

הקדמה מוכנה - הרצאה 9 דרישה 2 - TDD (תצורה)

עדיף לכתוב דרישה רחבה וקלה מאשר אנסטר ולגמל את התקלה
בדרך הישירה, עם דרישה קלה יותר ועדוק אחר סיננים.

FIRST - ראשי תיבות מתוארות - מתי דרישה יחידה (אזהר?)

Fast: מהירות וקלה מאוד

INDEPENDENT: ב דרישה אינה גלויה בדרישה אחרת/זוהי חסני

REPEATABLE: האם הדרישה עקבית? אם נרף ח פעמים,

האם נקבל את אותה התוצאה? קטור גלוייה דגורמים חזוניים
כגון שמים, למני חנה וכו'... (נצב שחזרה תהיה גלוייה
גלויה דגורמים חזוניים כבי לחץ את האופן בתוצאותיה.

SELF-CHECKING: אם ניתן להרחיף את הדרישה באופן אוטומטי.

ולא זריק המערכת אנוסה כבי לזעזע את הדרישה עזרה / או עזרה.

TIMELY: מתי לכתוב את הדרישה, לפני, מתוך, אחריו.

(TDD אמר לכתוב את הדרישה לפני.)

TDD

כתמים test שנועד ליפול (כי עדיין את קוד),

השד הידא נכתוב קוד ראשוני על מנת להעביר את הדרישה (באופן חידוי).

השד האחרון נקבע סיננים וסיפורים דקוד.

הערה: נצבף לדבדב את הדרישה מכלייה חזונייה, (באחר את האליקט)
ולא גלוייה כבי שהדרישה געזור מחד.

הדגמה (דוגמה חנוכה) בי 4, 1 PART חזרה)

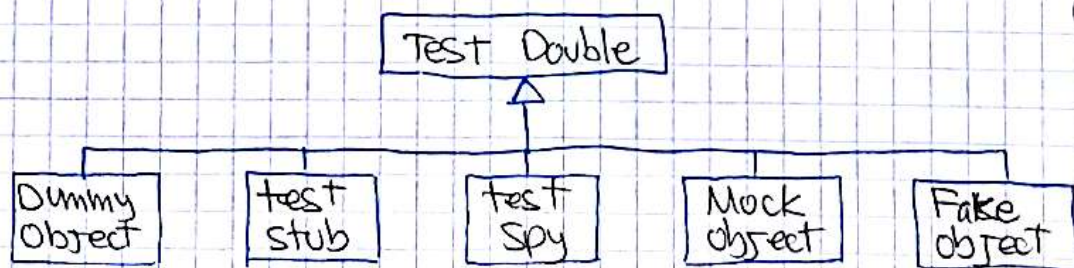
קוראים לספרייה chai שנוסה את הקבירה בתוך ה- mocha.
הפרוקה describe עושה את הקבירה.

השלב הראשון ה- test יכל, כי אין פונקציה Court Summary.
נעשה שני דקוד, שרק מחזיר את הקבירה, (return),
השלב השני אמור להיות שיפור (ההדגמה הנ"ל - אין מה לשפר)

דוגמה ספרייה נוספת ד- JavaScript למקרה, דמיון
לחלק את המערכת, נוצר מחלקים עם התוצאה חזרה
(דוגמה מההדגמה: tax, calculate)

Test Doubles

בוסר נוצר לקבצת דקוד עם קוד מסוים, שכלי מחלקים חזונים,
וכן כפי שצגה את אמה עקובה (כמו המחלקים) ולא שמתה,
לא עם אמה לקבצת את הקבירה ולחלק את אמה תוצאותיה,
ולא קוד חזרה התוצאה.



דוגמה דקוד 2: דקוד של Alarm וכלי חזון אמה.

מחלקים לדקוד עם שטחה תנועה של אדם עם חזון.

החזון לא עוזר כרגע (לא מחלק כרגע, לא חזון).

נראה לפונקציה שמחזיר את החזונים שנובה (לדגם עזר בלחן לה וס...)

למה דקוד 19: משובח (כונה: 1).

סיכום מושגים:

Dummy - ערך שנזרם עבור קריאה למונח, אך לא משמש את הקבירה

Stub - מחזיר תשובה ללא שאלה, ולא קושיקה.

Spy - מאפשר לבדוק מה קרה בתוך הקבירה

Mock - כולל את הקונטאים ומשמש לבדיקת התנהגות.

fake - מייצג אתי אך בשם יאיר.

האם כדאי להשתמש?

דעו: - מחזיר תוצאה, ביצוע מלא

- מאפשר תיכון ממשך

(דעו: - סימול וקריאה

- לא קונקרים את הקבירה האמיתית

- מצריך נסיון בתיכון

- בקוד קיים לא תמיד אפשרי

כסוי קוד ע"י קבירה - איך נוצר אם עדיין מספיק קבירה?

קופסה שחורה - לא מעניין מה יש בפנים, רק שמעשה את העבודה.

קופסה לבנה - אפשר לנו שהקוד כתוב כראוי.

כסוי קוד ע"י פיסקל: לא שורה יש משום שדוק אותה

(Co - ב שורה (קבירה)