

3.1. JavaScript asoslari

Chop etilgan: 2015 yil 31-mayYangilandi: 2020 yil 23-iyul47 ta



JavaScript Netscape brauzerlari uchun skript tili sifatida yaratilgan. Microsoft shuningdek, uning potentsialini tan oldi va uni **JScript** nomi bilan **qo'shdi**. Internet Explorer 3-da til standartlariga qisman ko'mak berib, natijada standartlar va JavaScript versiyalari bilan chalkashlikka olib keldi. Shu sababli, Netscape, Microsoft va boshqa manfaatdor kompaniyalar ECMA (Evropa Kompyuter ishlab chiqaruvchilari uyushmasi) tashkilotiga murojaat qilishdi, u erda ECMA-262 tilining birinchi spetsifikatsiyasi tasdiqlandi. "JavaScript" nomi ro'yxatdan o'tgan savdo belgisi bo'lganligi sababli, yangi standart uchun ECMAScript (yoki qisqasi ES) dan foydalanishga qaror qilindi. Dastlab ECMAScript ssenariy tili sifatida foydalanish uchun ishlab chiqilgan, ammo keyinchalik keng tarqalgan bo'lib dasturlash tili sifatida ishlatila boshlandi.

JavaScript

JavaScript-ning yaratilishi joriy hujjatni (serversiz skript deb nomlangan) qayta yuklamasdan HTML hujjat ob'ektlarini dinamik boshqarish g'oyasiga asoslandi. Vaqt o'tishi bilan tilning imkoniyatlari kengaydi:

- JavaScript-dan foydalanib, bitta sahifali dasturlarni mijoz tomonida ishlatishingiz mumkin;
- Server serverida JavaScript Node.js kabi texnologiyalardan foydalaniladi;
- JavaScript ish stoli dasturlarini Electron yordamida yaratishga yordam beradi va Raspberry Pi singari yagona taxtali kompyuterlar bilan ishlashda ishlatilishi mumkin.
- Shuningdek, JavaScript TensorFlow.js texnologiyasidan foydalangan holda brauzerda mashina o'rganish modellarini o'rgatish uchun ishlatiladi.

JavaScript nima?

1) JavaScript bu skript tili yoki skript tili. Skript bu dastur kodi - ishga tushirishdan oldin (masalan, kompilyatsiya) qayta ishlashni talab qilmaydigan ko'rsatmalar to'plami. Veb-sahifa yuklanganda JavaScript kodi brauzer mexanizmi tomonidan talqin qilinadi. Brauzer tarjimoni asl dastur yoki so'rovni sintaktik tahlil qilish, qayta ishlash va bajarishni amalga oshiradi.

2) JavaScript - bu prototipli merosga ega ob'ektga yo'naltirilgan til. U bir nechta o'rnatilgan ob'ektlarni qo'llab-quvvatlaydi va shuningdek o'zingizning shaxsiy (shaxsiy) ob'ektlaringizni yaratishingiz yoki yo'q qilishga imkon beradi. Ob'ektlar xususiyatlarni bir-biridan to'g'ridan-to'g'ri meros qilib olishlari mumkin, bu protetib ob'ektlar zanjirini hosil qiladi.

<https://html5book.ru/osnovy-javascript>

1/9

20.08.2020

JavaScript asoslari

Veb-sahifalardagi JavaScript

Tarkibi:

1. Skriptlarni html-document ga ulash
2. JavaScript-dagi ma'lumotlar turlari va o'zgaruvchilar
 - 2.1. JavaScript-dagi o'zgaruvchilar
 - 2.2. O'zgaruvchan ma'lumotlar turlari
 - 2.3. Global va mahalliy o'zgaruvchilar

1. Skriptlarni html-document ga ulash

JavaScript skriptlari **o'rnatilgan**, ya'ni. ularning tarkibi hujjatning bir qismi va **tashqi**, kengaytma bilan alohida faylda saqlanadi `.js`. Skriptlarni html hujjatiga quyidagi usullar bilan kiritish mumkin:

Giperhavola ko'rinishida.

Buning uchun kodni alohida faylga joylashtirishingiz va sarlavhadagi faylga havolani kiritishingiz kerak

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

Html

yoki sahifaning tanasi.

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

Html

<https://html5book.ru/osnovy-javascript>

2/9

Ushbu usul odatda katta skriptlar yoki turli veb-sahifalarda qayta ishlatiladigan skriptlar uchun ishlatiladi.

Hodisani boshqaruvchi sifatida .

