



Uvod u JavaScript

Dragi polaznici, ova prezentacija nije primarno namijenjena za učenje već služi kao **pomoćni materijal**. Kao takva, ne može zamijeniti predavanja, literaturu kao niti vašu (pro)aktivnost na nastavi i konzultiranje s predavačem u slučaju potencijalnih pitanja i/ili nejasnoća.

JavaScript

20 sati predavanja, 40 sati vježbi

Parcijalni ispit

Završni ispit

Cjeline

Uvod u JavaScript

Osnove programiranja

DOM manipulacija

Napredni JavaScript

Upravljanje pogreškama

HTML5 razvojni koncepti

Uvod u JavaScript

Povijest JavaScripta

JavaScript (JS)

Uz **HTML** i **CSS**, jedna od temeljnih tehnologija web-a.

Kreiran 1995. od strane kompanije Netscape kao jezik za browsere koji opisuje ponašanje web stranica.

Koristi se i naziv ECMAScript, koji predstavlja standard jezika.

- zadnji objavljen standard jezika je ES2019



Klasifikacija jezika

JavaScript != Java

JavaScript je skriptni programski jezik visoke razine, dobar i za objektno-orijentirano i funkcionalno programiranje.

Skup funkcija koje pruža ovisi o okolini u kojoj se koristi.

Verzije JavaScript jezika

Od početka jezika u 90-ima JavaScript redovito dobiva nove verzije jezika sa dodanim ili oduzetim dijelovima temeljnih funkcionalnosti. Time se jezik razvija kroz vrijeme, ali i od nas zahtijeva da ga nanovo učimo i nadogradimo svoje znanje.

Najvažnije verzije koje nas zanimaju su:

...

ES5

ES6

...

Vrijednost učenja JavaScripta

Široko raširen jezik, ne koristi se više samo u browser okolini.

<https://insights.stackoverflow.com/survey/2019>



Most Popular Technologies

Programming, Scripting, and Markup Languages

All Respondents

Professional Developers



Primjeri JavaScripta na web-u

<https://www.facebook.com>

<https://www.google.com/maps>

Primjeri:

https://www.awwwards.com/inspiration/search?text=javascript&type=external_element

<https://www.creativebloq.com/web-design/examples-of-javascript-1233964>

Što ako isključimo JavaScript?

<https://www.smashingmagazine.com/2018/05/using-the-web-with-javascript-turned-off/>

Važno je razvijati web stranice imajući na umu osnove web developmenta, s JavaScriptom kao nadogradnjom.



Uvod u JavaScript

Razvojni alati

Za one koji žele znati više

[https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Installing_basic_software)

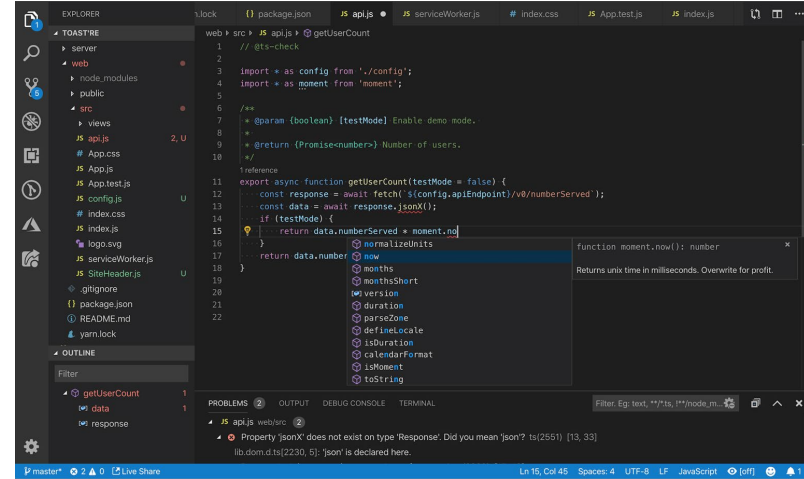
[US/docs/Learn/Getting_started_with_the_web/Installing_basic_software](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Installing_basic_software)

<https://code.visualstudio.com/docs/nodejs/working-with-javascript>

VS Code

Visual Studio Code uključuje ugrađeni JavaScript IntelliSense, uklanjanje pogrešaka, oblikovanje, navigaciju kroz kod i mnoge druge napredne jezične značajke.

