 

#### Commands Examples

* mv "file\_name" "dir\_name"/"new\_file\_name" = יעביר את הקובץ לתיקיה אחרת עם שם חדש
* mv "old\_file\_name" "new\_file\_name" = שינוי שם קובץ
* mv "path/source\_folder" "path/destination\_folder" = העתקת תיקיה לנתיב חדש
* cp "file\_name" "dir\_name" = יעתיק את הקובץ לתיקיה אחרת
* cp –r "old\_dir\_name" "new\_dir\_name" = יצירת עותק של תיקיה עם שם חדש
* tar –zcvf "file\_name.tar.gz" "file\_name" = zipדחיסת תיקיה לקובץ
* useradd "user\_name" –m –s /bin/bash = bashיצירת יוזר ותיקיית בית עם
* usermod –aG "group\_name" "user\_name"= הוספת יוזר לקבוצה
* awk –F: '{ print $1}' /etc/passwd=
* sed 's/"old\_value"/"new\_value"/g' > "file\_name"= משנה את כל המילים בטקסט למילה אחרת ושומר לקובץ
  + ישמור את השינויים לקובץ המקור –i
* chown –R user2:user2 user2= ישנה את הבעלות על התיקיה   
   owner:group file name
  + תופס גם את התוכן של התיקיה –R
* Chmod 750 user2=ישנה את ההרשאות של התיקיה
* grep ^"later" "path"= כל המילים שמתחילות באות מסוימת
* grep "later"$ "path"= כל המילים שמסתיימות באות מסוימת

#### Flags

**לכל פקודה יש אוסף פלגים משלה!!**

דוגמות:

-l = long: הפלט יהיה מפורט יותר

-t = time: מראה לפי הזמן

-r = reversed: הפלט יהיה בסדר הפוך

-R = recursive: יראה גם את כל התיקיות והקבצים שמתחת

-"number": מדפיס מספר מסוים של שורות

-c= compressדחיסה :

-z= zip: סוג הארכיון

-f=file: הקובץ שאליו זה יכנס (מיד אחריה יגיע שם הקובץ)

-v= verbose: ידפיס את הפעולות שהוא עושה – אחיד לכל הפקודות

-x= extract: חילוץ

-s= shell בחירת

trap "rm –f \*.tmp" EXIT = ביציאה מהסקריפט הוא ינקה את כל הקבצים שהוא יצר(-f = force)

* ניתן לאחד פלגים. דוגמה: -ltr

#### Package management

### apt

#### Stands for Advanced Package Tool

#### You can use the apt command to install, delete or remove apps.

##### Install examples

vim

spell

changeme

tree

#### nano text editor

##### Opening and Creating Files:

##### nano "file\_name"

The above command will open a new file with filename as shown in the output. In case the file already exists it will open the same and in case the file is not there in the current directory it will create a new one.

##### Saving and Exiting:

##### To save your changes press Ctrl+o.

To exit nano press Ctrl+x

It will ask you for the filename. In case you want to save the changes to a new file or want to create a new file then change the name. else, keep the name same.

