



Front-end (HTML / CSS / JavaScript)
Тема: JavaScript —

Программалар



Arrays (массивы)

Массив - (специальная переменная) бирден ашык маани камтылган өзгөчө өзгөрмө.

JS массивдери бир гана өзгөрмөгө бир нече маанини сактоо үчүн колдонулат.

массивдин синтаксиси:

```
let arrays = [item1, item2, ...];
```

Эгерде сизде элементтердин тизмеси болсо,
H: машиналардын тизмеси, анда машиналарды сактоо өзүнчө
өзгөрмөлөргө мындай көрүнүшү мүмкүн:

```
var car1 = "Mercedes";
```

```
var car2 = "Volvo";
```

```
var car3 = "BMW";
```

Бирок, машинаны аралап, белгилүү бир нерсени табууну
кааласак кандай кылабыз? Если алар 3 эмес 300 болуп калса
кандай кылабыз?

JS' те бир ат менен тузулгон озгормо менен биз коптогон маани камтышыбыз мумкун, ал маанилерге кайрылыш учун элементин индексин колдонобуз.

H:

```
let cars = ["Mercedes", "Volvo", "BMW"];
```

```
let cars = [  
  "Mercedes",  
  "Volvo",  
  "BMW"  
];
```

Arrays (массивы)

массивди түзүү үчүн эки синтаксис варианты бар:

```
let arr = new Array( );
```

```
let arr = [ ];
```

Негизи ар дайым синтаксистин экинчи версиясы колдонулат.

```
let fruits = ["Яблоко", "Апельсин", "Слива"];
```

Массивдин элементтери нөлдөн баштап, номерленет.

```
let fruits = ["Яблоко", "Апельсин", "Слива"];
```

```
alert( fruits[0] );      // Яблоко
```

```
alert( fruits[1] );      // Апельсин
```

```
alert( fruits[2] );      // Слива
```

...массивди алмаштырсак да болот:

```
let fruits = [ "Яблоко", "Апельсин", "Слива" ];
```

```
fruits[2] = 'Груша';    // ["Яблоко", "Апельсин", "Груша"]
```

... же массивге жаңы элемент кошсок да болот:

```
fruits[3] = 'Лимон';    // ["Яблоко", "Апельсин", "Груша", "Лимон"]
```

length

Массивдеги элементтердин жалпы санын чыгарыш
учун **length** свойствасы колдонулат.

```
let fruits = [ "Яблоко", "Апельсин", "Слива" ];  
alert( fruits.length );    // 3
```

массивы - методуы

Методуы pop / push / shift / unshift

push - Массивтин аягына элемент кошот.

unshift - Массивтин башына элемент кошот.

shift - Массивтин башындагы элементи жок кылат,
(экинчи турган элемент биринчи болуп калат).

pop - Массивтин акыркы элементин жок кылат

Массивдин аягы менен иштөө:

Метод pop

pop - Акыркы элементти жок кылат.

```
let fruits = [ "Яблоко", "Апельсин", "Груша" ];
```

```
alert ( fruits.pop() );      // удаляем "Груша"
```

```
alert ( fruits );           // Яблоко, Апельсин
```

Массивдин аягы менен иштөө:

Метод push

push - Аягына элемент кошот

```
let fruits = [ "Яблоко", "Апельсин" ];
```

```
fruits.push("Груша");
```

```
alert( fruits );    // Яблоко, Апельсин, Груша
```

Массивдин башталышы менен иштөө:

Memog shift

shift - Массивден биринчи элементти жок кылат

```
let fruits = ["Яблоко", "Апельсин", "Груша"];
```

```
alert( fruits.shift() );           // удаляем Яблоко
```

```
alert( fruits );                  // Апельсин, Груша
```

Массивдин башталышы менен иштөө:

Memog unshift

unshift - Башына элемент кошот

```
let fruits = [ "Апельсин", "Груша" ];
```

```
fruits.unshift ("Яблоко");
```

```
alert ( fruits );           // Яблоко, Апельсин, Груша
```

push/unshift Бир нече элементти бир заматта
кошо алам:

```
let fruits = ["Яблоко"];  
fruits.push("Апельсин", "Груша");  
fruits.unshift("Ананас", "Лимон");  
alert ( fruits );  
// ["Ананас", "Лимон", "Яблоко", "Апельсин", "Груша"]
```

Arrays (массивы)

Массивти негизи объект жана озгочо структура деп эсептешибиз керек, порядкаланган маалыматтар менен иштоого мумкундук берген. Бул үчүн массивде атайын методдор бар. Массивдер JS' те бир эле түрдөгү маалымат менен иштөө үчүн жакшы болот, эгерде бизде маалыматтарыбыздын тузулушу ар кандай түрдө болсо анда объект колдонгонубуз жакшы болот.