12. DOM bilan ishlash

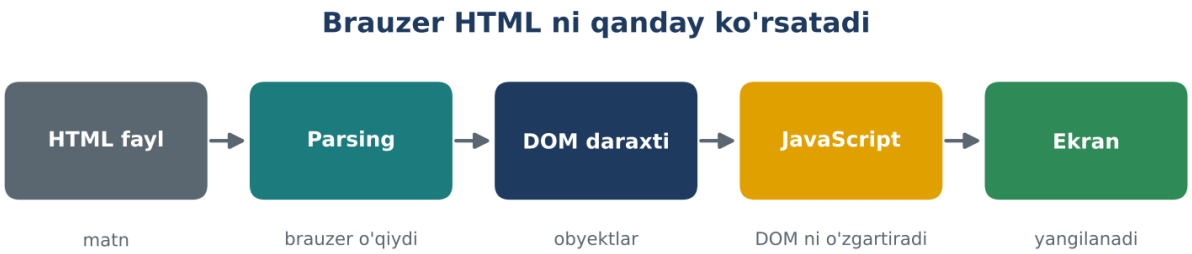


DOM — brauzer HTML sahifani JavaScript ga "ko'rsatadigan" shakl. Uning yordamida sahifani **o'qish va o'zgartirish** mumkin. Bu dars DOM ni noldan tushuntiradi. Kodlar sinovdan o'tgan; ular aynan brauzerda ham xuddi shunday ishlaydi.

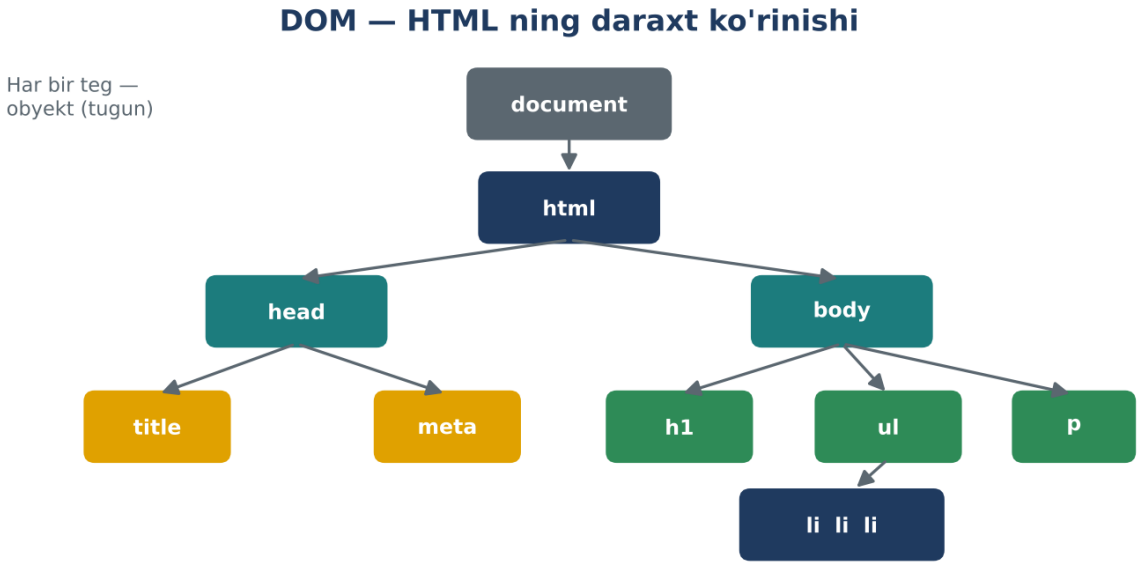
> **Bu darsda nimani o'rganasiz**
>
> DOM nima va `document` obyekti.
>
> Elementni tanlash: `getElementById`, `querySelector`.
>
> Matn va HTML ni o'zgartirish.
>
> Atribut va uslub bilan ishlash, `classList`.
>
> Yangi element yaratish va o'chirish.
>
> Sahifaga JavaScript ulash.

1.1. DOM nima?

Siz HTML yozasiz: `<h1>`, `<p>`, `<ul>`. Brauzer bu matnni o'qib, xotirada **obyektlar daraxti**ni yaratadi. Bu daraxt **DOM (Document Object Model)** deyiladi. Har bir teg — shu daraxtdagi bitta obyekt. JavaScript bu obyektlar orqali sahifani boshqaradi.