##### Copping, cutting, and pasting:

##### Ctrl+k is used to cut and Ctrl+u is used to paste the text.

##### To search a word in a file:

##### To search Press Ctrl+w

#### To move to the next match, press Alt+w

#### **מה זה vim?**

#### bashrc

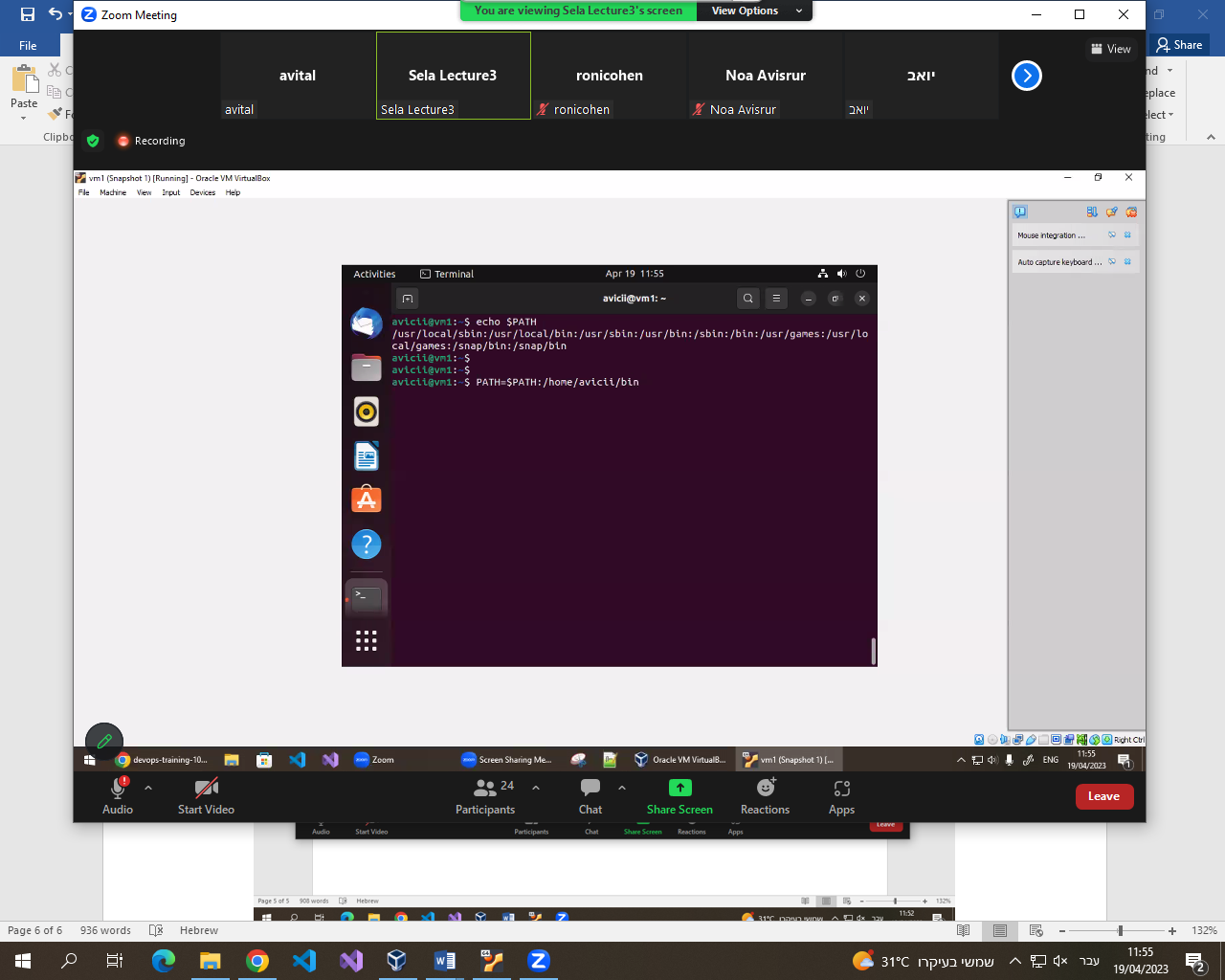
Bashrc= קובץ שמכיל הגדרות מערכת.  
יש אחד על תיקיית הבית של היוזר שמשפיע רק על היוזר  
מגיעים אליו באמצעות: nano .bashrc  
ויש אחד שיושב ב/etc/bash/bash.rc והוא משפיע על כל היוזרים