IntelliSense vam pomaže sa inteligentnim dovršavanjem koda i informacije na hover kako biste brže i ispravnije mogli pisati kôd.



Chrome DevTools

Chrome DevTools skup je alata za web programere ugrađenih izravno u preglednik Google Chrome i najčešće je korišten web developer alat.

Tab	Uloga
Elements	za HTML i CSS
Console	za debuggiranje grešaka i JS koda
Sources	Pregled svih izvora stranice i za postavljanje breakpointova
Performance	Praćenje redosljeda izvođenja i utjecaja na brzinu stranice
Network	Prikaz redosljeda loadanja resursa i detalja requesta

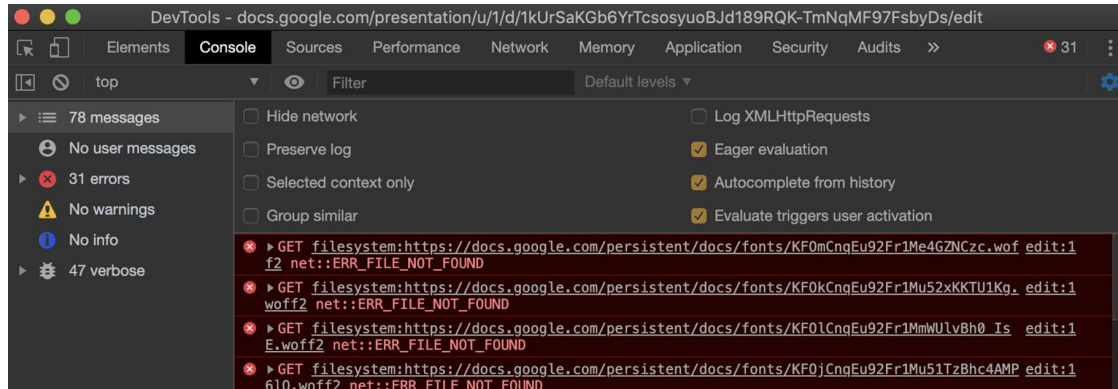
Chrome DevTools

Console

Vjerojatno najvažniji tab za JS developera.

Kroz konzolu pratimo errore koji se javljaju i koji nam ruši aplikaciju.

console.log i **debugger**; naredbe nam omogućuju praćenje koda i njegov tijek.



Firefox

Mozilla Foundation je sudjelovao u razvijanju JavaScript jezika pa Firefox web preglednik ima odličnu podršku za testiranje i razvijanje JavaScripta.

- isti alati kao Google Chrome
- plus WebIDE i **Scratchpad**

Njihova stranica je referentna za JavaScript standarde:

<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

Npm i node paketi

Node Package Manager

Otkad je JavaScript razvijen i za poslužiteljsku stranu, postalo je jako popularno razvijati i downloadati pakete razvijene u NodeJS-u.

www.npmjs.com ima na tisuće besplatnih paketa za preuzimanje i korištenje.

Gulp	Automatiziracija zadataka kao što su minifikacija
JSHint	Analiza koda i ispravljanje grešaka
lodash	Za rad sa naprednim tipovima podataka
Babel	Za rad sa različitim verzijama JS-a

Ostali alati

Postman	Za razvijanje i testiranje backend API-ja
Sublime text	Alternativni text editor
CircleCI	Za deploy koda
Terminal	Tekstualni pristup operativnom sustavu
Github/Bitbucket	Za kontrolu verzija koda
Sourcetree	Vizualni prikaz git protokola

Uvod u JavaScript

JavaScript na web stranici

Klijentski JavaScript

Prije nego počnemo razgovarati o JavaScriptu, vrijedno je razmisliti o web stranicama i aplikacijama koje prikazujemo u web preglednicima.



Ugradnja JavaScripta u HTML

1. Inline, između `<script>` i `</script>`
2. **Iz vanjske datoteke definirane src atributom `<script>` tag-a**
3. U atributu alata za obradu HTML događaja, kao što su onclick ili onmouseover
4. U URL-u koji koristi poseban javascript: protokol.

```
<script>  
...  
</script>
```

```
<script src="scripts.js"></script>
```

Izvođenje JavaScript koda

Odvija se u 2 faze:

1. Učitava se sadržaj dokumenta i kôd iz `<script>` elemenata je pokrenut.
2. Druga faza se odvija asinkrono i kao reakcija na događaje koji se odvijaju unutar web preglednika (klik, loadanje dokumenta, itd.)