1-rasm. HTML dan ekrangacha



2-rasm. DOM daraxti

DOM ning kirish eshigi — `document` obyekti. Barcha ishlar `document` dan boshlanadi.

Bu qo'llanmadagi misollar

Quyida ko'rsatiladigan natijalar quyidagi namunaviy HTML ustida olingan. Siz ham o'z faylingizga shu HTML ni yozib, kodlarni brauzerda sinab ko'rishingiz mumkin.

namunaviy HTML (body ichida)

```
<h1 id="sarlavha" class="katta">Salom</h1>
<p class="matn">Birinchi abzas</p>
<p class="matn qizil">Ikkinchi abzas</p>
<ul id="royxat">
  <li>Olma</li><li class="tanlangan">Nok</li><li>Uzum</li>
</ul>
<input id="ism" type="text" value="Ali">
<button id="tugma">Bos</button>
```

1.2. Elementni tanlash



3-rasm. Elementni tanlash usullari

1-jadval. Tanlash usullari

Usul	Nima qaytaradi	Misol
`document.getElementById("id")`	Bitta element yoki `null`	`getElementById("sarlavha")`
`document.querySelector("sel")`	**Birinchi** mos element yoki `null`	`querySelector(".matn")`
`document.querySelectorAll("sel")`	**Barcha** mos elementlar (`NodeList`)	`querySelectorAll("li")`

`querySelector` ichidagi matn — **CSS selektor**: `#id`, `.klass`, `teg`, `ul li`, `p.matn` kabi.

```
const sarlavha = document.getElementById("sarlavha");
const abzas = document.querySelector(".matn");        // birinchi .matn
const hammasi = document.querySelectorAll(".matn");   // ikkalasi
const li = document.querySelectorAll("#royxat li");

console.log(sarlavha.textContent);
console.log(abzas.textContent);
console.log(hammasi.length, li.length);
console.log(document.querySelector(".yoq"));          // topilmasa: null
```

Natija:

```
Salom
Birinchi abzas
2 3
null
```

null ga e'tibor

Element topilmasa `null` qaytadi. `null.textContent` xato beradi (`TypeError`). Element mavjudligiga ishonch bo'lmasa, avval tekshiring: `if (el) { ... }`.

1.3. Matnni o'zgartirish

2-jadval. Matn xossalari

Xossa	Nima qiladi	Eslatma
`textContent`	Elementdagi **oddiy matn**	Xavfsiz, tavsiya etiladi
`innerHTML`	Ichidagi **HTML kod**	Teglarni ham o'qiydi/yozadi; ehtiyotkor bo'ling
`value`	`input` ning qiymati	Faqat forma elementlari uchun

```
const s = document.getElementById("sarlavha");

console.log(s.textContent);
s.textContent = "Yangi sarlavha";
console.log(s.textContent);

const r = document.getElementById("royxat");
console.log(r.innerHTML);
r.innerHTML = "<li>Yangi 1</li><li>Yangi 2</li>";
console.log(r.children.length);

const inp = document.getElementById("ism");
console.log(inp.value);
inp.value = "Vali";
console.log(inp.value);
```

Natija:

```
Salom
Yangi sarlavha
<li>Olma</li><li class="tanlangan">Nok</li><li>Uzum</li>
2
Ali
Vali
```

1.4. Atributlar

```
const inp = document.getElementById("ism");

console.log(inp.getAttribute("type"));
inp.setAttribute("placeholder", "Ismingiz");
console.log(inp.getAttribute("placeholder"));
console.log(inp.hasAttribute("disabled"));

inp.disabled = true;                  // mantiqiy atribut — xossa sifatida
console.log(inp.hasAttribute("disabled"));
inp.removeAttribute("disabled");
console.log(inp.disabled);
```

```
const s = document.getElementById("sarlavha");
console.log(s.id, s.className);      // id va class xossalari
```

Natija:

```
text
Ismingiz
false
true
false
sarlavha katta
```

1.5. Uslub va classList

Elementga uslub berishning ikki yo'li bor:

- `element.style.xossa = "qiymat"` — to'g'ridan-to'g'ri (kamdan-kam hollarda).
- `classList` bilan CSS **klass** qo'shish/olib tashlash — **tavsiya etiladigan** usul. Uslublarni CSS faylda yozasiz, JavaScript faqat klassni almashtiradi.

