

# סיכום שיעור קודם

---

- **תרגיל פתיחה - פוקימון**

# JSON Local Storage

## היום בהרצאה

---

- JSON
- localStorage
- include a header across multiple pages
- Query string

# JSON (JavaScript object notation)

---

JSON הוא פורמט לשימרת מידע:

- המידע מוחזק בזוגות של key: value עם נקודותים ביניהם , key:value , key:value על ידי פסיק
- מתחילה ונסגר בסוגרים מסולסלים {}
- יכול להכיל מערכים, המרכיבים מוגדרים על ידי סוגרים מרובעים []

# JSON Keys

---

JSON key is string :

```
{"name": "John"}
```

# JSON Values

---

- String : {"name": "John"}
- Number : {"age": 30}
- Object : {"employee": {"name": "John", "age": 30, "city": "New York"}}
- Array : {"employees": ["John", "Anna", "Peter"]}
- Boolean : {"sale": true}
- Null : {"middlename": null}

```
let superHeroes = {  
    "squadName": "Super hero squad",  
    "homeTown": "Metro City",  
    "formed": 2016,  
    "secretBase": "Super tower",  
    "active": true,  
    "members": [  
        {  
            "name": "Molecule Man",  
            "age": 29,  
            "secretIdentity": "Dan Jukes",  
            "powers": ["Radiation resistance", "Turning tiny", "Radiation blast"]  
        },  
        {  
            "name": "Madame Uppercut",  
            "age": 39,  
            "secretIdentity": "Jane Wilson",  
            "powers": [  
                "Million tonne punch",  
                "Damage resistance",  
                "Superhuman reflexes"  
            ]  
        },  
    ]  
}
```

JSON structure

מה ידgo?

```
console.log(superHeroes.homeTown);  
console.log(superHeroes['active']);  
console.log(superHeroes['members'][1]['powers'][2]);
```

```
let arr = [
  {
    "name": "Molecule Man",
    "age": 29,
    "secretIdentity": "Dan Jukes",
    "powers": ["Radiation resistance", "Turning tiny", "Radiation blast"]
  },
  {
    "name": "Madame Uppercut",
    "age": 39,
    "secretIdentity": "Jane Wilson",
    "powers": [
      "Million tonne punch",
      "Damage resistance",
      "Superhuman reflexes"
    ]
  }
];
```

הדף את השדה : "Million tonne punch"

# תרגיל חימום

---



thai

Bangkok, 100

King Thai, 50

thai house, 35

zozo rest, 50

TomYam house, 48

Masaman, 33



indian

Chapati, 15

Delhi Kitchen, 60

Pushkar delicates, 25

Bombay fire, 30

בצע את הפעולות הבאות על ה JSON הנוכחי בשלד

JSONExPreRestConsole\_Skelton

.1 הדפו את סוגי המסעדות ("thai", "indian")

.2 הדפו את סוגי המסעדות ומתחילה הדפו את רשימת המסעדות

לכל סוג ומחיר.

# פתרונות

---

```
for (let key in amsterdam["resturants"]) {  
    console.log(key);  
  
    for (let i in amsterdam.resturants[key]) {  
        console.log(amsterdam.resturants[key][i]);  
    }  
}
```

## תרגיל ביתה המשך

---

- יש ליצור באופן דינامي רדיו לכל סוג מסעדה שקיים בJSON
- כאשר המשתמש ילחץ על סוג מסעדה מסוים (checked) יש להציג את שמות כל המסעדות מסוג זה.
- בלחיצה נוספת (unchecked) יעלמו המסעדות מסוג זה.

# Cheep Restaurant

Choose a type of restaurant for see list of resturants:

- thai
- indian

# Cheep Restaurant

Choose a type of restaurant for see list of resturants:

- thai
- indian

Bangkok: 100

King Thai: 50

thai house: 35

zozo rest: 50

TomYam house: 48

Masaman: 33

# JSON במחוזת ובובייקט

---

פורמט `sojs` מאוד דומה ל `javascript` אך קל להמיר אובייקטים של `JSON` ל `Javascript` ולהפך:

- המרת `JSON` ל `Javascript`

**`JSON.parse()`**

- המרת `Javascript` ל `JSON`

**`JSON.stringify()`**

# JSON.Parse

```
<body>

    <h2>Creating an Object from a JSON String</h2>

    <p id="demo"></p>
    <p id="demo1"></p>

    <script>

        const txt = '{"name":"John", "age":30, "city":"New York"}'
        const obj = JSON.parse(txt);
        document.getElementById("demo").innerHTML = obj.name + ", " + obj.age;

        const text = '{"cars" : [ "Ford", "BMW", "Audi", "Fiat" ]}';
        const obj1 = JSON.parse(text);
        document.getElementById("demo1").innerHTML = obj1.cars[0];

    </script>

</body>
```

# תאריכים ופונקציות

---

- **תאריך יש להביר כמחרוזת**

למשל: "birth": "1986-12-14"