Har bir html elementida ma'lum bir vaqtda ishlaydigan JavaScript hodisalari mavjud. Siz html-elementga atribut sifatida kerakli hodisani qo'shishingiz va kerakli funktsiyani ushbu atributning qiymati sifatida belgilashingiz kerak. Ishdan voqea farovonligini javoban deb nomlangan bir funktsiyasi **bir voqea ishleyicisi** . Hodisa qo'zg'atilishi natijasida bog'liq kod bajariladi. Ushbu usul asosan qisqa stsenariylarda qo'llaniladi, masalan, tugma bosilganda fon rangini o'zgartirish uchun sozlashingiz mumkin:

```
<script>
var colorArray = ["#5A9C6E", "#A8BF5A", "#FAC46E", "#FAD5BB", "#F2FEFF"]; // создаем массив с цветами фона
var i = 0;

function changeColor(){
    document.body.style.background = colorArray[i];
    i++;
    if( i > colorArray.length - 1){
        i = 0;
    }
}
</script>
<button onclick="changeColor();">Change background</button>
```

Html

Elementning ichida <script> .

Element <script> hujjatning istalgan joyiga joylashtirilishi mumkin. Tegning ichida brauzer tomonidan o'qilgandan so'ng darhol bajariladigan yoki chaqirilayotgan vaqtda bajariladigan funktsiyaning tavsifi mavjud. Funktsiyaning tavsifi har qanday joyda joylashtirilishi mumkin, asosiysi shundaki, nomlangan vaqtda funktsiya kodi allaqachon yuklangan.

Odatda JavaScript kodi hujjat (element <head>) boshiga yoki ochilish tegidan keyin joylashtiriladi <body> . Agar sahifani, masalan, hisoblagichni yuklaganingizdan so'ng skript ishlatilsa, uni hujjatning oxiriga qo'yish yaxshiroqdir:

```
<footer>
<script>
document.write("Введите свое имя");
</script>
```

Html

<https://html5book.ru/osnovy-javascript>

3/9

```
</footer>
</body>
```

2. JavaScript-dagi ma'lumotlar turlari va o'zgaruvchilar

Kompyuterlar ma'lumot - ma'lumotlarni qayta ishlaydilar. Ma'lumotlar har xil shakllarda yoki turlarda taqdim etilishi mumkin. Ko'pgina JavaScript funktsiyalari oddiy ob'ektlar to'plami va ma'lumotlar turlaridan foydalangan holda amalga oshiriladi. Satrlar, raqamlar va mantiq bilan bog'liq funktsional imkoniyatlar satr, raqamli va mantiqiy ma'lumotlar turlariga asoslangan. Oddiy iboralar, sanalar va matematik operatsiyalarni o'z ichiga olgan boshqa funktsiyalar RegExp, Date va Math ob'ektlari bilan bajariladi.

Javoblar JavaScript - bu ma'lumotlar turining maxsus sinfi, uchta ma'lumot turidan bittasining - qiymat, raqamli yoki mantiqiy qiymatning sobit qiymatlari:

```
"это строка"
3.14
true
alert("Hello"); // "Hello" - это литерал
var myVariable = 15; // 15 - это литерал
```

JavaScript

Oddiy ma'lumotlar turi - bu ma'lum bir turdagi ma'lumotlarning namunasidir, masalan, satr, son, boolean `null` va `undefined` .

2.1. JavaScript-dagi o'zgaruvchilar

JavaScript tomonidan ishlangan ma'lumotlar **o'zgaruvchan** . O'zgaruvchilarga dasturni bajarish paytida o'zgarishi mumkin bo'lgan ma'lumotlarni (qiymatlarni) kompyuter xotirasida saqlaydigan konteynerlar deyiladi. O'zgaruvchilar **nomi** , **turi** va **qiymatiga ega** .

O'zgaruvchan nom yoki **identifikator** faqat harflar `a-z` , `A-Z` raqamlar `0-9` (raqam o'zgaruvchan nomdagi birinchi bo'lishi mumkin emas), belgi `_` (faqat o'zgaruvchi yoki funktsiya nomidagi birinchi belgi bo'lishi mumkin) va pastki chiziqcha belgilaridan iborat `_` bo'lishi mumkin. O'zgaruvchan ism uzunligi cheklanmagan. O'zgaruvchan nomlarni rus alifbosi harflarida yozish mumkin, ammo tavsiiya etilmaydi, buning uchun ular Unicode-da yozilishi kerak.

Siz JavaScript kalit so'zlarini o'zgaruvchi nom sifatida ishlata olmaysiz. O'zgaruvchan nomlar JavaScript-da harflar hisobga olinadi, bu o'zgaruvchilar `var message;` va `var Message;` har xil o'zgaruvchilar degan ma'noni anglatadi .

`var` O'zgaruvchi, masalan, o'zgaruvchining nomi bilan birga kalit so'z bilan yaratiladi (e'lon qilinadi) `var message;` . O'zgaruvchini ishlatishdan oldin uni e'lon qilishingiz kerak.