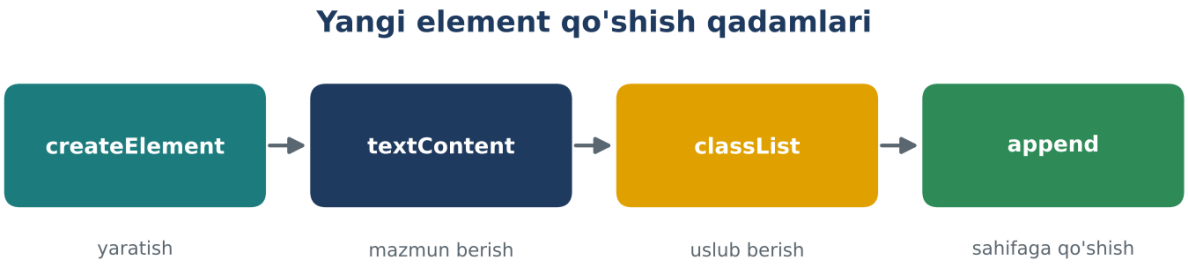
```
const s = document.getElementById("sarlavha");

s.style.color = "red";
s.style.fontSize = "32px";          // CSS font-size ?
 fontSize (camelCase)
console.log(s.getAttribute("style"));

console.log(s.classList.contains("katta"));
s.classList.add("qizil", "qalin");
s.classList.remove("katta");
console.log(s.className);
console.log(s.classList.toggle("yashirin"));    // yo'q edi ?
 qo'shildi (true)
console.log(s.classList.toggle("yashirin"));    // bor edi ?
 olib tashlandi (false)
console.log(s.className);
```

Natija:

```
color: red; font-size: 32px;
true
qizil qalin
true
false
qizil qalin
```

> **Eslab qoling**
>
> CSS dagi `font-size` JavaScript da `fontSize` bo'ladi (defis o'rniga katta harf: **camelCase**). Eng qulay yo'l: uslubni CSS da yozing, JS da `classList.toggle("yashirin")` qiling.

1.6. Yangi element yaratish va qo'shish



4-rasm. Element yaratish

```
const royxat = document.getElementById("royxat");

const yangi = document.createElement("li");   // 1) yaratish
yangi.textContent = "Anor";                    // 2) mazmun
yangi.classList.add("yangi");                  // 3) klass
royxat.append(yangi);                          // 4) sahifaga qo'shish

console.log(royxat.children.length);
console.log(royxat.lastElementChild.textContent);

// Boshiga qo'shish
const birinchi = document.createElement("li");
birinchi.textContent = "Banan";
royxat.prepend(birinchi);
console.log(royxat.firstElementChild.textContent);
```

Natija:

```
4
Anor
Banan
```

1.7. Elementni o'chirish

```
const royxat = document.getElementById("royxat");

document.querySelector(".tanlangan").remove();     // elementning o'zi o'chadi
console.log(royxat.children.length);

royxat.firstElementChild.remove();
console.log(royxat.innerHTML);

royxat.innerHTML = "";                             // ichini tozalash
console.log(royxat.children.length);
```

Natija:

```
2
<li>Uzum</li>
0
```

1.8. Sahifaga JavaScript ulash

JavaScript kodini HTML ga ikki usulda ulash mumkin. Hamma vaqt `<script>` ni `</body>` dan oldin yoki `defer` bilan yozing — shunda kod ishga tushganda elementlar tayyor bo'ladi.

index.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Mening sahifam</title>
  <script src="app.js" defer></script>   <!-- defer: HTML tugagach ishlaydi --
</head>
<body>
  <h1 id="sarlavha">Salom</h1>
  <button id="tugma">Bos</button>
</body>
</html>
```

app.js

```
// app.js
const sarlavha = document.getElementById("sarlavha");
const tugma = document.getElementById("tugma");

tugma.addEventListener("click", () => {
  sarlavha.textContent = "Tugma bosildi!";
});
```

> **Eng mashhur xato**
>
> `<script>` ni `<head>` ga `defer` siz yozsangiz, kod ishga tushganda elementlar hali yaratilmagan bo'ladi va `getElementById` `null` qaytaradi. Yechim: `defer` yoki `<script>` ni `</body>` dan oldin yozish.

1.9. Kod izi

