**תיעוד התכנית**

1. הטענת קובץ EXEL התקנה- npm xlsx

פונקציה onFileChange(event) בקומפוננט ***SeetComponent***

1. עובר על מערך הנתונים ובודק חפיפה של הסעיפים כך:

אם השדה לא רלוונטי לבדיקה לפי השדה isTitleRefRelevant או לפי השדה IsTitleRelevant – יוצא ללא בדיקה

אם אחד מכיל את המילה חוק והשדה השני מכיל את המילה פקודה(פקוד) –מסומן כשגוי

עובר על הref וה title ובודק כמה מילים שמורות קיימות בערך מכיוון שיכול להיות מקרה של פרק .. סימן .. סעיף

מערך המילים השמורות:

const bef\_num = ['תקנות', 'תקנה', 'סימן', 'סעיף', 'חלק', 'סעיפים', 'ו־', 'פרק'];

אם מכיל מילה אחת לא מתחשב במילה עצמה אלא משווה את התוכן שאחריה

אם מכיל יותר ממילה אחת בודק לפי תוכן המילה

לדוג:

* :**Ref** סעיף\_6 **title**: ו־9 -לא יתייחס איזה מילה אלה למספר או לאות שקיים אחריה
* :**Ref** סימן ב׳ לפרק ה׳ בחלק ד׳ לחוק חדלות פירעון ושיקום כלכלי

**title**: חוק חדלות פירעון ושיקום כלכלי#חלק ד פרק ה סימן ב

במקרה זה יתחשב בסוג המילה

בכל השוואה בין המספרים או האותיות חותך מהמילה השמורה +1 עד התו שאחרי המספר/ המילה לפי מערך אופציות סגור-

['׳',' ','(',')','#',"'"]

פונקציה checkData() בקומפוננט ***SeetComponent***

1. מציג את התוצאות בטבלה הניתנת לחיפוש – ע"י שימוש ב @input()- קבלת מידע מקומפוננט האב
2. חיפוש בטבלה-

פונקציה filter\_data() בקומפוננט ***TableRowComponent***

1. מציג את נתוני הטבלה במטריצת הבלבול ואת רמת הדיוק לפי הבדיקה

העברת מידע בין קומפוננטות ע"י קומפוננט האב תוך שימוש ב @output(), @input()

כדי שיתעדכן בכל חיפוש שהמשתמש יעשה

On @input() set בקומפוננט ***DetailsComponent***

1. כפתור להורדת הEXEL עם תוצאות הבדיקה

פונקציה export בקומפוננט ***SeetComponent***

שימוש בספרית angular material

Component

1. Seet: כפתור לטעינה המוסתר כאשר קובץ הוטען

* Details: פרטי הטבלה- נתוני הטבלה(מטריצת בלבול ורמת הדיוק)
* Table: הטבלה

כל הספריות שהותקנו הותקנו דרך הterminal