### Alias

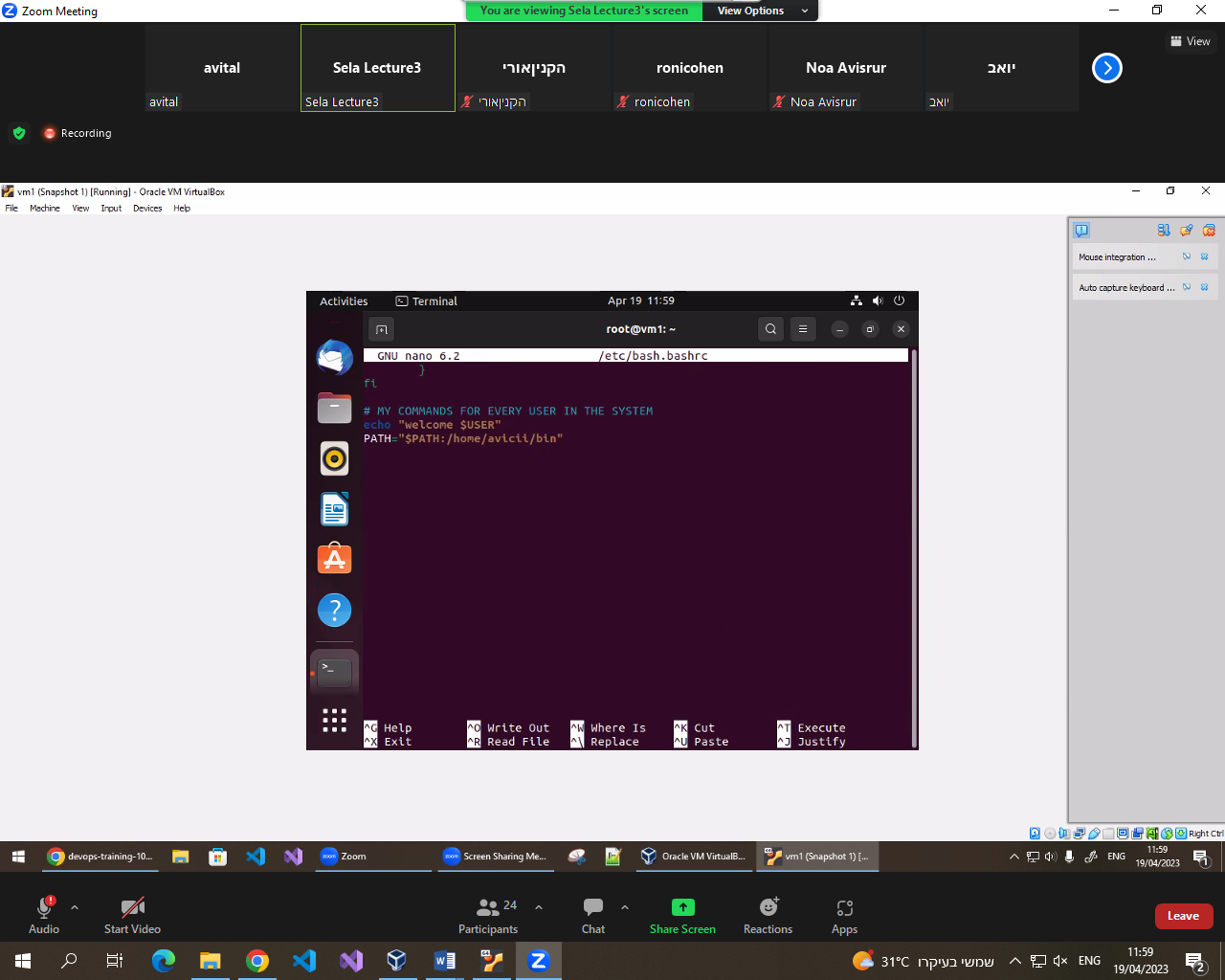
רשימה של ביטויים שעוזרים לקצר פעולות ארוכות, ניתן לראות אותם תחת alias.  
דוגמה:  
מוסיפים alias חדש בקובץ bashrc.