```
const r = document.getElementById("royxat");
const li = document.createElement("li");
li.textContent = "Anor";
r.append(li);
r.firstElementChild.remove();
console.log(r.children.length);
```

3-jadval. Kod izi

Qadam	Ro'yxat (li lar)	Izoh
1	Olma, Nok, Uzum	3 ta element

Qadam	Ro'yxat (li lar)	Izoh
2–3	Olma, Nok, Uzum	`li` yaratildi, hali sahifada yo'q
4	Olma, Nok, Uzum, Anor	`append` qo'shdi
5	Nok, Uzum, Anor	Birinchi element o'chirildi
6	3 chiqadi	Uzunlik 3

1.10. Mini loyihalar

dom_loyiha.js

```
// 1) Massivdan ro'yxat yasash
const mevalar = ["Olma", "Nok", "Uzum", "Anor"];
const ul = document.createElement("ul");
for (const m of mevalar) {
  const li = document.createElement("li");
  li.textContent = m;
  ul.append(li);
}
document.body.append(ul);
console.log(ul.outerHTML);

// 2) Barcha paragraflarni kattalashtirish
document.querySelectorAll(".matn").forEach((p) => {
  p.textContent = p.textContent.toUpperCase();
});
console.log(document.querySelector(".matn").textContent);

// 3) Sanagich matni
const n = document.querySelectorAll("li").length;
const info = document.createElement("p");
info.textContent = `Ro'yxatda ${n} ta element bor`;
document.body.append(info);
console.log(info.textContent);
```

Natija:

```
<ul><li>Olma</li><li>Nok</li><li>Uzum</li><li>Anor</li></ul>
BIRINCHI ABZAS
Ro'yxatda 7 ta element bor
```

1.11. Eng ko'p uchraydigan xatolar

4-jadval. Tipik xatolar

Xato	Sababi	To'g'ri yozuv
`getElementById("#id")` (# bilan)	Bu usul # ni qabul qilmaydi	`getElementById("id")` yoki `querySelector("#id")`

Xato	Sababi	To'g'ri yozuv
`querySelectorAll(...).textContent = ...`	`NodeList` ning o'zida bunday xossa yo'q	`forEach` bilan har biriga
Element `null` bo'lib qolgan	Skript sahifadan oldin ishlagan yoki id noto'g'ri	`defer`, id ni tekshiring
`el.style.font-size = ...`	Defis JS da ishlamaydi	`el.style.fontSize`
Foydalanuvchi matnini `innerHTML` ga yozish	Xavfli (XSS)	`textContent` ishlating
`classList.add(".qizil")`	Nuqta yozilmaydi	`classList.add("qizil")`

1.12. Mashqlar

1. `h1` ning matnini o'zgartiring.
2. Barcha `.matn` paragraflarining matniga "!" qo'shing.
3. `input` ning qiymatini o'qib, konsolga chiqaring va o'zgartiring.
4. `h1` ga `classList.toggle` bilan klassni qo'shib-olib tashlang.
5. Ro'yxatga yangi `li` qo'shing va uni boshiga ham qo'shing.
6. Ro'yxatdagi oxirgi elementni o'chiring.
7. `.tanlangan` klassli elementning matnini chiqaring.
8. Massivdagi 5 ta ismdan `ul` yasab sahifaga qo'shing.

1.13. O'zingizni tekshiring

5-jadval. Qisqa test

№	Savol	Javob
1	DOM nima?	HTML ning obyektlar daraxti
2	DOM ga kirish eshigi?	`document`
3	Bitta element tanlash?	`getElementById`, `querySelector`
4	Hammasini tanlash?	`querySelectorAll`
5	Oddiy matnni o'zgartirish?	`textContent`
6	Klass qo'shish?	`classList.add`
7	Element yaratish?	`document.createElement`

№	Savol	Javob
8	Elementni o'chirish?	`element.remove()`

6 va undan ko'p to'g'ri javob — **2-darajaga** tayyorsiz.

Xulosa

DOM — sahifaning obyektlar daraxti. `querySelector` bilan tanlanadi, `textContent`, `classList`, `setAttribute` bilan o'zgartiriladi, `createElement` + `append` bilan yangi element qo'shiladi, `remove()` bilan o'chiriladi.

Chuqurroq o'rganmoqchimisiz?

Bu mavzuning o'rta va murakkab darajalari JavaScript adaptiv kursida — daraja testi sizga mos materialni o'zi tanlaydi, oxirida sertifikat.