מקרי שימוש

1. משתמש מקבל את האלמנט ואז עושה השמה ל מאפיין input למשתנה פשוט – עובד.
2. משתמש מקבל את האלמנט ואז עושה השמה ל מאפיין input למשתנה מסוג Bollean- עובד.
3. משתמש מקבל את האלמנט ואז עושה השמה ל מאפיין input למשתנה מסוג Function – עובד.

בכל המקרים האמורים מקבלים עדכון על ערך חדש...

מקרי שימוש עם אלמנטים חיצוניים.

1. משתמש יעשה השמה (דרך java script) של ערך פשוט 5 והוא ישתקף על הattribute input.
2. משתמש יעשה השמה (דרך java script של ערך החוזר מפונקציה) והערך החוזר ישתקף על המאפיין.

על מי לבצע?

אם הערך מגיע מבושל למודול שאחראי על attributes אזי אין צורך במודול של attribute כי פונקציית setAttribute קיימת ואמורה להספיק.

אם set attribute צריכה לבשל את הערך לדוגמה במקרה שזו פונקציה לבצע אותה אזי יש לכתוב מודול של Attribute שכשעושים set יצטרך להתחשב במקרים השונים לדוגמה כשהערך המגיע אליו שווה לפונקציה.

עבור מקרי שימוש עם אלמנטים חיצוניים אין שום צורך לכתוב מודול של attribute כי כשמשתמש עושה השמה של מאפיינים הערך משתבשל במודול של מאפיינים.

אבל.... מה יקרה אם נוסיף את הדרישה הבאה.

1. משתמש יעשה השמה ל attribute וכתוצאה מכך ישתנה המאפיין.

כלומר אותה דרישה רק הפוכה כאן וודאי שמודול של attribute צריך לדעת לפענח את הנתונים ואין לו לסמוך על מודול מאפיינים.

מסקנה.

יש לכתוב מודול של attribute שיודע לפענח משתנים (לדוגמה לקחת פונקציה ולהפעיל אותה).

משהגענו לכדי כך הוי אומר שצריך מודול חדש היודע לפענח משתנים ששני המודולים מאפיינים ו attributes צורכים.