

Slider

Подробное объяснение кода:

1. Переменные и элементы DOM

```
javascript
```

```
const track = document.querySelector('.slider-track');
```

- **Что делает?** Находит элемент с классом `slider-track` в HTML-документе.
- **Для чего?** Это основной контейнер, содержащий все слайды. Мы будем изменять его позицию с помощью CSS-свойства `transform`.

```
javascript
```

```
const slides = document.querySelectorAll('.slider-item');
```

- **Что делает?** Находит все элементы с классом `slider-item` и сохраняет их в виде коллекции (NodeList).
- **Для чего?** Эти элементы представляют каждый слайд. Мы будем использовать их для расчёта ширины и количества слайдов.

```
javascript
```

```
const prevBtn = document.querySelector('.control.prev');  
const nextBtn = document.querySelector('.control.next');
```

- **Что делает?** Находит кнопки "Назад" (`.control.prev`) и "Вперёд" (`.control.next`) в DOM.

- **Для чего?** Эти кнопки будут переключать слайды назад и вперёд.

2. Инициализация текущего слайда

javascript

```
let currentIndex = 0; // Текущий слайд
```

- **Что делает?** Устанавливает начальный индекс активного слайда (первый слайд).
- **Для чего?** Мы будем увеличивать или уменьшать этот индекс при переключении слайдов.

3. Функция для обновления позиции слайдов

javascript

```
function updateSliderPosition() {  
  const slideWidth = slides[0].clientWidth; // Ширина одного  
  слайда  
  track.style.transform = `translateX(-${currentIndex * slide  
  width}px)`;  
}
```

- **Что делает?**
 1. Рассчитывает ширину одного слайда с помощью `clientWidth` (включает ширину + внутренние отступы).

В данном контексте `clientWidth` — это свойство DOM-элемента, которое возвращает ширину внутренней области элемента, **включая ширину содержимого и внутренние отступы (padding)**, но **без учёта границ (border) и полосы прокрутки (scrollbar)**.

Как работает `clientWidth` в примере:

```
javascript

const slideWidth = slides[0].clientWidth;
```

1. Что происходит?

- `slides[0]` выбирает первый слайд из коллекции `slides` (все элементы с классом `.slider-item`).
- `clientWidth` измеряет ширину этого слайда.

2. Что возвращает?

- Ширину видимой части первого слайда.

3. Для чего это нужно?

- В данном случае ширина одного слайда используется для вычисления расстояния, на которое нужно сдвинуть весь трек (`slider-track`) при переключении слайдов:

```
javascript

track.style.transform = `translateX(-${currentIndex
* slideWidth}px)`;
```

Детали работы `clientWidth`:

1. Включает:

- Ширину содержимого (content).
- Внутренние отступы (padding).

2. Не включает:

- Границы элемента (border).

- Внешние отступы (margin).
 - Полосу прокрутки (если есть).
1. Сдвигает весь `slider-track` на расстояние, равное ширине текущего слайда умноженной на `currentIndex`.
 - Например, если `currentIndex = 1` и ширина слайда 300px, трек сдвинется на `300px`.
- **Для чего?** Чтобы показать нужный слайд, мы используем `transform: translateX()` для изменения позиции.
-

4. Обработчик для кнопки "Назад"

javascript

```
prevBtn.addEventListener('click', () => {  
  if (currentIndex > 0) {  
    currentIndex--;  
    updateSliderPosition();  
  }  
});
```

- **Что делает?**
 1. Проверяет, находится ли слайдер не на первом слайде (`currentIndex > 0`).
 2. Уменьшает `currentIndex` на 1.
 3. Вызывает функцию `updateSliderPosition()`, чтобы обновить позицию трека.
 - **Для чего?** Чтобы переключать слайдер на предыдущий слайд.
-

5. Обработчик для кнопки "Вперёд"

javascript

```
nextBtn.addEventListener('click', () => {  
  if (currentIndex < slides.length - 1) {  
    currentIndex++;  
    updateSliderPosition();  
  }  
});
```

- **Что делает?**

1. Проверяет, находится ли слайдер не на последнем слайде (`currentIndex < slides.length - 1`).
2. Увеличивает `currentIndex` на 1.
3. Вызывает функцию `updateSliderPosition()` для обновления позиции.

- **Для чего?** Чтобы переключать слайдер на следующий слайд.

6. Обработчик события `resize`

javascript

```
window.addEventListener('resize', updateSliderPosition);
```

- **Что делает?** Пересчитывает ширину слайда и обновляет позицию трека при изменении размера окна браузера.
- **Для чего?** Это полезно для адаптивного дизайна. Например, если пользователь уменьшит или увеличит окно, ширина слайдов изменится, и позиция трека должна быть скорректирована.

Общее объяснение:

- **HTML:**

- Содержит контейнеры для слайдов и кнопки управления.
- **CSS:**
 - Создаёт базовый стиль для слайдера с `display: flex` для размещения слайдов в ряд.
 - Использует `overflow: hidden` для скрывания частей трека, выходящих за границы видимой области.
- **JavaScript:**
 - **Управление слайдами:**
 - Использует кнопки для увеличения или уменьшения индекса `currentIndex`.
 - Изменяет стиль трека через `transform`, чтобы переключаться между слайдами.
 - **Респонсивность:**
 - Слушает событие `resize` для адаптации слайдов к новым размерам окна.

Результат:

Когда пользователь нажимает на кнопки "Назад" или "Вперёд", слайдер перемещается, показывая следующий или предыдущий слайд. При изменении размера окна слайды корректируются, чтобы избежать визуальных ошибок.