**Events – React**

לצורך המחשה ניצור פונקציונליות של פתיחת וסגירה של עגלת קניות פשוטה ע"י כפתור

1. בקומפוננטה שבה קיים האלמנט שבו נתמקד ניצור Interface לצורך ציון Flag שיקבל ערך בוליאני (True / False ) ועל פיו הלוגיקה תעבוד ובנוסף, ניצור סטייט לפתיחה וסגירת העגלה.
2. import React, { FC, useState } from "react";
3. import styles from "./ShoppingCart.module.scss";
4. interface CartProps {}
5. interface Flag {
6. isOpen: boolean;
7. }
8. const Cart: React.FC<CartProps> = () => {
9. const [isOpen, setIsOpen] = useState(false);
10. return (
11. <div className={styles.ShoppingCart}>
12. <button>2 pizza(s)</button>
13. <div className={styles.listContainer}>
14. <ul className={styles.cartList}>
15. <h4>Shopping Cart:</h4>
16. <li className={styles.listItem}>Napolitana</li>
17. <li className={styles.listItem}>Marinara</li>
18. </ul>
19. </div>
20. </div>
21. );
22. };
23. export default Cart;
24. כעת, בכפתור נוסיף Event – onClick ונכתוב את הלוגיקה.  
    בדוגמא כתבתי את הלוגיקה בתוך הסוגריים אבל ניתן ועדיף לכתוב פונקציה בשדה הקומפוננטה.
25. import React, { FC, useState } from "react";
26. import styles from "./ShoppingCart.module.scss";
27. interface CartProps {}
28. interface Flag {
29. isOpen: boolean;
30. }
31. const Cart: React.FC<CartProps> = () => {
32. const [isOpen, setIsOpen] = useState(false);
33. return (
34. <div className={styles.ShoppingCart}>
35. <button onClick={() => {setIsOpen((prevState) => !prevState)}}>2 pizza(s)</button>
36. <div className={styles.listContainer}>
37. <ul className={styles.cartList}>
38. <h4>Shopping Cart:</h4>
39. <li className={styles.listItem}>Napolitana</li>
40. <li className={styles.listItem}>Marinara</li>
41. </ul>
42. </div>
43. </div>
44. );
45. };
46. export default Cart;
47. כעת, כדי להחל את הפונקציונליות נוכל להשתמש בstyle ולעבוד עם Display על מנת לקבל את התוצאה הרצויה של פתיחה וסגירה. שימו לב יש להחל את הdisplay על האלמנט הרצוי ולא על האלמנט הראשי של הקומפוננטה כי כל הקומפוננטה תיעלם.  
    בדוגמא הזו נשתמש בternary operator .
48. import React, { FC, useState } from "react";
49. import styles from "./ShoppingCart.module.scss";
50. interface CartProps {}
51. interface Flag {
52. isOpen: boolean;
53. }
54. const Cart: React.FC<CartProps> = () => {
55. const [isOpen, setIsOpen] = useState(false);
56. return (
57. <div className={styles.ShoppingCart}>
58. <button onClick={() => {setIsOpen((prevState) => !prevState)}}>2 pizza(s)</button>
59. <div style={{display: isOpen ? "block" : "none"}} className={styles.listContainer} >
60. <ul className={styles.cartList}>
61. <h4>Shopping Cart:</h4>
62. <li className={styles.listItem}>Napolitana</li>
63. <li className={styles.listItem}>Marinara</li>
64. </ul>
65. </div>
66. </div>
67. );
68. };
69. export default Cart;

**זהו!**

כעת העגלה תפתח ותסגר בלחיצת כפתור!

שימו לב: בדוגמא הזו אראה איך ניתן ליצור Event שלא בתוך האלמנט המרונדר אלא כפונקציה שדורשת generics

* במקום הלוגיקה בonCick שבתוך האלמנט יש להעביר handleClick וליצור פונקציה בשם handleClick. בפרמטר הפונקציה יש להזין e או event או כל שם אחר שתרצו לפרמטר ולאחר מכן לקבוע type ע"י נקודתיים ולכתוב את הtype הרצוי (נכון לעכשיו: React.MouseEvent). הgeneric השני שנצטרך להעביר הוא על איזה סוג אלמנט נעבוד עם הevent וכמובן לא נשכח להוסיף type של הפונקציה שכמו כן לא נרצה כרגע שתחזיר כלום לכן יהיה void. והפרמטר יראה כך: React.MouseEvent<HTMLButtonElement>) :void:e)

import React, { FC, useState } from "react";

import styles from "./ShoppingCart.module.scss";

interface CartProps {}

interface Flag {

  isOpen: boolean;