### Path

משתנה שמחזיק את כל הנתיבים שבהם יושבות פקודות שאני יכולה להריץ.   
אם הנתיב של הפקודה לא יושב במשתנה הזה אני לא אוכל להריץ אותה.   
הוא יושב ב /etc/environment

  
ניתן להוסיף נתיב נוסף ב /etc/environment  
או באמצעות (למרות שיש שם איזה באג שזה יוסיף את אותה השורה פעם נוספת בכל התחברות)

Nano /etc/bash/bashrc



### Profile

#### Users and groups

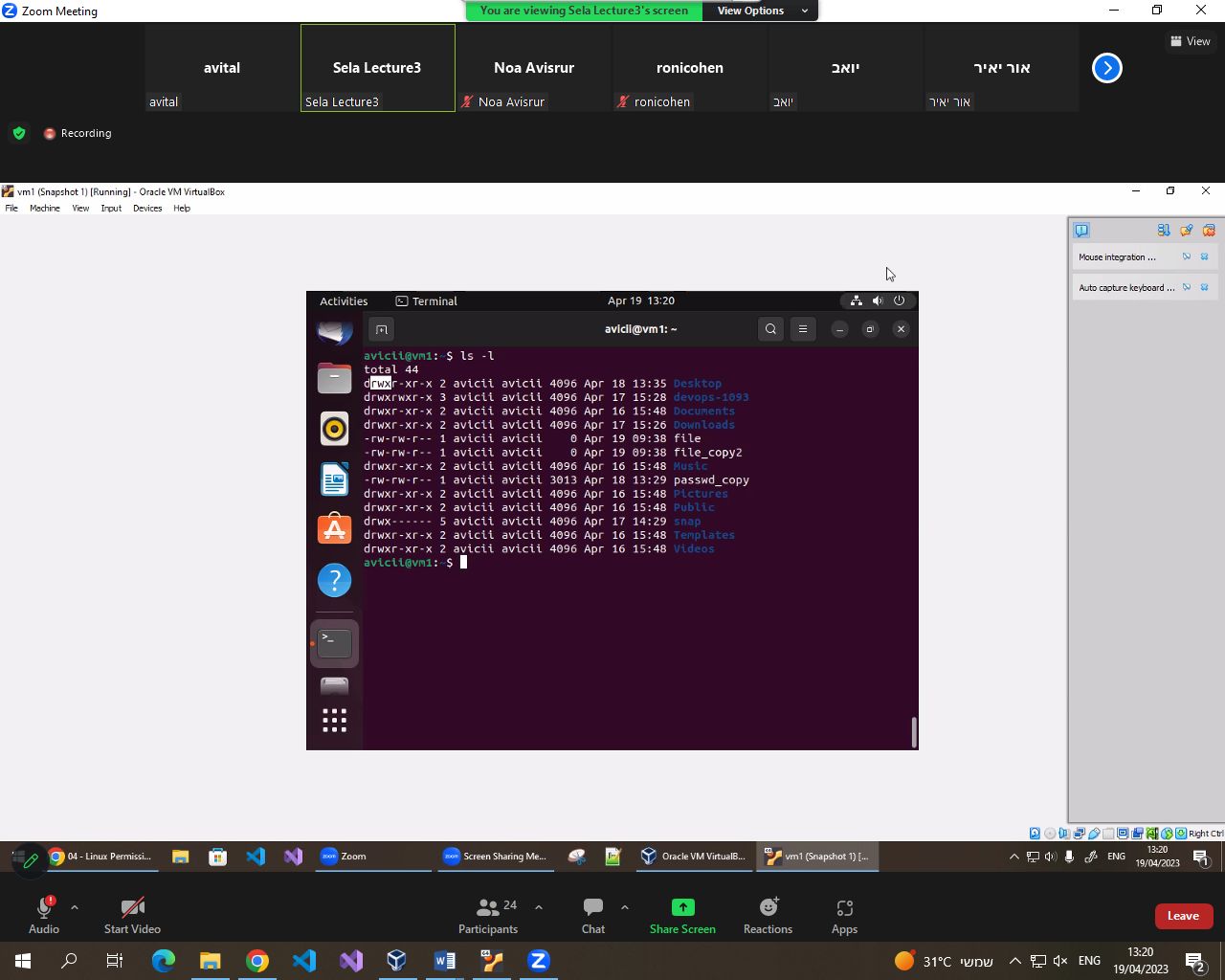
Useradd -m -s /bin/bash/"username"= יוצר משתמש בפורמט בש

* -m -s ?

