**תמר כהן**

**הדרכה על GIT**

Git, Git Bash, GitHub

# מה זה ?

שימוש בGit מאפשר עבודה משותפת על אותו פרויקט, צפיה בשינויים שבוצעו וכו'

# להבנה

גיט שומרת stages של הפרויקט. בכל שלב ניתן לעדכן את הstage הנוכחי או לחזור לstage מהעבר.

עם כל שינוי שבוצע נוכל להגדיר stage חדש שיוגדר להיות הstage הנוכחי ולשלב הודעה המתארת את השינוי בין הstage הנוכחי לstage שיהפוך להיות הנוכחי.

# התקנה

יש המון מדריכים באינטרנט 😊 בחרתי לא להתעסק עם זה פה

**הערה –** אני הורדתי Git Desktop למחשב. אפשר לעבוד ישירות מולו במקום מול הGit Bash, הממשק שלו יותר ידידותי (כפתורים במקום פקודות קוד)

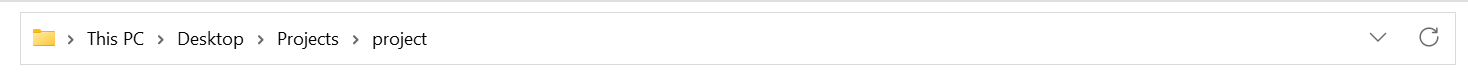
באופן כללי, אני ממליצה להוריד Git Desktop ולסנכרן אותו עם המשתמש בGitHub, זה חוסך שלבים של הכנסת שם משתמש וסיסמא בGit Bash ומזהה ישירות את חשבון הGitHub.

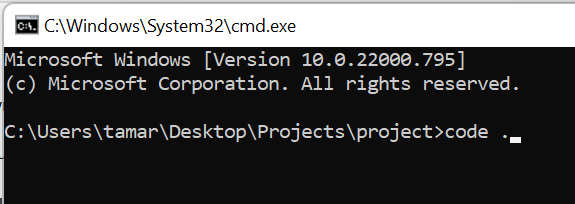
ההדגמות פה מסתמכות על כך שהורדתם Git Desktop למחשב.

# פתיחת פרויקט והעלאתו לגיטהאב

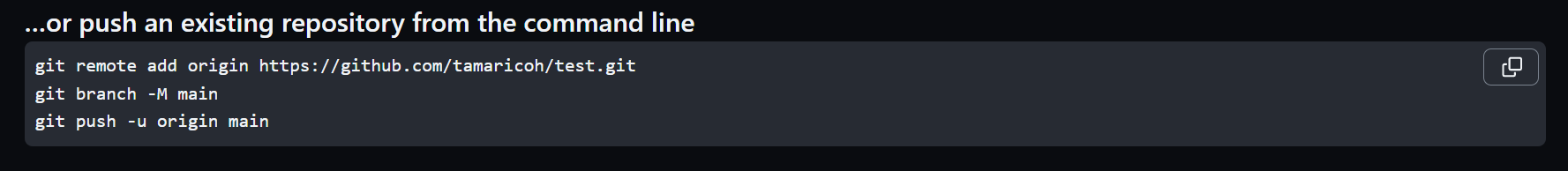
1. ניצור תיקיית פרויקט בPC שלנו ונפתח אותה בVSC

* ע"מ לפתוח בVSC ניתן לפתוח את התיקיה בFile Explorer ולכתוב בשורת הaddress "cmd" . כעת, יפתח במחשב הCommand Prompt ובו נקליד "code ."





1. ניצור בגיטהאב repository חדש ונעתיק את הCommand line שנקבל (git remote)



1. בVSC נפתח דרך הטרמינל את Git Bash ונדביק את הCommand line שהעתקנו בשלב הקודם
2. נריץ פקודות git init בGit Bash על מנת ליצור פרויקט Git חדש

