

Конспект «Прокрутка и операторы»

Окно браузера и прокрутка

Окно (или вкладка) браузера в JavaScript обозначается словом `window`.

Обработчик событий onscroll

Чтобы отследить прокрутку, используем обработчик событий `onscroll`. Он срабатывает каждый раз, когда страницу прокручивают, даже если её сдвинули всего на один пиксель.

```
window.onscroll = function () {  
    console.log('Страница прокручена');  
}
```

Свойство pageYOffset

Свойство `pageYOffset` окна браузера содержит количество пикселей, на которое пользователь прокрутил страницу по вертикали:

```
// Если мы на самом верху страницы  
console.log(window.pageYOffset); // Выведет: 0  
  
// Прокрутим страницу на 200px  
console.log(window.pageYOffset); // Выведет: 200
```

Величина горизонтальной прокрутки хранится в свойстве `pageXOffset`.

Метод scrollTo

Чтобы прокрутить страницу, используем метод `scrollTo`:

```
window.scrollTo(координата-X, координата-Y);
```

Координата X указывает, куда нужно прокрутить страницу по горизонтали, а координата Y — куда нужно прокрутить страницу по вертикали. Когда браузер выполнит инструкцию, указанная точка окажется в левом верхнем углу окна. Координаты задаются в пикселях, но указывать единицы измерения не нужно:

```
// Прокрутит страницу на 100px вправо и на 50px вниз
```

```
window.scrollTo(100, 50);
```

Если прокрутить страницу до указанных координат не получается, браузер прокрутит её так далеко, как сможет, но увеличивать страницу не будет. Если страница помещается в окно целиком и полосы прокрутки нет, то браузер проигнорирует эту инструкцию.

Обработчик событий onchange

Обработчик событий `onchange` срабатывает, когда пользователь выбирает новое значение из выпадающего списка.

```
// Находим выпадающий список

let select = document.querySelector('select');

// Добавляем обработчик событий

select.onchange = function () {

    // Выводим в консоль новое значение

    console.log(select.value);

};
```

Обработчик событий `onchange` можно использовать с разными элементами. Например, он срабатывает, когда пользователь переключает [чекбокс](#) или [радиокнопки](#).

Оператор строгого равенства

Чтобы проверить, равны ли два значения, используем оператор строгого равенства. Он обозначается тремя знаками равно:

```
'a' === 'a'; // Результат: true

'a' === 'b'; // Результат: false
```

Оператор строгого равенства сравнивает два значения и возвращает `true`, если они равны, и `false`, если они не равны. Значения, которые проверяет оператор, называют *операндами*.

Оператор строгого неравенства

Оператор строгого неравенства работает противоположно оператору строгого равенства. Он сравнивает два значения и возвращает `false`, если значения равны, и `true`, если они не равны.

Оператор строгого неравенства обозначается восклицательным знаком и двумя знаками равно:

```
'a' !== 'a'; // Результат: false
'a' !== 'b'; // Результат: true
```

В JavaScript также есть оператор *нестромого* равенства `==` и оператор нестромого неравенства `!=`. О них мы поговорим в одной из [следующих частей](#).

Оператор логическое И

Чтобы объединить две части условия, используем логическое И. Оно обозначается двойным амперсандом — `&&`.

```
if (article.dataset.category !== filter.value && filter.value !== 'all') {
    article.classList.add('hidden');
}
```

Логическое И возвращает `true`, только если обе части условия возвращают `true`. Если хотя бы одна из частей вернёт `false`, то условие целиком также будет считаться ложным:

```
true && true;    // Результат: true
true && false;   // Результат: false
false && true;   // Результат: false
false && false;  // Результат: false
```

Оператор логическое ИЛИ

Логическое ИЛИ обозначается двумя вертикальными чертами и возвращает `true`, если хотя бы один из операндов возвращает `true`:

```
true || true;   // Результат: true
true || false;  // Результат: true
false || true;  // Результат: true
false || false; // Результат: false
```