Passwd "username"= נותן סיסמה ליוזר

#### Permissions

כדי לראות את ההרשאות של האובייקטים נעשה: ls -l

D =directory

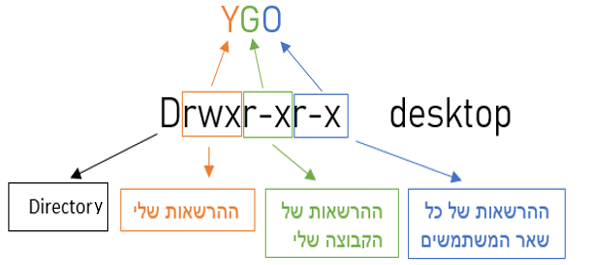
- = file

R = read

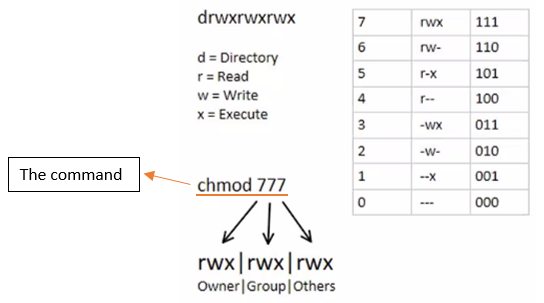
W = write

X = execute

עבור כל אובייקט יש 3 סוגי הרשאות: משמאל לימין: אני, הקבוצה שלי, כל השאר

דוגמה:

* לכל תיקיה יש יוזר שהוא האונר שלה (בדרך כלל מי שיצר אותה) וגרופ שהוא האונר שלה.

מתן הרשאות:

### Scripting – bash

#! /usr/bin/bash

בכתיבת סקריפטים נשתדל שהפקודות יהיו כמה שפחות אינטראקטיביות כי זה תוקע את האוטומציה.

#### Loops

|  |  |
| --- | --- |
| While loop | For loop נשתמש כשאנחנו יודעים את הערך הראשון והאחרון או כשיש לנו מספר סופי של ערכים |

#### Variables

I can define a variable and then call it in a script

By convention, Unix shell variables will have their names in UPPERCASE

##### Defining and Accessing Variables

### To define a variable: "Variable\_name"="variable\_value"

### To access the value stored in a variable: $"variable\_name"

* $1 = קורא לארגומנט הראשון מהאינפוט
* $2= קורא לארגומנט השני
* $#= סופר את מספר הארגומנטים
* $@= קורא לכל הערך שהוזן

#### Regular expression

|  |  |
| --- | --- |
| lt | < |
| le | <= |
| gt | > |
| ge | >= |
| ^ | מתחיל ב |
| $ | נגמר ב |
| \w | [a-zA-Z0-9] (כולל רווחים) |

דברים שימושיים

###### דרך לראות את השינויים בתוכן של קובץ בזמן אמת תוך כדי הרצת הסקריפט:

tail –f "file name"

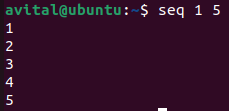
* -f=follow

###### דרך ליצור תיקיות לצורך בדיקת סקריפטים:

rm dir/\* && touch dir/file1 dir/file2 dir/file3

הדפסת רצף של מספרים

seq = sequence

דוגמה:

דוגמה בסקריפט:



#### [קישור למדריך המלא](https://www.w3schools.io/terminal/bash-tutorials/)