

STANDARDI U PRIMJENI INTERNETSKE TEHNOLOGIJE

Predavanje 10

Array

- Koriste se za pohranu kolekcija vrijednosti

- **Deklaracija i Inicijalizacija**

```
const voce = new Array("jabuka", "banana", "kruška");
```

```
const voce = ["jabuka", "banana", "kruška"];
```

- **Pristup Elementima**

- Elementima unutar polja može se pristupiti koristeći njihove indekse

```
const banana = voce[1];
```

- **Promjena Veličine**

- JavaScript polja su dinamička, što znači da se njihova veličina može mijenjati dodavanjem ili uklanjanjem elemenata.
- Metode poput **push()**, **pop()**, **shift()**, i **unshift()** se često koriste za manipulaciju elementima polja.

- **Metode za Iteraciju**

- Postoje razne metode za iteraciju kroz elemente polja, uključujući **forEach()**, **map()**, **filter()**, **reduce()**, i druge.

Array

- **Spread Operator**

- omogućava jednostavno kopiranje svih elemenata jednog polja u drugo.

```
const novoVoce = [...voce];
```

- **Destructuring**

- sintaktički način za ekstrakciju elemenata iz polja direktno u varijable

```
const [prvo, drugo, trece] = novoVoce;
```

- **Metode za Pretragu**

- **indexOf()**, **find()**, **findIndex()**, i **includes()** su neke od metoda koje se koriste za pretragu elemenata unutar polja.

- **Sortiranje**

- **sort()** i **reverse()** metode se koriste za sortiranje i obrtanje redoslijeda elemenata unutar polja

String

- **Ugrađene metode**

- **toLowerCase(), toUpperCase(), trim(), split(), startsWith(), endsWith(), includes()**, i druge.
- Ove metode olakšavaju obradu i analizu stringova.

- **Pretraga i zamjena**

- **indexOf(), lastIndexOf(), match(), search()** su neke od metoda za pretragu unutar stringa.
- **replace()** se koristi za zamjenu dijelova stringa.

- **Duljina stringa**

- **length** svojstvo daje duljinu stringa.
- Primjer: „Hello World”.**length** vraća broj karaktera u stringu.

Date

- Pruža širok raspon funkcionalnosti potrebnih za različite primjene, od jednostavnog prikazivanja datuma do složenih izračuna i manipulacija s datumima
- **Kreiranje**
 - Date objekti se mogu stvoriti koristeći `new Date()` konstruktor.
 - Postoji nekoliko načina za inicijalizaciju datuma, uključujući trenutno vrijeme, milisekunde (od 1. siječnja 1970.), ili specifične datum i vrijeme.
- **Metode za Dobivanje Datuma/Vremena**
 - **`getFullYear()`, `getMonth()`, `getDate()`, `getHours()`, `getMinutes()`, `getSeconds()`**, itd.
- **Metode za Postavljanje Datuma/Vremena**
 - **`setFullYear()`, `setMonth()`, `setDate()`, `setHours()`**, itd.
- **Vremenske Zone**
 - JavaScript obrađuje datum i vrijeme prema lokalnoj vremenskoj zoni korisnika.
 - Metode kao što su **`getUTCFullYear()`, `getUTCHours()`**, itd., vraćaju vrijednost u **UTC** (Coordinated Universal Time).

Date

- **Formatiranje Datuma**

- Date objekt se može pretvoriti u string koristeći metode poput **toString()**, **toLocaleDateString()**, i **toLocaleTimeString()** koje pružaju različite formate

- **Vremenska Razlika**

- Razlika između datuma se može izračunati u milisekundama i pretvoriti u željene jedinice (sekunde, minute, sate, dani).

- **ISO Formati Datuma**

- JavaScript podržava ISO 8601 format datuma, koji je koristan za prijenos datuma u string formatu, posebno u web aplikacijama

```
const date = new Date();  
const isoFormat = date.toISOString();  
//2023-05-04T13:49:48.077Z
```

- **Validacija Datuma**

- Provjeravanje ispravnosti datuma se može obaviti koristeći **isNaN()** funkciju s **Date** objektom

Window

- **Globalni objekt**

- Window objekt je globalni objekt u kontekstu web preglednika. To znači da sve globalne varijable i funkcije postaju svojstva Window objekta.