O'zgaruvchi tayinlash operatsiyasidan foydalangan holda qiymat bilan **ishga** tushiriladi `=` , masalan `var message="Hello";` , ya'ni. o'zgaruvchini yaratish `message` va **asl** qiymati saqlanadi `"Hello"` . O'zgaruvchi qiymatsiz e'lon qilinishi mumkin, bu holda unga odatiy qiymat beriladi `undefined` . Ssenariyni bajarish paytida o'zgaruvchining qiymati o'zgarishi mumkin. Bir xil satrda turli xil o'zgaruvchilar e'lon qilinishi mumkin, ularni vergul bilan ajratish mumkin:

```
var message="Hello", number_msg = 6, time_msg = 50;
```

JavaScript

2.2. O'zgaruvchan ma'lumotlar turlari

JavaScript - bu ochilmagan til, uni e'lon qilishda ma'lum bir o'zgaruvchiga ma'lumot turini ko'rsatish shart emas. O'zgaruvchining ma'lumotlari turi olingan qiymatlarga bog'liq. O'zgaruvchining turi ma'lumotlar bilan operatsiyalarni bajarish jarayonida o'zgarishi mumkin (**dinamik typing**). Turlari ishlatiladigan kontekstga qarab avtomatik ravishda o'zgartiriladi. Masalan, operator va raqamli qiymatlarni o'z ichiga olgan iboralarda `+` , JavaScript raqamli qiymatlarni satrlarga o'zgartiradi:

<https://html5book.ru/osnovy-javascript>

5/9

```
var message = 10 + " дней до отпуска"; // вернет "10 дней до отпуска"
```

JavaScript

Operator yordamida o'zgarmaydigan ma'lumot turlarini olishingiz mumkin `typeof` . Ushbu operator mos keladigan turni aniqlaydigan satrni qaytaradi.

```
typeof 35; // вернет "number"
typeof "text"; // вернет "string"
typeof true; // вернет "boolean"
typeof [1, 2, 4]; // вернет "object"
typeof undefined; // вернет "undefined"
typeof null; // вернет "object"
```

JavaScript

JavaScript-dagi barcha ma'lumotlar turlari ikki guruhga bo'linadi - **oddiy** ma'lumotlar turlari (*ibtidoiy ma'lumotlar turlari*) va **aralashma** ma'lumotlar turlari (*kompozit ma'lumotlar turlari*) .

Uchun **oddiy** ma'lumotlar turlari satrni, son, mantiqiy, ham o'z ichiga oladi `null` va `underfined` .

2.2.1. Qator turi

Ikki yoki bitta tirnoq bilan o'ralgan belgilar qatorini saqlash uchun ishlatiladi. Bir yoki ikki tirnoq bilan o'ralgan bo'sh belgilar to'plami - bo'sh satr. Qo'shtirnoq raqami ham simdir.

```
var money = ""; // пустая строка, ноль символов
var work = 'test';
var day = "Sunday";
var x = "150";
```

JavaScript

Siz bitta tirnoqni qo'shtirnoq ichidagi qatorga qo'shishingiz mumkin va aksincha. Xuddi shu turdagi tirnoq orqaga qaytarish belgisi bilan o'chiriladi `\` (**qochish ketma-ketligi** deb ataladi):

```
document.writeln("\u0414\u043e\u0431\u0440\u0435 \u0443\u0442\u0440\u043e, \u0418\u0432\u0430\u043d \u0418\u0432\u0430\u043d\u044b\u0447!\n"); // выведет на экран "Доброе утро, Иван Иванович!"
```

Qatlamlarni taqqoslash va ulash operatsiyasidan foydalangan holda bog'lash mumkin `+`. Avtomatik majburlash sizga raqamlar va simlarni birlashtirishga imkon beradi. Strings doimiy bo'lib, satr yaratilgandan so'ng uni o'zgartirish mumkin emas, lekin boshqa satrlarni bog'lash orqali yangi satr yaratilishi mumkin.

2.2.2. Raqam turi (raqam)

Raqamli qiymatlar uchun ishlatiladi. Javascript tilidagi raqamlar ikki xil: sonlar (*butun son*) va suzuvchi nuqta raqamlari (*suzuvchi nuqta raqami*). Masalan `1`, butun sonlar musbat `2` va manfiy bo'lishi mumkin, masalan `-1`, `-2` yoki nolga teng. `1` va `1.0` - bir xil qiymat. JavaScript-dagi ko'p sonlar o'nlik belgilar bilan yozilgan, sakkizlik va o'n oltilik tizimlardan ham foydalanish mumkin.

Yilda o'nlik tizimi, raqamli o'zgaruvchilar qiymatlari arab raqamlari bilan belgilanadi `1, 2, 3, 4, 5, 6, 7, 8, 9, 0`.