Loadanje JS na stranici

Zadano ponašanje koje određuje izvođenje JavaScript koda na web stranici u prvoj fazi je **sinkrono** i **blokirajuće**. Što znači da će dijelovi naše stranice čekati dok se ne učitaju i izvede sva logika iz naših JS skripti!

Kad HTML parser naiđe na `<script>` element, on mora loadati i pokrenuti skriptu prije nego što će moći nastaviti s analizom i prikazom dokumenta.

Sinkrone, asinkrone i odgođene skripte

Različiti tipovi učitavanja:

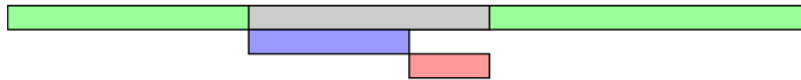
1. **Sinkrono** - uključenje JS skripte prije kraja `<body>` elementa - kad je 99% html-a već učitano
2. **Asinkrono** - stavljanjem **async** atributa na `<script>` element: loadanje skripte ne blokira dokument
3. **Odgođeno** - stavljanjem **defer** atributa na `<script>` element: ni loadanje ni izvođenje JS-a neće blokirati dokument

Legend

- HTML parsing
- HTML parsing paused
- Script download
- Script execution

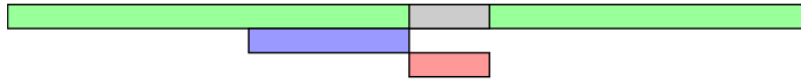
<script>

Let's start by defining what **<script>** without any attributes does. The HTML file will be parsed until the script file is hit, at that point parsing will stop and a request will be made to fetch the file (if it's external). The script will then be executed before parsing is resumed.



<script async>

async downloads the file during HTML parsing and will pause the HTML parser to execute it when it has finished downloading.



<script defer>

defer downloads the file during HTML parsing and will only execute it after the parser has completed. **defer** scripts are also guaranteed to execute in the order that they appear in the document.



JavaScript na web stranici

Pratite upute u Upute.txt

Vrijeme trajanja vježbe: 25min

Uvod u JavaScript

Biblioteke i frameworks

Biblioteka (Library)

Skup unaprijed napisanog JavaScript koda koji omogućava lakši razvoj JavaScript aplikacija.

Ima više kategorija biblioteka, koje se bave:

1. DOM manipulacijom
2. Grafikom na web-u
3. User Interface-om
4. Testiranjem koda
5. AJAX (komunikacija sa serverom)

jQuery

jQuery je JavaScript biblioteka dizajnirana za pojednostavljenje manipulacije HTML-om, kao i za rukovanje događajima, CSS animaciju i Ajax.

Jedna od povijesno najpopularnijih JS biblioteka:

jQuery koristi čak **97,5%** svih web stranica čiju JavaScript biblioteku poznajemo, tj. **74,2%** svih web stranica.

Uz temeljnu jQuery biblioteku postoje i mnoge biblioteke razvijene na temelju jQuery-ja: <https://www.creativebloq.com/jquery/top-jquery-plugins-6133175>

jQuery

```
document.addEventListener("ready", (function(){  
    document.getElementsByTagName("button")[0].addEventListener("click", (function(){  
        document.querySelector("p").hide();  
    }));  
}));
```

```
$(document).ready(function(){  
    $("button").click(function(){  
        $("p").hide();  
    });  
});
```

JS frameworks (okviri)

JavaScript framework je aplikacijska apstrakcija napisana u JavaScriptu.

Razlikuje se od JavaScript knjižnice u načinu na koji ga koristimo: biblioteka nudi funkcije koje bi mogle biti pozvane kad nam zatrebaju i pomažu nam u kodiranju, dok okvir definira cjelokupni dizajn i arhitekturu aplikacije.

Angular

Backbone

React (iako nije punokrvni framework)

Vue.js (sličan Reactu)

MVC arhitektura

Model, **V**iew, **C**ontroller

Model je odgovoran za održavanje podataka o aplikaciji i poslovne logike.

View je korisničko sučelje aplikacije koje prikazuje podatke.

Controller obrađuje zahtjeve korisnika i poziva odgovarajući View s podacima Modela.

U tom smislu, **React** je View dio frameworka, i pomaže nam u kreiranju interaktivnih korisničkih sučelja.