- **DOM Manipulacija**

- Window objekt sadrži **document** svojstvo koje se koristi za pristup i manipulaciju DOM-om stranice

- **Lokacija i povijest navigacije**

- Svojstva kao što su **window.location** i **window.history** omogućuju manipulaciju URL-om trenutne stranice i navigacijskom poviješću

- **Timers**

- Window objekt nudi funkcije poput **setTimeout()** i **setInterval()** za izvršavanje koda nakon određenog vremenskog intervala ili ponavljajuće u određenim intervalima.

- **Alerts, Prompts, i Confirms**

- Funkcije kao što su **alert()**, **prompt()**, i **confirm()** su dio Window objekta i koriste se za interakciju s korisnikom.

- **Sesija i Lokalna Pohrana**

- **sessionStorage** i **localStorage** omogućuju pohranu podataka u pregledniku korisnika

document

- **Selektiranje elementa**

- Metode poput **getElementById()**, **getElementsByClassName()**, **getElementsByTagName()**, i **querySelector()** omogućuju selektiranje elemenata na stranici.

- **Stvaranje novih elementa**

- Metode kao što su **createElement()** i **createTextNode()** se koriste za stvaranje novih HTML elemenata i tekstualnih čvorova.

- **Dodavanje ili uklanjanje elementa**

- **appendChild()** i **removeChild()** su metode za dodavanje ili uklanjanje elemenata u DOM-u

- **Rukovanje Događajima**

- **addEventListener()** i **removeEventListener()** omogućuju vezivanje funkcija na događaje za određene elemente.

- **Atributi i stilovi**

- Možete mijenjati attribute i stilove elemenata koristeći **setAttribute()**, **getAttribute()**, **removeAttribute()**, kao i direktno pristupanje CSS stilovima kroz **style** svojstvo..

- **Rad s klasama**

- **classList** svojstvo nudi metode poput **add()**, **remove()**, **toggle()**, i **contains()** za upravljanje CSS klasama elemenata.

Browser APIs

- Browser API su skupovi funkcija koje omogućuju programerima web aplikacija da koriste funkcionalnosti preglednika i operativnog sistema. Evo nekoliko primjera i kategorija Browser API-ja

- **DOM API**

- Omogućava selekciju, dodavanje, uklanjanje i izmjenu HTML elemenata.

- **Fetch API**

- Omogućuje asinkrono dohvaćanje resursa preko mreže. Zamjenjuje stariji **XMLHttpRequest** API.

- **Canvas API**

- Koriste se za crtanje i renderiranje 2D odnosno 3D grafika

- **Storage API**

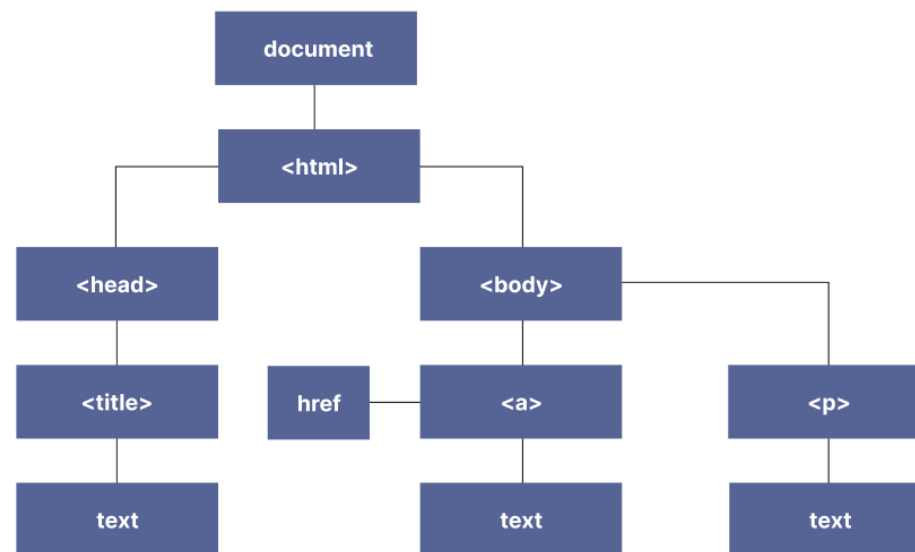
- Uključuje **localStorage** i **sessionStorage** za pohranu podataka na klijentskoj strani, omogućujući spremanje ključ-vrijednost parova.

