

STANDARDI U PRIMJENI INTERNETSKE TEHNOLOGIJE

Predavanje 09

JavaScript

- JavaScript je programski jezik koji se najčešće koristi za stvaranje interaktivnih efekata unutar web preglednika. Kao jedan od tri temeljna jezika web razvoja, uz HTML i CSS, JavaScript omogućava programerima da dodaju dinamičke značajke na web stranice.

Što je JavaScript

- JavaScript je dinamički, interpretirani programski jezik koji omogućava interaktivnost na web stranicama.
- Koristi se za dodavanje dinamičkih efekata i funkcionalnosti koje HTML i CSS ne mogu sami postići.

Osnovne značajke

- **Dinamički:** Sposoban je mijenjati sadržaj na stranici u stvarnom vremenu.
- **Slabog tipiziranja:** Varijable mogu držati vrijednosti bilo kojeg tipa bez izričite deklaracije tipa.
- **Objektno-orijentiran:** Podržava paradigmu objektno-orijentiranog programiranja (OOP).

JavaScript

Primjena JavaScripta

- **Manipulacija DOM-om** (Document Object Model) za dinamičku promjenu sadržaja, strukture i stila stranica.
- **Validacija formi** prije slanja na poslužitelj.
- **Kreiranje animacija**, igara, i interaktivnih korisničkih sučelja.
- Razvoj aplikacija za **klijentsku** i **poslužiteljsku** stranu (Node.js).

JavaScript okviri i biblioteke

- Popularni okviri kao što su **React**, **Angular**, i **Vue.js** omogućavaju razvoj dinamičnih i responzivnih single-page aplikacija (**SPA**).
- Biblioteke poput **jQuery**-a olakšavaju manipulaciju DOM-om i AJAX zahtjeve.

JavaScript je ključan za razvoj modernih web aplikacija.

Razumijevanje JavaScripta otvara vrata naprednom web razvoju, uključujući front-end i back-end tehnologije.

JavaScript - funkcije

JavaScript funkcije su temeljni blokovi koda koji omogućuju grupiranje više naredbi za izvođenje određenog zadatka.

Hoisting

- JavaScript mehanizam gdje se deklaracije funkcija premještaju na vrh njihovog obuhvata (*scope*) prije izvršavanja koda, što omogućava pozivanje funkcija prije njihove deklaracije u kodu.
- Funkcijski izrazi ne ulaze u hoisting mehanizam

```
function imeFunkcije() {  
    // Blok koda za izvršavanje  
}
```

```
function pozdrav(ime) {  
    console.log("Pozdrav, " + ime);  
}
```

```
function zbroj(a, b) {  
    return a + b;  
}
```

```
const kvadrat = function(x)  
    return x * x;  
};
```

JavaScript - događaji

Što su JavaScript događaji

- Događaji su akcije ili događanja koja se mogu detektirati unutar DOM-a (Document Object Model), što omogućava web aplikacijama da reagiraju na njih

Primjeri događaja

- **Miš:** click, dblclick, mouseover, mouseout, mousemove
- **Tipkovnica:** keypress, keydown, keyup
- **Forma:** submit, change, focus, blur
- **Prozor:** load, resize, scroll, unload

Dodavanje slušača

- JavaScript omogućava dodavanje slušača događaja na elemente koristeći **addEventListener** metodu.

Uklanjanje slušača

- Slušači događaja mogu se ukloniti koristeći **removeEventListener** metodu.

Propagacija događaja

- Događaji se propagiraju kroz DOM u dva smjera: **bubbling** (od najdubljeg do najvišeg elementa) i **capturing** (od najvišeg do najdubljeg).
- Propagaciju je moguće zaustaviti koristeći **event.stopPropagation()**.

JavaScript - događaji

Prekid predefiniranog ponašanja

- `event.preventDefault()` sprečava predefinirano ponašanje preglednika (npr., sprečava slanje forme pri kliku na submit gumb).

Delegacija događaja

- Tehnika za dodavanje slušača događaja na roditeljski element umjesto na pojedinačne dječje elemente, koristeći činjenicu da se događaji propagiraju. Ovo je posebno korisno kada imate mnogo elemenata koji trebaju osluškivati na isti događaj, ili kada elementi mogu biti dinamički dodani ili uklonjeni iz DOM-a.

```
<ul id="todo">
  <li>Zadatak 1</li>
  <li>Zadatak 2</li>
  <li>Zadatak 3</li>
  <!-- Novi itemi mogu biti dinamički dodani -->
</ul>
```

```
<script>
  document.getElementById('todo')
    .addEventListener('click', function (e) {
      if(e.target.tagName === 'LI'){
        console.log(e.target.textContent)
      }
    })
</script>
```