}

const Cart: React.FC<CartProps> = () => {

  const [isOpen, setIsOpen] = useState(false);

const handleClick = (e: React.MouseEvent<HTMLButtonElement>) :void => {

  setIsOpen((prevState) => !prevState)

}

  return (

    <div className={styles.ShoppingCart}>

      <button onClick={handleClick}>2 pizza(s)</button>

      <div style={{display: isOpen ? "block" : "none"}} className={styles.listContainer} >

        <ul className={styles.cartList}>

          <h4>Shopping Cart:</h4>

          <li className={styles.listItem}>Napolitana</li>

          <li className={styles.listItem}>Marinara</li>

        </ul>

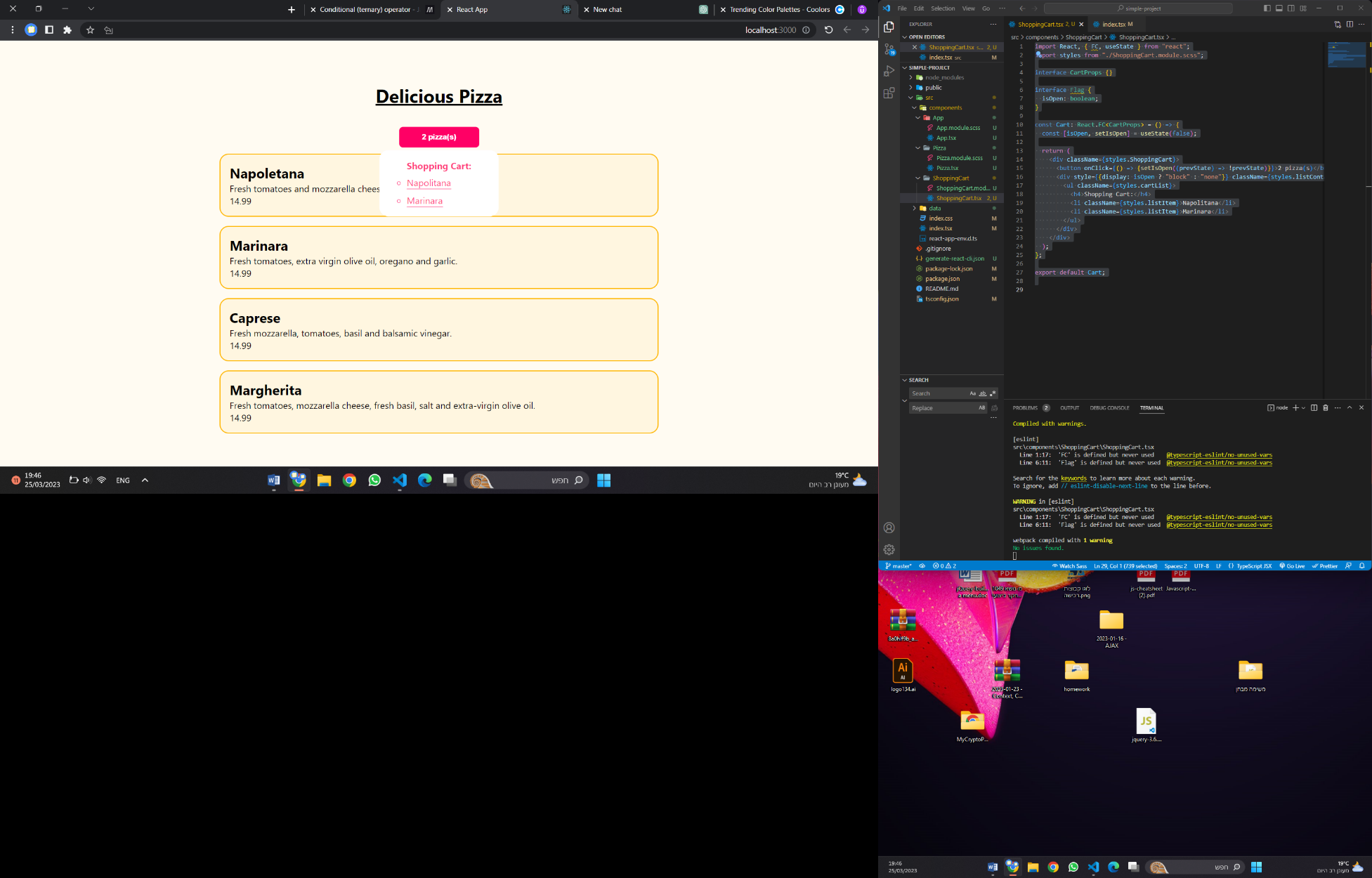
      </div>

    </div>

  );

};

export default Cart;

התוצאה לפניכם :

