

JavaScript: Objekti i Korištenje gotovih skripti

Sadržaj:

1. Uvod
2. Povijest JavaScripta
3. Osnovne značajke JavaScripta
4. Što su objekti u JavaScriptu
5. Tipovi objekata
6. Kreiranje i korištenje objekata
7. Nasljeđivanje objekata (prototype)
8. Korištenje gotovih skripti
9. Prednosti i mane gotovih skripti
10. Primjeri koda (objekti + skripte)
11. Zaključak
12. Literatura

1. Uvod

Razvojem interneta i web aplikacija pojavila se potreba za jezicima koji omogućuju interaktivnost i dinamičnost web stranica. JavaScript je postao standard za razvoj klijentskih aplikacija koje se izvršavaju u preglednicima. Ovaj seminar bavi se dvama ključnim područjima JavaScripta: objektima i korištenjem gotovih skripti.

2. Povijest JavaScripta

JavaScript je nastao 1995. godine kada ga je Brendan Eich razvio za Netscape Navigator preglednik. Prvotno nazvan LiveScript, ubrzo je preimenovan u JavaScript kako bi iskoristio tadašnju popularnost Jave, iako ova dva jezika nisu direktno povezana. Danas je JavaScript neizostavan dio gotovo svake web stranice.

3. Osnovne značajke JavaScripta

JavaScript je:

- Dinamički tipiziran
- Višestruko paradigmatiski (podržava objektno orijentirano, funkcionalno i imperativno programiranje)
- Izvršava se na strani klijenta (u pregledniku)
- Podržava događaje, manipulaciju DOM-om i AJAX komunikaciju

Njegova svestranost čini ga neizostavnim alatom modernog weba.

4. Što su objekti u JavaScriptu

Objekti su temeljna struktura za pohranu podataka i funkcionalnosti. Objekt u JavaScriptu je zbirka svojstava, pri čemu svako svojstvo ima ključ (naziv) i vrijednost.

Primjer:

```
let knjiga = {  
  naslov: "JavaScript osnove",  
  autor: "Ivan Ivić",  
  godina: 2022,  
  ispisInfo: function() {  
    console.log(this.naslov + ", " + this.autor);  
  }  
};
```

Objekt knjiga ima svojstva naslov, autor i godina, te metodu ispisInfo.

5. Tipovi objekata

U JavaScriptu postoje dvije glavne vrste objekata:

- **Standardni objekti** (npr. Object, Array, Date, Math)
- **Korisnički definirani objekti** (kreirani od strane programera)

Objekti omogućuju grupiranje podataka i funkcija u logičke cjeline.

6. Kreiranje i korištenje objekata

Objekte možemo kreirati na više načina:

- Literali objekta:

javascript

KopirajUredi

```
let osoba = { ime: "Ana", godine: 28 };
```

- Funkcija konstruktor:

javascript

KopirajUredi

```
function Osoba(ime, godine) {  
    this.ime = ime;  
    this.godine = godine;  
}
```

```
let osoba1 = new Osoba("Marko", 30);
```

- Object.create() metoda:

javascript

KopirajUredi

```
let noviObjekt = Object.create(osoba);
```

Metode omogućuju izvođenje operacija nad objektima.

7. Nasljeđivanje objekata (prototype)

JavaScript koristi **prototipno nasljeđivanje**. Svaki objekt ima svoj prototip, odnosno drugi objekt od kojeg može nasljeđivati svojstva i metode.

Primjer:

javascript

KopirajUredi

```
function Student(ime, godina) {  
    this.ime = ime;
```

```
this.godina = godina;  
}  
Student.prototype.predstaviSe = function() {  
  console.log("Ja sam " + this.ime);  
};
```

```
let student1 = new Student("Luka", 1);  
student1.predstaviSe(); // Ispisuje: Ja sam Luka
```

Prototipovi omogućuju stvaranje hijerarhije i dijeljenje funkcionalnosti između objekata.

8. Korištenje gotovih skripti

Gotove skripte (biblioteke) omogućuju brži razvoj aplikacija bez potrebe za pisanjem sveg koda iznova. Popularne biblioteke uključuju:

- **jQuery** – pojednostavljuje rad s DOM-om i događajima
- **Moment.js** – rad s datumima i vremenima
- **Chart.js** – izrada grafova
- **Lodash** – rad s nizovima i objektima

Uključivanje skripte:

html

KopirajUredi

```
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
```

Korištenjem gotovih rješenja štedi se vrijeme i povećava kvaliteta softvera.

9. Prednosti i mane gotovih skripti

Prednosti:

- Brža izrada aplikacija
- Manje programerskih pogrešaka
- Standardizirana rješenja

Mane:

- Ovisnost o trećim stranama
- Veličina datoteka može usporiti učitavanje stranice
- Ponekad nepotrebno korištenje velikih biblioteka za male funkcionalnosti

10. Primjeri koda

Primjer objekta:

javascript

KopirajUredi

```
let racunalo = {  
  procesor: "Intel i7",  
  memorija: "16GB",  
  ukljuci: function() {  
    console.log("Računalo se uključuje...");  
  }  
};  
  
racunalo.ukljuci();
```

Primjer gotove skripte (jQuery):

html

KopirajUredi

```
<!DOCTYPE html>
<html lang="hr">
<head>
  <meta charset="UTF-8">
  <title>Primjer jQuery</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
  <button id="gumb">Klikni me!</button>
  <script>
    $("#gumb").click(function() {
      alert("Pozdrav iz jQuery-ja!");
    });
  </script>
</body>
</html>
```

11. Zaključak

JavaScript objekti predstavljaju osnovni alat za upravljanje podacima i funkcionalnostima unutar aplikacija. Gotove skripte omogućuju bržu izradu, veću pouzdanost i smanjeno vrijeme razvoja. Kombiniranjem znanja o objektima i korištenjem provjerenih alata možemo izraditi stabilne, moderne i učinkovite aplikacije.

12. Literatura

- Mozilla Developer Network (MDN) – JavaScript dokumentacija
- Eloquent JavaScript, Marijn Haverbeke
- W3Schools – JavaScript Tutorials
- JavaScript.info