- **Sensors API**

- Omogućuje pristup raznim senzorima uređaja, kao što su akcelerometar, žiroskop, itd.

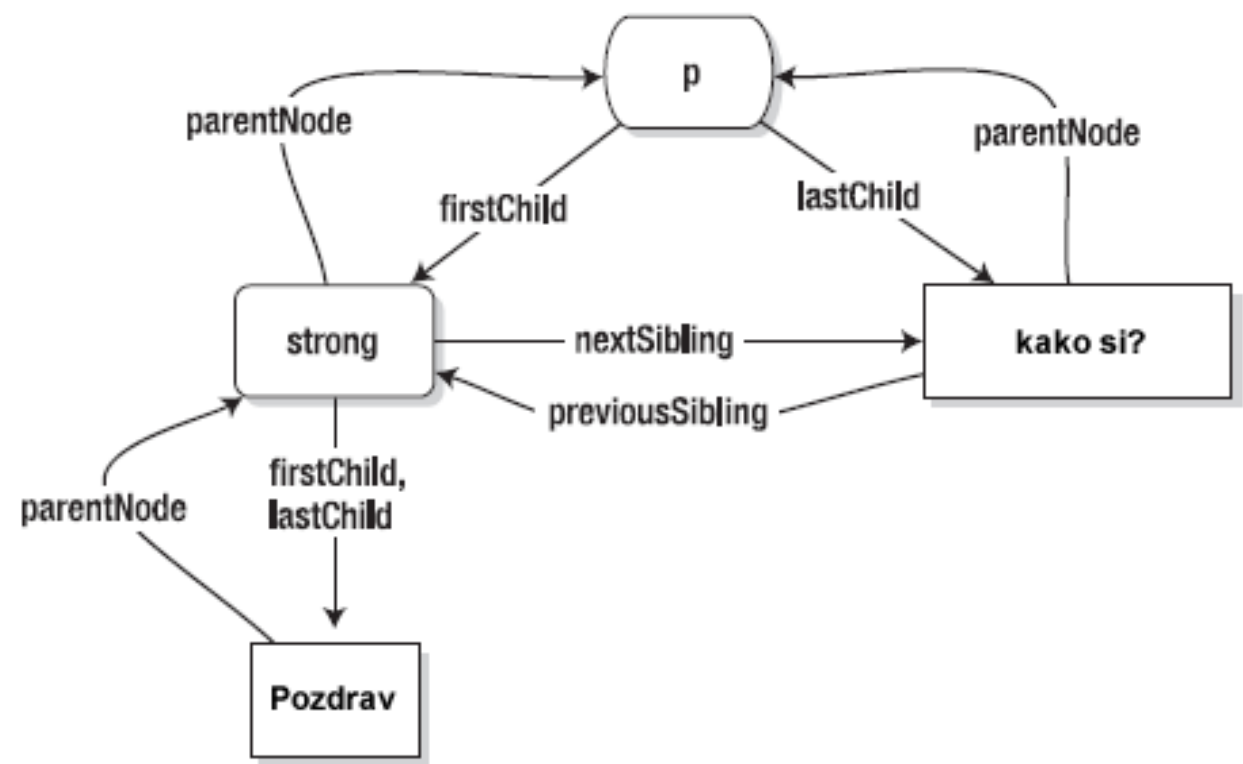
DOM - Document Object Model

- DOM je programsko sučelje koje omogućava skriptama da pristupe i manipuliraju HTML dokumentima.
- U kontekstu web preglednika, DOM predstavlja strukturu web stranice kao hijerarhijsko stablo objekata, omogućujući programerima da mijenjaju sadržaj, strukturu i stil web stranice dinamički.



Međusobni odnosi elemenata

```
<p>  
  <strong>Pozdrav</strong> kako si?  
</p>
```



Moguća
navigacija
čvora

