

פרויקט אנדרואיד 236271 – יבש 1

מגיש: יובל מרדכי בן הרוש 318659869

חלק א:

1. שורות הקוד הבאות הן אלו שיוצרות אשליה של רשימה אינסופית ע"י בדיקת האינדקס איתו רוצים לגשת לרשימה ואם האינדקס גדול מאורך הרשימה אז יוצרים 10 איברים חדשים לרשימה

```
if (index >= _suggestions.length) {  
  _suggestions.addAll(generateWordPairs().take(10));  
}
```

אם נסיר את השורות הללו ונגלול למטה הרשימה לא תגדל ויהיה ניסיון גישה מחוץ לגבולות הרשימה (Range Error):

```
RangeError (index): Invalid  
value: Not in inclusive  
range 0..9: 10  
See also:  
https://flutter.dev/docs/testing/errors
```

2. שיטה נוספת להפריד בין אברים ב ListVeiw היא ע"י שימוש במתודה separated ע"י העברת פונקציה אנונימית בתור ארגומנט בשדה ListSeperatorBuilder שבאופן מובנה מפרידה בין אברי הרשימה ע"י קריאה לאותה פונקציה אנונימית שהעברנו שמחזירה Divider

```
separatorBuilder: (BuildContext context, int index) => const Divider(),
```

זו שיטה עדיפה כי היא קריאה יותר, "נקיה" יותר ונעזרת בפיצ'ר מובנה בקלאס של ListView כמו כן, במימוש המקורי שלנו ה Divider נחשב כאיבר ברשימה לכל דבר וברשימה של 100 איברים מחצית מהם יהיו Dividers בעוד בשיטה השנייה יש הפרדה ברורה בין itemBuilder ל SeparatorBuilder

3. כל איבר ברשימה משנה את תצוגתו ע"י לחיצה על אייקון הלב (הלב הופך משקוף לצבע אדום) וגם מתווסף לרשימת המועדפים וכל שינוי כזה דורש קריאה ל Set_state כדי לטעון מחדש את ה widget במצבו החדש.

חלק ב:

1. MaterialApp הוא ווידג'ט שכולל בתוכו מספר ווידג'טים נפוצים הנחוצים לצורך תפקוד אפליקציה בסיסית ומשתמש בפונקציות ב material-design.
לווידג'ט זה מספר שדות ביניהם:
א. actions - מאפשר לדרוס מיפוי של פעולות דיפולטיביות ולספק מיפוי אחר ע"י קריאה לפונקציית callback שתטפל בבקשה לביצוע הפעולה
ב. Home – מאפשר להגדיר את הווידג'ט הראשית/התחלתי שיופיע בהצגת ה-MaterialApp
ג. Color – הצבע המרכזי שיאפיין את האפליקציה ואת הממשק שלה
2. שדה key מאפשר לשלוט באיך ווידג'ט אחד יחליף אחר ב- widget tree ע"י מתן מזהה key ייחודי לכל אלמנט כך שאם למשל יתווסף אלמנט Dismissible אחר עם אותו key אז הוא יעדכן את האלמנט הקיים ולא יוסיף אחר ל widget tree.