\*(אחכ ניתן להמיר ל date ב javascript על ידי שימוש באובייקט Date)

- **פונקציה יש להביר כמחרוזת**

למשל: "age": "function () {return 30;}"

\*(אחכ ניתן להמיר לפונקציה על ידי שימוש ב eval של javascript)

```
<script>
  const txt = `{"name":"John", "age":function () {return 30;}, "city":"New York"}`;
  const obj = JSON.parse(txt);

  // Evaluate the string as a function using eval
  const ageFunction = eval('(' + obj.age + ')');

  // Call the function
  const age = ageFunction();
  console.log(age); // Output: 30

  //-----
  const txt1 = `{"name":"John", "date":"01/01/2025", "city":"New York"}`;
  const obj1 = JSON.parse(txt1);

  // Evaluate the string as a date
  const date = Date(obj1.date);
  console.log(date);
</script>
```

# JSON.stringify

---

```
<body>

    <h2>Create a JSON string from a JavaScript object.</h2>
    <p id="demo"></p>
    <script>
        const obj = {name: "John", age: 30, city: "New York"};
        const myJSON = JSON.stringify(obj);
        document.getElementById("demo").innerHTML = myJSON;
    </script>
</body>
```

---

# העברה נתונים בין דפים

## העברת נתונים בין דפים

---

יש מספר דרכים להעברת נתונים בין דפים בפיתוחצד לקוח, הדרכים העיקריות:

1. **localStorage** – שמירת מידע פר אטר (דומיין) על מחשב הלקוח. מאובטח כלפי יוצרים אחרים – העברת מידע סמויה.

2. **Query string** – העברת נתונים דרך ה-URL - [www.mySite.co.il?username=yael](http://www.mySite.co.il?username=yael) (נסתכל רק על דוגמא) משמש להעברת מידע לא מסויים.

3. **cookie** – מידע נשמר על מחשב הלקוח וועבר לשרת בכל פניה. (לא ניגע בקורס זה)

# localStorage

-מחסן מקומי לשימירת נתונים העובד בצורה של key\value.

-בכל צמד של key\value, הערך יישמר **כמחרוזת**.

The screenshot shows the Chrome DevTools interface with the Application tab selected. In the left sidebar under 'Application', 'Manifest', 'Service Workers', and 'Storage' are listed. Under 'Storage', 'Local Storage' is expanded, showing an entry for 'http://localhost:59445'. The main area displays a table with 'lastname' as the key and 'Smith' as the value. Red circles highlight the 'Application' tab, the 'lastname' key, and the 'Smith' value.

Key	Value
lastname	Smith

-המחסן בעל נפח אחסון של 5 MB.

# localStorage

---

- עלרכים אין תאריך תפוגה עד שהם לא נמחקים.
- גם אם נסגור את הדף ונפתח אותו שוב הערך יהיה קיים! (כנ"ל אם נכבה את המחשב).
- כל הדפים באותו הדומיין יכירו את המידע ששמרנו.



# localStorage

---

השנת ערך	הבאית ערך
<ul style="list-style-type: none"><li>• <code>localStorage.setItem("KEY", VALUE);</code></li><li>• <code>localStorage["KEY"] = VALUE;</code></li><li>• <code>localStorage.KEY = VALUE;</code></li></ul>	<ul style="list-style-type: none"><li>• <code>localStorage.getItem("KEY");</code></li><li>• <code>localStorage["KEY"]</code></li><li>• <code>localStorage.KEY</code></li></ul>

# localStorage

---

מחיקת ערכים	בדיקה אם המפתח קיים
<ul style="list-style-type: none"><li>• <code>localStorage.removeItem("KEY");</code></li><li>• <code>localStorage.clear();</code></li></ul>	<ul style="list-style-type: none"><li>• <code>if (localStorage["name"] != undefined)</code> <code>name = localStorage["name"] ;</code></li></ul>

# שמירת אובייקט ב localStorage

---

ב localStorage ניתן לשמר רק מחרוזות  
על מנת לשמר אובייקט נהפוך אותו למחרוזת json  
על ידי שימוש ב JSON.stringify(myObject)

```
const myObject = {  
    name: "john doe",  
    age: 32,  
    gender: "male",  
    profession: "optician"  
}
```

```
localStorage.setItem("myObject", JSON.stringify(myObject));
```

# החזירת אובייקט מה localStorage

---

```
let newObject = localStorage.getItem("myObject");  
  
console.log(JSON.parse(newObject));
```

# תרגיל ביתה pay

---

1. עבדו על ה Skelton המצורף
2. צרו כפטור עבור כל צעוז
3. לחיצה על הצעוז תוסיף אותו לסל
4. לחיצה על הסל תעביר לדף pay אשר יציג את רשימת הפריטים שנבחרו
5. כל טעינת דף יש לרוקן את העגלה ב pay ולהתחליל למלא מחדש.