# פקודות בGit Bash

## פקודות לינוקס

* **pwd**

פקודה המחזירה לנו את המיקום הנוכחי שלנו

* **ls**

פקודה המחזירה רשימה של הקבצים בתיקיה בה אנו נמצאים

* **ls -a**

פקודה המחזירה רשימה של הקבצים בתיקיה כולל קבצים במצב hidden

* **cd XXX**

פקודה המבצעת ניתוב לתיקיה פנימית כאשר XXX זו התיקיה הפנימית

* **cd..**

פקודה המבצעת ניתוב לתיקיית האב של התיקיה הנוכחית

* **touch XXX.ZZZ**

פקודה היוצרת קבצים בתיקיה בה אנו נמצאים כאשר XXX זה שם הקובץ וZZZ סוג בקובץ

* **Clear**

פקודה המנקה את מסך הGit Bash

## פקודות גיט

כללי וחשוב

* **git init**

פקודה המגדירה repository בתיקיה

* **git status**

פקודה המציגה את מצב קבצי הפרויקט (אם יש קבצים שלא הועלו לstage או שינויים שלא ביצענו להם commit)

סנכרון בין git מקומי לGitHub

* **git clone**

פקודה לסנכרון בין תיקיה במחשב המקומי לrepository קיים.

* **git push**

פקודה המעדכנת את סטטוס הgit בGitHub לזה של הgit המקומי

* **git pull**

פקודה המעדכנת את סטטוס הgit המקומי לזה של הgit בGitHub

העלאה לstage

* **git add XXX**

פקודה המוסיפה לstage קובץ חדש או קובץ שעבר עדכון כאשר XXX זה שם הקובץ שנרצה להוסיף / לעדכן

* **git add .**

פקודה המוסיפה לstage את כל הקבצים בתיקיה הנוכחית

ביצוע commit לstage (עדכון stage נוכחי)

* **git commit -m "XXX"**

פקודה המעדכנת את הstage הנוכחי (מעדכנת את השינויים שביצענו)

בXXX נהוג לרשום הודעה קצרה המתארת את השינוי שבוצע

למשל Adding readme file

מחיקת קובץ

* **git rm XXX**

פקודה המוחקת קובץ בתיקיה בה אנו נמצאים, כאשר XXX שם הקובץ

הצגת היסטוריית פעולות

* **git log**

פקודה המציגה את היסטוריית הCommits שבוצעו

* **git log --author "XXX"**

פקודה המציגה את היסטוריית הCommits שבוצעו ע"י XXX (כאשר XXX שם משתמש)

* **git log -n X**

פקודה המציגה את X הCommits האחרונים שבוצעו

עבודה עם ענפים

* **git branch**

פקודה המציגה את רשימת הענפים

* **git branch XXX**

פקודה היוצרת ענף חדש בשם XXX

* **git checkout -b XXX**

פקודה היוצרת ענף חדש בשם XXX

* **git checkout XXX**

פקודה המעבירה מהענף הנוכחי לענף XXX

* **git branch -d XXX**

פקודה המוחקת את ענף XXX

שחזור אחורנית

* **git checkout XXX**

פקודה המחזירה את הקובץ XXX למצב בו היה הקובץ בcommit האחרון

* **git checkout .**

פקודה המחזירה את כל הקבצים למצב בו היו בcommit האחרון

* **git reset XXX**

פקודה המוציאה את הקובץ XXX משלב הstage לפני הcommit (כלומר, אחרי הadd)

* **git restore --staged XXX**

פקודה המחזירה את הקובץ XXX משלב הstage האחרון (עושה unstage)

# README.mb

* **כותרות**

כדי ליצור כותרת <h1> בקובץ נוסיף # ורווח לפני המלל

כדי ליצור כותרת <h2> בקובץ נוסיף ## ורווח לפני המלל

וכן הלאה

* **עיצוב טקסט**

\*נטוי\* \_נטוי\_

\*\*מודגש\*\* \_\_מודגש\_\_

* **רשימות**
* רשימה לא מסודרת ניתן ליצור עם \* -
* רשימה מסודרת ניתן ליצור עם 1. וכן הלאה
* **קישור**

כדי ליצור קישור נכתוב [XXX](https://...)

וכך XXX יהיה קישור בתוך הREADME שיוביל לכתובת בסוגריים

* **קופסת טקסט בדוגמא קוד**

``` XXX ``` כאשר XXX זה הדוגמא של הקוד