Yilda sakkiz qirrali formatda, raqamlar dan zinalari o'z ichiga olgan natija bo'ladi `0` uchun `7` va prefiks bilan boshlangan `0`.

Uchun hexadecimal formatda, bir prefiks `0x` (`0X`) bo'lgan qo'shilgan dan raqamlar bir qator kuzatib `0` uchun `9` yoki harflar `a` (`A` uchun) `f` (`F` dan qadriyatlariga mos keladigan) `10` uchun `15`.

```
var a = 120; // целое десятичное числовое значение
var b = 012; // восьмеричный формат
var c = 0xffff; // шестнадцатеричный формат
var d = 0xACFE12; // шестнадцатеричный формат
```

Suzuvchi nuqta raqamlari kasr kasr sonlari yoki eksponent raqamlardir. Raqamlarning eksponensial notifikatsiyasi quyidagi shaklni o'z ichiga oladi: kasrning o'nli kasrli raqami va undan keyin harf bilan bosh harfni `e` ham kichik va kichik harflarda ko'rsatilishi mumkin, keyin ixtiyoriy belgi `+` yoki `-` butun son ko'rsatkichi.

```
var a = 6.24; // вещественное число
var b = 1.234E+2; // вещественное число, эквивалентно 1.234 X 102
var c = 6.1e-2; // вещественное число, эквивалентно 6.1 X 10-2
```

2.2.3. Boolean turi

Ushbu tur ikki xil ma'noga ega, `true` (haqiqiy), `false` (noto'g'ri). Shartlarni taqqoslash va tekshirish uchun ishlatiladi.

```
var answer = confirm("Вам понравилась эта статья?\n Нажмите ОК. Если нет, то нажмите Cancel.");
if (answer == true)
{
    alert("Спасибо!");
}
```

Oddiy qiymatlarning maxsus turlari ham mavjud:

null turi - bu tip mavjud `null` bo'lmagan ob'ektlarni ifodalash uchun ishlatiladigan bitta qiymatga ega.

aniqlanmagan turi - o'zgaruvchining turi o'zgaruvchining `undefined` boshlang'ich qiymati, shuningdek ob'ektning mavjud bo'lmagan mulki yo'qligini anglatadi.

Kompozit ma'lumotlar turlari bir nechta qiymatlardan iborat. Bularga ob'ektlar va maxsus turdagi ob'ektlar - massivlar va funktsiyalar kiradi. Ob'ektlar xususiyatlar va usullarni o'z ichiga oladi, massivlar elementlarning indekslangan to'plamidir, va funktsiyalar ko'rsatmalar to'plamidan iborat.

2.3. Global va mahalliy o'zgaruvchilar

Ko'lami bo'yicha o'zgaruvchilar **global** va **mahalliy** bo'linadi. **Miqdori** - bu o'zgaruvchiga bog'liq bo'lgan skriptning o'zgaruvchiga bog'liqligi va uning qiymatini qaytaradigan qismi. Funktsiya tanasida e'lon qilingan o'zgaruvchilar **mahalliy** deb nomlanadi va ulardan faqat shu funktsiyada foydalanish mumkin. Mahalliy o'zgaruvchilar tegishli funktsiya bilan birgalikda yaratiladi va yo'q qilinadi.

Element `<script>` ichida yoki funktsiya ichida e'lon qilingan, ammo kalit so'zni ishlatmasdan o'zgaruvchilar **global** `var` deb nomlanadi. Sahifani brauzerga yuklagan vaqt davomida ularga kirish mumkin. Bunday o'zgaruvchilardan barcha funktsiyalardan foydalanish mumkin, bu ularga ma'lumot almashish imkonini beradi.

Global o'zgaruvchilar **global nomlar maydoniga** tushadi, bu alohida dastur tarkibiy qismlari o'zaro ta'sir qiladigan joy. O'zgaruvchilarni shu tarzda e'lon qilish tavsiya etilmaydi, chunki shunga o'xshash o'zgaruvchan nomlar har qanday boshqa kod tomonidan ishlatilishi mumkin,

bu esa skriptning ishdan chiqishiga olib keladi.

JavaScript-dagi global makon global ob'ekt bilan ifodalanadi `window`. Global o'zgaruvchilar qo'shish yoki o'zgartirish global ob'ektni avtomatik ravishda yangilaydi. O'z navbatida, global ob'ektni yangilash global nomlar maydonini avtomatik ravishda yangilaydi.

Agar global va mahalliy o'zgaruvchilar bir xil nomga ega bo'lsa, unda mahalliy o'zgaruvchi global darajadan ustun turadi.

Turli xil kod bloklarida funktsiya ichida e'lon qilingan mahalliy o'zgaruvchilar bir xil doiraga ega. Shu bilan birga, barcha o'zgaruvchan deklaratsiyalarni funktsiyaning boshida joylashtirish tavsiya etiladi.