

Плавный скролл до элемента на JavaScript

Разработаем следующее: после клика по ссылке будет происходить плавный скролл до элемента на JavaScript. Реализуем это без использования плагинов, сначала при помощи библиотеки jQuery, а затем на чистом JavaScript

Плавный скролл до якоря.

С помощью jQuery.

Для всех ссылок, начинающихся с #.

На чистом JavaScript - метод `Element.scrollToView()`.

На чистом JavaScript - метод `Window.scrollTo()`.

Сначала в HTML реализуем обычный «грубый» переход к секции, а затем уже добавим jQuery для создания плавности скролла.

Добавим `codeid` секциям к которым мы хотим выполнять скролл, а ссылкам атрибут `codehref` в формате `code#name`. Решетка - означает идентификатор (id), а name - имя идентификатора.

Наш HTML файл получится примерно такой.

```
<nav class="scrollto">
  <ul>
    <li><a href="#element1">Ссылка-1</a></li>
    <li><a href="#element2">Ссылка-2</a></li>
    <li><a href="#element3">Ссылка-3</a></li>
  </ul>
</nav>

<section id="element1">Секция-1</section>
<section id="element2">Секция-2</section>
```

```
<section id="element3">Секция-3</section>
```

Атрибут href указывает к какому элементу необходимо осуществить переход.

Плавный скролл с использованием jQuery

Теперь добавим плавности с помощью библиотеки jQuery. В js-файле добавьте следующий код.

```
$('.scrollto a').on('click', function() {  
  
    let href = $(this).attr('href');  
  
    $('html, body').animate({  
        scrollTop: $(href).offset().top  
    }, {  
        duration: 370, // по умолчанию «400»  
        easing: "linear" // по умолчанию «swing»  
    });  
  
    return false;  
});
```

Чтобы добавить плавность перехода для необходимого элемента, просто добавьте навигации класс `scrollto`

Изменить скорость скролла можно поменяв значение `duration`

Плавный скролл для всех ссылок, начинающихся с

Для этого мы обратимся к селектору атрибута тега `a` при помощи `^` - что будет означать, что мы выбираем все ссылки, начинающиеся с `#` (решётки)

```
$(a[href^="#"]).on('click', function() {  
  
    let href = $(this).attr('href');  
  
    $('html, body').animate({  
        scrollTop: $(href).offset().top  
    });  
    return false;  
});
```

Плавный скролл на чистом JavaScript - метод `Element.scrollToView()`

Будем использовать метод `scrollIntoView()`. Это стандартный нативный метод JavaScript.

```
const smoothLinks = document.querySelectorAll('a[href^="#']');  
for (let smoothLink of smoothLinks) {  
    smoothLink.addEventListener('click', function (e) {  
        e.preventDefault();  
        const id = smoothLink.getAttribute('href');  
  
        document.querySelector(id).scrollIntoView({  
            behavior: 'smooth',  
            block: 'start'  
        });  
    });  
};
```

У метода `scrollIntoView()` есть недостаток.

Если у нас будет навигация `position: fixed`, то нам необходимо добавить отступ сверху на `x`-пикселей, т.е. высоту навигации.

Для решения данной задачи отлично подойдёт метод `Window.scrollTo()`.

Плавный скролл на чистом JavaScript - метод `Window.scrollTo()`

Метод `Window.scrollTo()` имеет параметр `top` в котором мы укажем количество пикселей, на сколько нам необходимо прокрутить страницу.

От общей высоты документа отнимем высоту навигации и получим необходимое смещение в пикселях по оси `Y`.

```
document.querySelectorAll('a[href^="#"]').forEach(link => {

  link.addEventListener('click', function(e) {
    e.preventDefault();

    let href = this.getAttribute('href').substring(1);

    const scrollTarget = document.getElementById(href);

    const topOffset = document.querySelector('.scrollto').offsetHeight;
    // const topOffset = 0; // если не нужен отступ сверху
    const elementPosition = scrollTarget.getBoundingClientRect().top;
    const offsetPosition = elementPosition - topOffset;

    window.scrollTo({
      top: offsetPosition,
      behavior: 'smooth'
    });
  });
});
```

Версия #1

Admin создал 21 марта 2025 14:36:59

Admin обновил 21 марта 2025 14:40:47