DevOps – Final Task

מגישים:

כפיר אמויאל 205710411

יונתן שרר 318317682

# Introduction

1. התוצאה שקיבלנו לאחר הקלדת pwd בטרמינל ב UBUNTU:
2. 
3. פקודת pwd – **P**rint **W**orking **D**irectory מציגה את הנתיב האבסולוטי מהתיקייה הנוכחית בה אנו נמצאים.

# 2. Linux

שקף 10:

1. כפי שניתן לראות על פי פקודת pwd ה user name שלי הוא kfiros94



1. ה Machine name שלי הוא ubuntoKfiros94

A screen shot of a computer

AI-generated content may be incorrect.

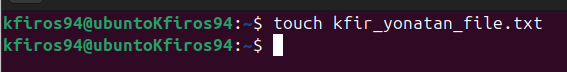
1. לפי צילום המסך בסעיף הקודם ה current directory הוא /home/kfiros94

שקף 26:

בשקף זה אנו מתבקשים ליצור קובץ חדש באמצעות פקודת touch.

יצרנו קובץ טקסט בשם kfir\_yonatan\_file.txt

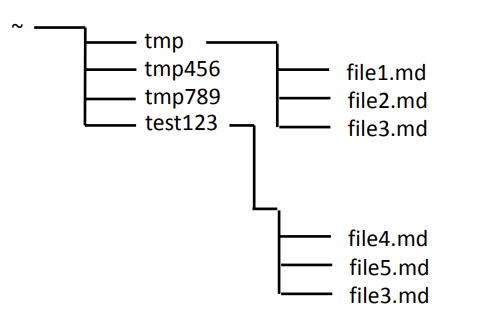
צילום מסך:



A white square with black text

AI-generated content may be incorrect.

לאחר מכן יש class exercise אנו מתבקשים ליצור tree structure לפי התרשים הבא:



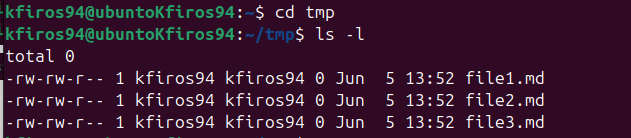
1. יצירת התיקיות באמצעות פקודת mkdir:



1. הוספת קבצים לתיקיית tmp באמצעות פקודת touch:



ואפשר גם לנווט דרך הטרמינל לתיקיית tmp כדי לראות את התוכן (הקבצים החדשים שיצרנו):



השתמשנו בפקודת ls -l כדי לראות את תוכן התיקייה, ואכן הקבצים נוצרו

-l זה דגל כדי לראות את התוכן ב long format כלומר להציג את התוצאה בפורמט מפורט (הרשאות, מספר קישורים, בעלים, קבוצה, גודל קובץ, תאריך שינוי אחרון ושמות הקבצים)

1. נוסיף כעת קבצים לתיקיית test123:



נבדוק הפעם שהקבצים נוספו דרך הUI:

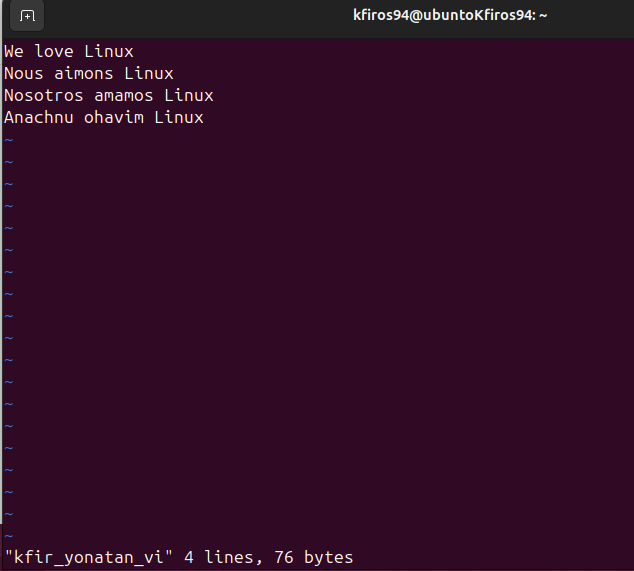
A screenshot of a computer

AI-generated content may be incorrect.

