

Разработка front-end

Язык JavaScript

План лекции

1. Исполнение JavaScript на странице.
2. Синхронное и асинхронное выполнение скриптов на странице.
3. Особенности синтаксиса.
4. Способы задания функций.
5. Область видимости.
6. Работа с DOM-деревом и события.

Возможности JavaScript

1. Обработка действий пользователя (кликов, нажатий на кнопки и т.д.).
2. Работа с DOM-деревом.
3. Работа с cookies.
4. Взаимодействие с окном браузера.
5. Взаимодействие с сервером.

Исполнение JavaScript на странице

```
<!DOCTYPE ...>
<html>
  <head>...</head>
  <body>
    ...
    <script>
      ...
    </script>
    ...
  </body>
</html>
```

```
<!DOCTYPE ...>
<html>
  <head>
    ...
    <script src="*.js"></script>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

Асинхронные скрипты

```
<!DOCTYPE ...>
<html>
  <head>...</head>
  <body>
    ...
    <script async>
      ...
    </script>
    ...
  </body>
</html>
```

```
<!DOCTYPE ...>
<html>
  <head>...</head>
  <body>
    ...
    <script defer>
      ...
    </script>
    ...
  </body>
</html>
```

Модальные окна

```
alert("Hello, world!");
```

```
/* возвращает введенную  
строку */
```

```
var promptRes = prompt("Сколько вам лет?");
```

```
// возвращает true/false
```

```
var confirmRes = confirm("Вы действительно хотите покинуть  
страницу?");
```

Переменные и типы данных

```
var a; // undefined - значение не присвоено
```

```
a = "Hello, world!"; // string - строка
```

```
a = 5; // number - число
```

```
a = true; // boolean - логический тип (булевый тип)
```

```
a = {}; // object - объект
```

```
typeof( a ); // узнать тип переменной
```

Специальные значения

`var a; // значение переменной - undefined`

`3 * "строка"; // значение NaN - ошибка вычислений`

`5 / 0; // значение Infinity`

`a = null; // значение переменной не известно`

Условные операторы

```
var a = 5;
```

```
if ( a > 3 ) {  
    ...  
} else {  
    ...  
}
```

```
var b = ( a > 3 ) ? true : false;
```

Конструкция Switch

```
var a = 5;  
  
switch ( a ) {  
    case 3:  
        ...  
        break;  
    case 5:  
        ...  
        break;  
    default:  
        ...  
        break;  
}
```

Циклы

```
for ( var i = 0; i < 10; i++ ) {  
    ...  
}
```

```
while ( 1 ) {  
    ...  
}
```

```
do {  
    ...  
} while ( 1 )
```

Операторы сравнения

// осуществляется приведение типа, все операторы вернут true

```
( 3 >= "1" );
```

```
( 3 < "5" );
```

```
( 3 == "3" );
```

```
( 3 != "5" );
```

// тип не приводится, вернет false

```
( 3 === "3" );
```

Сравнение с null

```
// все операторы вернут true
```

```
( null == undefined ); // равны только друг другу  
( null >= 0 ); // null будет приведен к 0
```

```
// все операторы вернут false
```

```
( null == 0 ); // null равен только undefined  
( null > 0 ); // null равен только undefined
```

Способы задания функций

```
function funcName ( a, b, ... ) {  
    ...  
}
```

```
var funcName = function ( a, b, ... ) {  
    ...  
}
```

```
var funcName = function me( a, b, ... ) {  
    ...  
}
```

Область видимости переменных

```
function myFunction () {  
    if ( 1 ) {  
        var a = 5;  
    }  
    console.log ( a ); // выведется 5  
}  
console.log( a ); // выведется undefined
```

Работа с DOM-деревом

```
document.body.style.color = "red";
```

```
document.getElementById( "tag-types" ).style.color =  
"green";
```

```
document.body.childNodes;
```

```
document.getElementById( "tag-types" ).parentElement;
```


События

```
document.getElementById( "parrot" ).onmouseover =  
    function () { ... };
```

```
document.getElementById( "parrot" ).onclick =  
    function () { ... };
```

```
<div id="parrot" onclick="...">  
    ...  
</div>
```

Полезные ссылки

1. <http://learn.javascript.ru/>
2. <https://www.w3schools.com/jsref/default.asp>

Спасибо за внимание!