

## Tartalom

HTML.....	2
1. HTML Dokumentumvázlat .....	2
2. HTML Címkék (Tags).....	3
3. Attribútumok .....	3
4. Bekezdések, Fejlécek, Listák.....	3
Javascript .....	4
1. Változók és adattípusok.....	4
2. Függvények .....	4
3. Eseménykezelés.....	4
4. Ciklusok.....	5
JQUERY.....	6
1. Eseménykezelés.....	6
2. Stílusok és Animációk .....	6
Fejlesztőkörnyezetek .....	7

# HTML

HTML, vagyis a HyperText Markup Language, a weboldalak strukturális alapját képezi. Az HTML egy olyan kódolási nyelv, amely segítségével leírhatjuk egy weboldal szerkezetét és tartalmát.

## 1. HTML Dokumentumvázlat

Az HTML kód minden weboldalnak egy dokumentumvázlatot alkot. Az alapstruktúra a következőképpen néz ki:

```
<!DOCTYPE html>

<html>

<head>

  <title>Weboldal Címe</title>

</head>

<body>

  <!-- Itt található a weboldal tartalma -->

</body>

</html>
```

**<!DOCTYPE html>:** Ez egy speciális tag, amelyet azért helyeznek el az HTML dokumentum elején, hogy a böngésző pontosan tudja, milyen HTML verziót kell értelmeznie az adott oldalon.

**<html>:** A dokumentum gyökéreleme.

**<head>:** A dokumentum fejléce, a metaadatokat és címet tartalmazza.

**<title>:** A weboldal címe, ami a böngésző fülén jelenik meg.

**<body>:** A dokumentum tartalmát tartalmazza.

[https://www.w3schools.com/html/html5\\_svg.asp](https://www.w3schools.com/html/html5_svg.asp)

## 2. HTML Címkék (Tags)

Az HTML címkék arra szolgálnak, hogy meghatározzák, milyen típusú tartalom van közöttük. A legtöbb címke páros, vagyis nyitó és záró címkéből áll.

```
<p>Ez egy bekezdés.</p>  
  
<a href="https://www.pelda.hu">Ez egy hivatkozás</a>  
  
<h1>Ez egy címsor</h1>
```

## 3. Attribútumok

Az attribútumok további információkat adnak a címkéknek. Az attribútumok a címkék belsejében helyezkednek el, és kulcs-érték párokból állnak.

```
<a href="https://www.pelda.hu" target="_blank">Hivatkozás új ablakban</a>  
  

```

href: URL hivatkozás.

target="\_blank": Az URL új ablakban való megnyitásához (opcionális).

src: Az kép forrása.

alt: Képalternatíva, szöveg amit akkor látnak, ha a kép nem tölthető be.

## 4. Bekezdések, Fejlécek, Listák

Az HTML lehetővé teszi a tartalom strukturált elrendezését, például bekezdésekkel, fejlécekkel és listákkal.

```
<h2>Alcím</h2>  
  
<p>Ez egy bekezdés.</p>  
  
<ul>  
  <li>Elem 1</li>  
  <li>Elem 2</li>  
</ul>
```

# Javascript

JavaScript egy programozási nyelv, amelyet általában weboldalakon alkalmaznak a felhasználói interaktivitás és dinamikus tartalom létrehozására. JavaScript a böngészőkben fut, és lehetővé teszi az interaktív funkciók hozzáadását egy weboldalhoz. Képes kezelni eseményeket, manipulálni a DOM-ot (Dokumentum Objektum Modell), és kommunikálni a felhasználóval.

## 1. Változók és adattípusok

Változókban tároljuk az adatokat. A változók létrehozása a **var**, **let** vagy **const** kulcsszavakkal történik.

```
let szam = 10;  
  
const konstans = "Állandó érték";
```

## 2. Függvények

Függvények segítségével csoportosíthatjuk a kódot, és újrafelhasználható részeket hozhatunk létre.

```
function udvozlet(nev) {  
  
    console.log("Hello, " + nev + "!");  
  
}  
  
udvozlet("Világ");
```

## 3. Eseménykezelés

JavaScript segítségével reagálhatunk a felhasználói interakciókra, mint például a gombra kattintás vagy az űrlap beküldése.

```
document.getElementById("gomb").addEventListener("click", function() {  
  
    console.log("A gombra kattintottak!");  
  
});  
  
//Feltételes utasítások:
```

```
//Az if, else if és else kulcsszavakkal feltételes utasításokat készíthetünk.
```

```
let szam = 5;
```

```
if (szam > 0) {
```

```
    console.log("A szám pozitív");
```

```
} else if (szam < 0) {
```

```
    console.log("A szám negatív");
```

```
} else {
```

```
    console.log("A szám nulla");
```

```
}
```

## 4. Ciklusok

Használhatunk for és while ciklusokat ismétlődő feladatok végrehajtására.

```
for (let i = 0; i < 5; i++) {
```

```
    console.log("Szám: " + i);
```

```
}
```

```
let j = 0;
```

```
while (j < 3) {
```

```
    console.log("Még egy ismétlés");
```

```
    j++;
```

```
}
```

# JQUERY

A jQuery egy JavaScript könyvtár, amely megkönnyíti a DOM manipulációt, eseménykezelést, animációkat és aszinkron kéréseket a webfejlesztés során. A jQuery-t a HTML oldalba illeszthetjük be a `<script>` címkék segítségével. Többféle módja van a telepítésnek, például CDN (Content Delivery Network) használatával vagy a könyvtár letöltésével és helyi fájl referálásával.

[https://www.w3schools.com/jquery/jquery\\_dom\\_get.asp](https://www.w3schools.com/jquery/jquery_dom_get.asp)

```
<!-- CDN használata -->
```

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

DOM Elemek Kiválasztása:

A jQuery segítségével egyszerűen kiválaszthatjuk a DOM elemeket, például az azonosító vagy osztály alapján.

```
let elem = $("#azonosito");// Az id alapján kiválasztás
```

```
let elemek = $(".osztaly");// Az osztály alapján kiválasztás
```

## 1. Eseménykezelés

A jQuery lehetővé teszi egyszerű eseménykezelést. Például egy gombra kattintás eseményére reagálhatunk:

```
$("#gomb").click(function() {  
    console.log("A gombra kattintottak!");  
});
```

## 2. Stílusok és Animációk

Stílusokat változtathatunk és animációkat hozhatunk létre a jQuery segítségével.

```
$("#elem").animate({ color: "red" }, 1000);  
  
$("#elem").css("font-size", "20px");// Stílus változtatás
```

## Fejlesztőkörnyezetek

- Visual Studio Code (VSCode): A VSCode ingyenes kódszerkesztő, amely támogatja a jQuery-t is. Integrált terminállal, pluginokkal és kiterjedt kódkiegészítéssel rendelkezik.
- Sublime Text: Egy gyors kódszerkesztő, amelyet sok fejlesztő használ JavaScript és jQuery fejlesztéshez.
- Atom: Ingyenes, nyílt forráskódú kódszerkesztő, amely a jQuery fejlesztéséhez is alkalmas. Könnyen testreszabható és számos hasznos funkciót tartalmaz.
- WebStorm: A JetBrains által készített JavaScript, HTML és CSS fejlesztéshez tervezett teljes körű fejlesztőkörnyezet. A jQuery is könnyen integrálható a WebStormban.
- Notepad++: Könnyű a kezelhetősége és gyorsasága miatt népszerű.