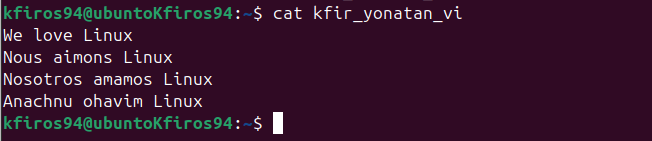
שקף 35:

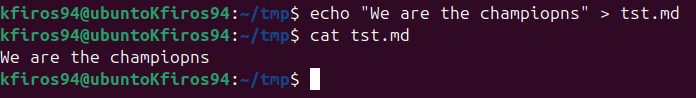
כעת נעשה שימוש בפקודות cat, man ונעשה גם את ה class exercise.

ראשית, ניצור קובץ בשם kfir\_yonatan\_vi ונשתמש ב vi כדי לכתוב תוכן לתוך הקובץ:



כעת נשתמש בפקודת cat כדי לראות את תוכן הקובץ:



נראה גם שימוש בפקודת echo בקובץ test.md שיצרנו שנמצא בתיקיית tmp:  


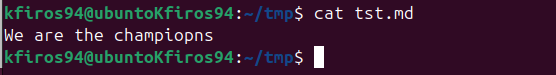
נסביר כעת שימוש בפקודה man עבור הפקודה ls:

הפקודה man = manual פותחת את ספר ההוראות של פקודה מסוימת.

לכן, man ls זה המדריך המלא לשימוש ב ls המציג אילו אפשרויות flags קיימות (למשל -a, -l, -h וכו'..).

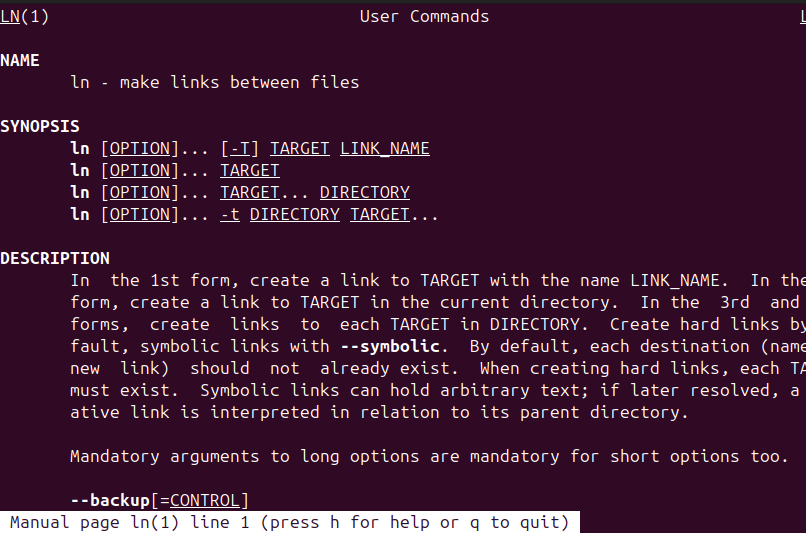
תשובות ל class Exercise:

1. שימוש בפקודה cat:



אנו רואים את התצוגה של tst.md שיצרנו מקודם.

1. הדפסת דוקומנטציה של פקודות ln ו mv (זה ארוך אז נראה רק חלק מהצילום מסך):



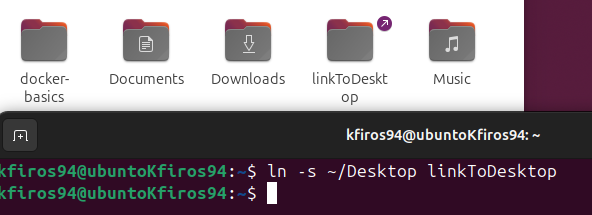
A computer screen with white text

AI-generated content may be incorrect.

1. ניצור soft link ל desktop:

ב Linux קוראים לזה symbolic link

והפקודה שנשתמש בה היא ln -s



Ln היא הפקודה ליצירת לינק

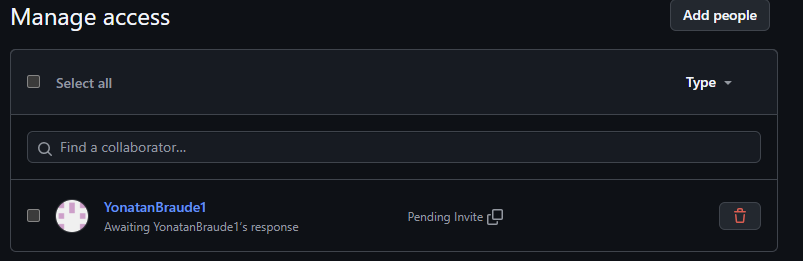
-s אומר לעשות לינק סימבולי

וכל מה שבא אחר כך זה היעד והשם של הקובץ.