Click the buttons to add to the cart

[robot](#)

[doll](#)

[ball](#)

[monopoly](#)



Click on the cart image to proceed to payment

[clear cart](#)

Click the buttons to add to the cart

[robot](#)

[doll](#)

[ball](#)

[monopoly](#)



Click on the cart image to proceed to payment

[clear cart](#)

## Your cart:

name: robot price: 100

name: doll price: 70

name: ball price: 30

name: monopoly price: 65

**Total Price = 265**

Header/footer across  
multiple pages

# Header/footer across multiple pages

---

- צור קובץ html נפרד המכיל את <header>
- בצע include לקובץ בקובץ html הנוכחי

הסבר:

1. header.html יכול包括 את כל הקוד של ה header.
2. על ידי fetchAPI - פונקציה javascript יש לבדוק לאיזה div בדף הנדרש להכניס את ה header.

# דוגמאות קיד

---

ראה דוגמא ב  
moodle.css  
header.html  
index.html

```
<!-- header.html -->
<header>
  <h1>My Website</h1>
  <nav>
    <ul>
      <li><a href="index.html">Home</a></li>
      <li><a href="about.html">About</a></li>
      <li><a href="galery.html">Gallery</a></li>
      <li><a href="client.html">Our Clients</a></li>
      <li><a href="career.html">Career</a></li>
      <li><a href="contact.html">Contact Us</a></li>
    </ul>
  </nav>
</header>
```

```
<body>
  <div id="header-placeholder"></div>
  <script>
    // JavaScript to load the header
    document.addEventListener('DOMContentLoaded', function() {
      fetch('header.html')
        .then(response => response.text())
        .then(data => {
          document.getElementById('header-placeholder').innerHTML = data;
        })
        .catch(error => console.error('Error loading header:', error));
    });
  </script>
  <main>
    <h2>Welcome to My Website</h2>
    <p>This is the home page content.</p>
  </main>

</body>
```

# העברת מידע באמצעות Query string

---

העברת מידע דרך ה-URL באמצעות הפרדת כתובות הדף ע"י ?

הנתונים עוברים ע"י צמד של מפתח-ערך, כאשר כל צמד מופרד באמצעות &

למשל, העברה של קניית מוצר:

```
<a href="queryStringProduct.html?product=milk">buy milk</a>
```

ההעברה של קניית מוצר + כמות יחידות:

```
<a href="queryStringProduct.html?product=milk&amount=3">buy 3 units of milk</a>
```

# הברת מידע באמצעות Query string

---

תפישת המידע בדף מקבל ע"י location.search , נותן לנו את החלק מס'1ן השאלה ועד סוף המחרוזת.

location.search → ?product=milk&amount=3

נעשה שימוש באובייקט **URLSearchParams** של שפת js על מנת להחזיר את הערך שאנו רוצים:

\*פונקציית get של האובייקטמחזירה את הערך של המפתח

```
let url = new URLSearchParams(location.search.substring(1));
url.get("product");
url.get("amount");
```

```
<body>
    <a href="queryStringProduct.html?product=milk&amount=3">buy 3 units of milk</a>
</body>
```

```
<html>                                         queryStringProduct.html  
  <head>  
    <meta charset="utf-8" />  
    <title></title>  
  </head>  
  <body>  
    <script>  
      console.log("keys:" + location.search); //?product = milk & amount=3  
      let url = new URLSearchParams(location.search.substring(1));  
      let spn = document.createElement("span");  
      document.body.appendChild(spn);  
      spn.innerHTML += "Number of " + url.get("product") + " : ";  
      spn.innerHTML += url.get("amount");  
    </script>  
  </body>  
</html>
```

# תרגיל ביתה – חלק א

---

1. הורד מה moodle את קובץ השלד
2. עליינו ליצור דף לוגין עבור מערכת
3. כניסה למערכת דורשת אימייל וסיסמה
4. המערכת בודקת את האימייל והסיסמה אל מול רשימת היוזרים (רשימת היוזרים נשמרת ב `localStorage`)
5. אם היוזר קיים – הוא מועבר לדף הבא
6. אם הוא אינו קיים הוא מקבל הודעה מתאימה

# חלק ב

**Band Information**

--select band--

—select band—

Metallica

U2

Abba

Imagine Dragons

## חלק א

הוסף לדף הבא מסעיף א רשימת בחירה של להקות JSON.

בשלד נתון אובייקט JSON המתאים להקות.

עם עליית הדף עליכם מלאו את **שמות הלהקות** שקיימות בJSON האלמנט select.

יש לрендר את האלמנטים דינמית ולהכניסם לתוך ה SELECT כמו בדוגמה.

**Band Information**

U2

**U2**

U2 are an Irish rock band from Dublin, formed in 1976. The group consists of Bono (lead vocals and rhythm guitar), the Edge (lead guitar, keyboards, and backing vocals), Adam Clayton (bass guitar), and Larry Mullen Jr. (drums and percussion).

## חלק ב

בחירה בלהקה תציג את שם הלהקה ואת התיאור שלה מה JSON

---

שאלות?