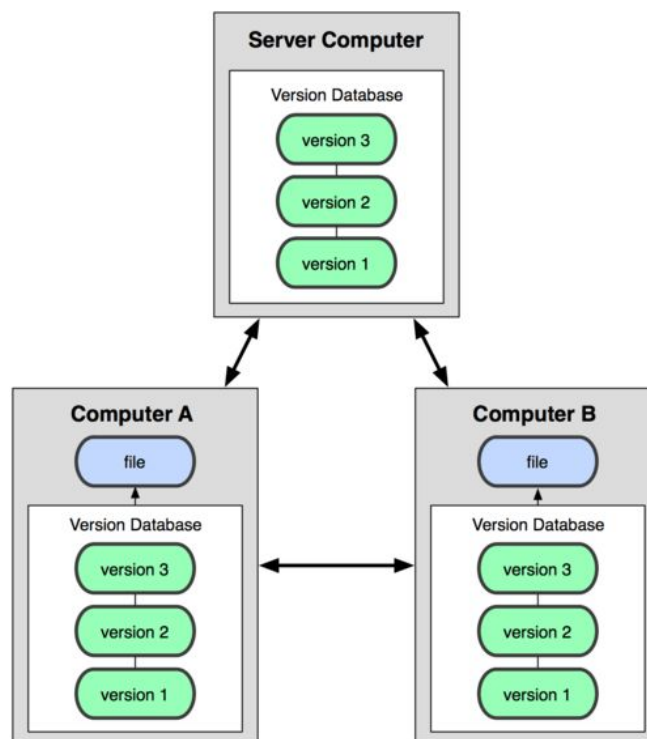


ניהול גרסאות

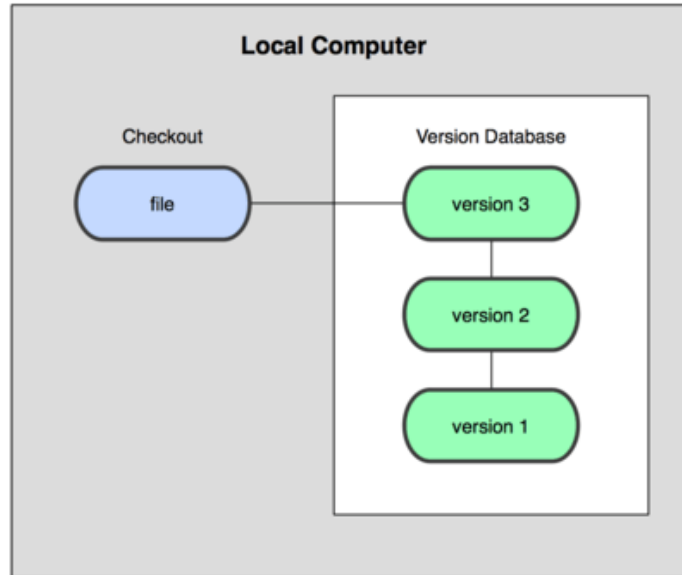
ניהול גרסאות הינו כלי אשר שומר שינויים(משימות/פונקציות שמומשו עד הסוף, עברו קומפילציה ועובדות) שבוצעו לפרויקט עליו עובדים. וידע במידת הצורך להחזיר את הקוד לגרסה הקודמת. כך שאם הגרסה החדשה אינה עובדת כמו שצריך(יש בה באגים, או אי תאימות לדרישות) ניתן יהיה במספר פעולות פשוטות להחזיר את הגרסה האחרונה שעבדה ולהתחיל את המשימה מחדש.

כמו כן ניתן לעבוד במקביל על כמה משימות ולאחר שהן מוכנות ניתן יהיה למזגן לגרסה אחת.

הערה: ניהול גרסאות אינו מיועד לשמירת גיבויים של הקוד.



בתרגיל הבא נבנה ניהול גרסאות פשוט, אשר יודע להחזיר את הגרסה הנוכחית, את הגרסה הראשית, את כל הגרסאות שיש, התוכנית תדע לקלוט גרסה חדשה וידע להחליף בין הגרסה הנוכחית לגרסה שתבחר ע"י המשתמש.



המידע שישמר עבור כל גרסה הוא:

- מזהה של הגרסה (מספר רנדומלי/ או סידורי)
- שם הגרסה
- תאריך ביצוע השינוי
- שם המשתמש שעדכן את הגרסה
- מספר שינויים לעומת הגרסה הקודמת (שורות קוד שהשתנו)

עקרון העבודה של ניהול גרסאות המופשט:

לכל פרויקט ישנו ניהול גרסה משלו.

יצירת פרויקט ריק מאתחלת את הפוינטרים באפס.

הוספת גרסה לפרויקט:

- מאגדת את הנתונים על הגרסה
- פוינטר של הגרסה הראשית יצביע על הגרסה החדשה בתנאי שהוא הצביע מקודם על הגרסה הקודמת לו.
- במקרה ולא הפוינטר ישאר במקומו.

שינוי גרסה נוכחית או ראשית:

- הפונקציה תקבל מזהה של הגרסה אליה רוצים להזיז את הפוינטר ותחליף בהתאם את המצביע הרלוונטי.

- במידה ונרצה לשנות את הגרסה הנוכחית לגרסה הראשית או להפך תהיה פונקציה נפרדת שתדע לעשות זאת.

הצגת הגרסאות:

ישנן שלוש פונקציות הדפסה:

- פונקציה שתדפיס על המסך את הגרסה הנוכחית
- פונקציה שתדפיס על המסך את הגרסה הראשית
- פונקציה שתדפיס על המסך את כל הגרסאות (ליד גרסה ראשית תודפס הודעה שהיא הראשית וליד הגרסה הנוכחית יודפס שהיא נוכחית חוץ ממקרה שהיא גם הראשית - במקרה זה יודפס רק שהפונקציה היא ראשית)

**ממשו את עקרונות העבודה הנ"ל והכינו תפריט מתאים.
בהצלחה.**