# המשך REACT

## תכנית העבודה:

1. **JSX**
2. **Components**
3. **Events**
4. Hooks

כשנסיים:

1. שילוב של REACT ו backend API

## JSX

דוגמא:

<script type="text/babel">

function Hello() {

return <h1>Hello World!</h1>;

}

זה HTML בתוך JS.

## Components

זה פונקציה שמכילה JSX.

אפשר להשתמש בשם של הפונקציה כדי ליצור אלמנט חדש.

דוגמא: ניקח את ה JSX הפשוט שעשינו ונהפוך אותו לקומפוננטה (פונקציונלית).

## Events

תרגילים:

1. כתבו רכיב שמקבל prop מספר כלשהו, כשלוחצים בודק אם המספר גדול מ 100 צובע את ה div באדום, אחרת בירוק.

<div id="myDiv"></div>

myDiv.style.backgroundColor="red";

1. כנ"ל רק שהמספר לבדיקה מגיע מ input.

<input id="myInput"/>

Num=myInput.value

1. כנ"ל רק להכניס ב input צבע ולשנות את הצבע לצבע שהתקבל.
2. רכיב שמקבל גיל ואם מתחת לגיל 18 צובע באדום אחרת בירוק
3. רכיב שמקבל קוד ואם הקוד נכון מדפיס "קיבלת 10% הנחה", ב prop להעביר את הקוד הנכון
4. כנ"ל רק לקבל קוד קופון ואם הקוד נכון להציג תמונה מתאימה (לשלוח ב prop מילון עם קוד ותמונה)
5. רכיב שמקבל רשימה של רחובות ובודק מול input אם רחוב קיים צובע בירוק אחרת אדום.

* onChange קורה ברגע שמכניס את הערך ל input

**תרגיל 15:**

1. להגדיר רשימה של מילונים עם todos:
   1. שם משימה
   2. תאריך
      1. Date("13-02-2023")
   3. תיאור
2. לכתוב רכיב שמציגה רשימה (טבלה) עם ה todos
3. לעצב את הרשימה עם CSS
4. להדפיס כמה משימות יש (מספר)
   1. List.length
5. לבדוק האם התאריך של המשימה עבר, כלומר הוא קטן מהתאריך של היום. אם כן להציג באדום
6. להוסיף לאובייקט תכונה completed כן או לא. אם כן להציג בירוק.
7. להוסיף רכיב נפרד שמציג את מספר המשימות שהושלמו (תסתכלו על המתודה filter של array)
8. להוסיף רכיב נפרד שמציג את מספר המשימות שמאחרות

**תרגיל 16:**

1. לכתוב רכיב שמציג רשימה של אנשי קשר (מילונים עם שם וטלפון), ו תיבת חיפוש
2. אם השם שמכניסים בחיפוש קיים ברשימה, לצבוע אותו בצהוב.