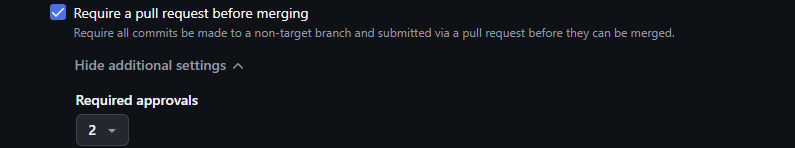
# Git

נבצע את המשימה על ידי כך שנבנה אתר html פשוט, ונשתמש ב repository של המטלה שלנו.

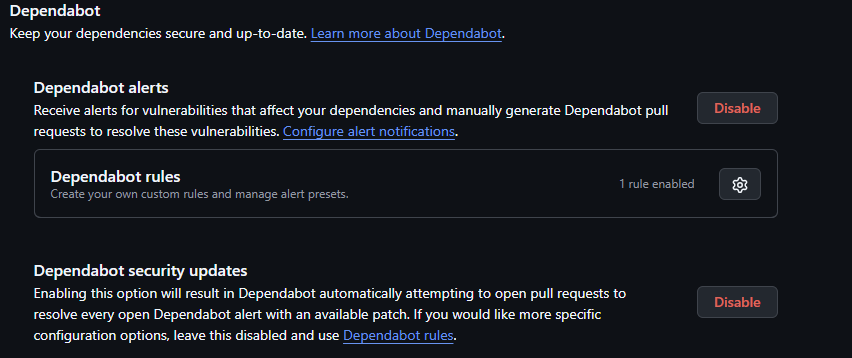
הוספתי את יונתן כ collaborator לריפוזיטורי שלנו:



תחת branches אני יוצר ruleset חדש על מנת לבקש pull request ולקבל לפחות 2 אישורים לפני merging ל main branch.



כעת, תחת security נאפשר את **Dependabot:**

****

הוא סורק באופן אוטומטי את התלויות שלך ויוצר בקשות משיכה (pull requests) כאשר מתגלה גרסה פגיעה או מיושנת, ובכך שומר על הפרויקט מאובטח ומעודכן ללא צורך במאמץ ידני.

1. למה אנחנו צריכים אותו?

* בפרויקטים מודרניים יש המון תלויות (dependencies) — ספריות קוד פתוח, פריימוורקים, כלים.
* לעיתים קרובות, ספריות אלו מתעדכנות: מתקנים בהן פרצות אבטחה, באגים, או מוסיפים שיפורים.
* אם אנחנו לא עוקבים אחרי העדכונים, הפרויקט שלנו עלול לכלול פרצות או לעבוד עם קוד לא יציב.
* Dependabot חוסך לנו את העבודה הידנית של לעקוב אחרי עדכונים. הוא בודק את ה-repository שלנו באופן קבוע ומעדכן אותנו.

1. מתי משתמשים בו?

* בפרויקטים תוכנה שמכילים קבצי ניהול תלות.
* כאשר אנחנו רוצים לשמור על עדכניות ואבטחה בלי צורך לבדוק הכל ידנית.
* מתאים מאוד לפרויקטים ב**-**GitHub, שם הוא משולב מעולה.

1. למה כל כך חשוב להשתמש בו?

* **אבטחה** - מזהה ספריות עם פגיעויות ומציע תיקונים אוטומטיים.
* **תחזוקה** - שומר את הפרויקט מעודכן, מונע "ריקבון תוכנה" (software rot)
* **חיסכון בזמן** - לא צריך לבדוק ידנית אם יצאו עדכונים
* **אוטומציה -** פותח Pull Requests מוכנים עם עדכונים - נשאר רק לבדוק ולמזג.

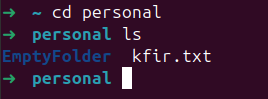
# Ansible

במשימה זו התבקשנו לכתוב סקריפט בלינוקס אשר יעתיק את תוכן תיקיית personal לתיקיית גיבוי ששמה backup\_archive<date> כלומר השם של התיקייה יכיל גם את חותמת הזמן הנוכחית.

**קובץ הקוד מצורף לגיט.**

צילומי מסך:

תוכן התיקייה אותו אני רוצה לגבות:



הרצת הסקריפט:

A computer screen with white text

AI-generated content may be incorrect.

בדיקה שזה אכן גובה:

A screenshot of a computer code

AI-generated content may be incorrect.