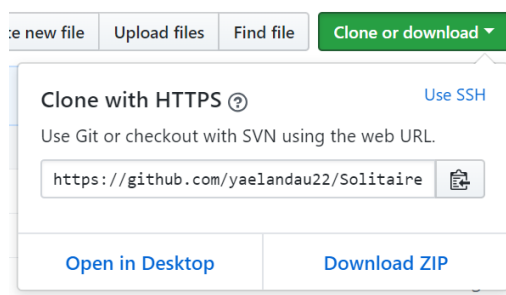


work – flow לעבודה עם git ו-github

הערה: לשימוש במדריך יש להתקין git על המחשב.

פתחתם פרויקט Java באקליפס ואתם מעוניינים לנהל לו גירסאות.

1. פתחו את ה- cmd (או ה- git bash) במיקום של תיקיית הפרויקט והריצו את הפקודה:
git init
מגניב! עכשיו נוצרה לכם בתקיה, תקיית git **נסתרת** שתנהל לכם גירסאות, תשמור גיבויים וכו'.
2. ביצעתם שינויים בפרויקט (שיניתם קוד, כתבתם מחלקות חדשות וכו') ואתם רוצים לשמור גירסה. הריצו את הפקודה הבאה:
git add --all
הפקודה הזו תיקח את כל הקבצים שעברו שינוי מאז שמירת הגרסה הקודמת (או את כולם אם זה השמירה הראשונה) ותעביר אותם לאיזור בשם staging area. באיזור זה יהיו כל הקבצים שנרצה לשמור את השינוי שלהם בגרסה הבאה.
3. סיימתם לבצע שינויים, והעברתם את כולם ל-staging area ואתם מוכנים לביצוע שמירת גרסה, הנקרא commit. הריצו את הפקודה:
git commit -m "some message to describe the changes in this commit"
כרגע ביצעתם שמירת גרסה. אתם יכולים להמשיך לעבוד על הפרויקט שלכם וכל פעם שתרגישו מוכנים לביצוע שמירת גרסה נוספת חזרו על שלב 2 ו-3.
5. כדי להעביר את ה- repository (אוסף ה-commits, גרסאות, שיצרתם) שלכם לענן של github יש קודם כל לפתוח חשבון שם. אח"כ צרו repository חדש באתר שלהם. כאשר נוצר ה-repository תקבלו כתובת url של אותו ה-repository. כמו בתמונה ->
6. חזרו ל- cmd שלכם והגדירו את הגישה לאותו repository מרוחק ע"י הפקודה: (שימו לב שתשנו אותה לפי ה-url שלכם)
git remote add origin https://github.com/yourName/yourRepositoryName
7. עכשיו אחרי שהגדרתם repository מרוחק תרצו כנראה להעלות את הגרסאות שלכם אליו. הריצו את הפקודה:
git push origin master
בשביל ביצוע הפקודה תצטרכו להכניס את השם משתמש שלכם וסיסמה לגיטהאב בכדי לוודא שיש לכם הרשאה לבצע push ל-repository הנ"ל.
8. עשו רפרש לדף שלכם בגיטהאב ותראו שהכל עלה, כולל כל ה-commit הישנים (אם היו כאלה).
9. אם יש לכם חבר נוסף שעובד אתכם על אותו הפרויקט וברצונכם לתת לו הרשאה לשינוי הקוד שב- repository הנ"ל יש ללכת להגדרות בדף הרפוזיטורי בגיטהאב ואז ב-collaborators להוסיף את החבר.
10. כדי שהחבר יוכל לשכפל את ה-repository במחשב שלו, שיפתח את ה-cmd במיקום בו הוא רוצה שישוכפל הפרויקט ושיריץ את הפקודה: (שימו לב שתשנו אותה לפי ה-url שלכם)
git clone https://github.com/yourName/yourRepositoryName
עכשיו כל ה-repository הועתק והחבר יכול גם לבצע שינויים וכו'.



שימו לב! כאשר אתם עושים שינויים במחשב, ושומרים גרסה – זה לא מתעדכן בענן (בגיטהאב) עד שלא תעשו push. לכן מומלץ ישר אחריי כל קומיט לבצע ישר העלאה לענן (במידה ואתם מעוניינים בשמירה בענן..).

בנוסף כדאי לשים לב, שאם החבר שעובד אתכם על אותו הפרויקט עשה שינויים בפרויקט והעלה אותם לענן – השינויים האלו **לא** יופיעו במחשב שלכם עד שלא תדאגו למשוך עדכונים אלו מהענן. ניתן לביצוע ע"י פקודת pull (ראו טבלה בהמשך). לכן, כדי למנוע סיבוכים מיותרים, כדאי להישאר (עד כמה שניתן) מסונכרנים ולבדוק לפני ביצוע שינויים בפרויקט שקודם כל יש לכם במחשב את הגרסה העדכנית ביותר שיש בענן. (כמובן שאם אתם עובדים לבד על הפרויקט אז אין את הבעיה הזו כלל..)

אוסף פקודות:

git init	לאתחל רפוזיטורי בתקיה הנוכחית
git add --all	להוסיף את כל הקבצים שהשתנו מאז הקומיט האחרון אל תוך אזור הסטייג'ינג
git commit -m "some message"	לבצע קומיט. כלומר, לאירוז גרסה חדשה.
git log	להציג את היסטוריית הקומיטים שנעשו עד כה ברפוזיטורי הנוכחי
git add remote origin YourGithubURL	להוסיף הפניה לרפוזיטורי המרוחק שבגיטהאב
git push origin master	להעלות את כל הרפוזיטורי שעשיתם לוקאלית אל הרפוזיטורי המרוחק (הגיטהאב שלכם)
git pull origin master	להוריד את הרפוזיטורי שבענן אל המחשב שלכם
git clone SomeGithubURLToCopy	להעתיק את תוכן הריפוזיטורי של פרויקט כלשהו אל המחשב שלכם
git checkout someCommitID	"לקפוץ" לגרסה ישנה כלשהי. ניתן לראות את המספר המזהה של קומיטים ישנים ע"י פקודת הלוג.
git diff commitID commitID	הצגת השינויי בין 2 קומיטים. הקומיט הראשון צריך להיות הישן יותר.
git branch branchName	יוצר ענף, שיתפצל החל מהקומיט הנוכחי עליו נמצאים בזמן הרצת הפקודה, בשם שתכתבו בפקודה עצמה.
git branch	יציג את שמות כל הענפים הקיימים. כדי "לקפוץ" לענף יש להשתמש בפקודת צ'קאאוט עם שם הענף.
git log --graph --oneline --all	תצוגה גרפית של כל הקומיטים והענפים שברפוזיטורי הנוכחי

יש כמובן עוד המון המון פקודות משיכול המסמך להכיל ☺
בהצלחה!