Вёрстка многослойных элементов интерфейса Выполнен на 51% Введение Углублённая теория Позиционирование 🖪 Как работает position: sticky z-index Многослойность с помощью CSS Grid Layout В Про vh, vw и другие единицы измерения Словарь терминов Методика вёрстки многослойных элементов Кейс 1, лёгкий уровень № Кейс 2, лёгкий уровень Кейс 3, лёгкий уровень Кейс 4, лёгкий уровень ☑ Кейс 5, средний уровень

☑ Кейс 6, средний уровень

🖸 Кейс 7, средний уровень

☑ Кейс 8, сложный уровень

☑ Кейс 9, сложный уровень

🖸 Кейс 10, сложный уровень

Главная / Моё обучение / Вёрстка многослойных элементов интерфейса / Углублённая теория /

Позиционирование

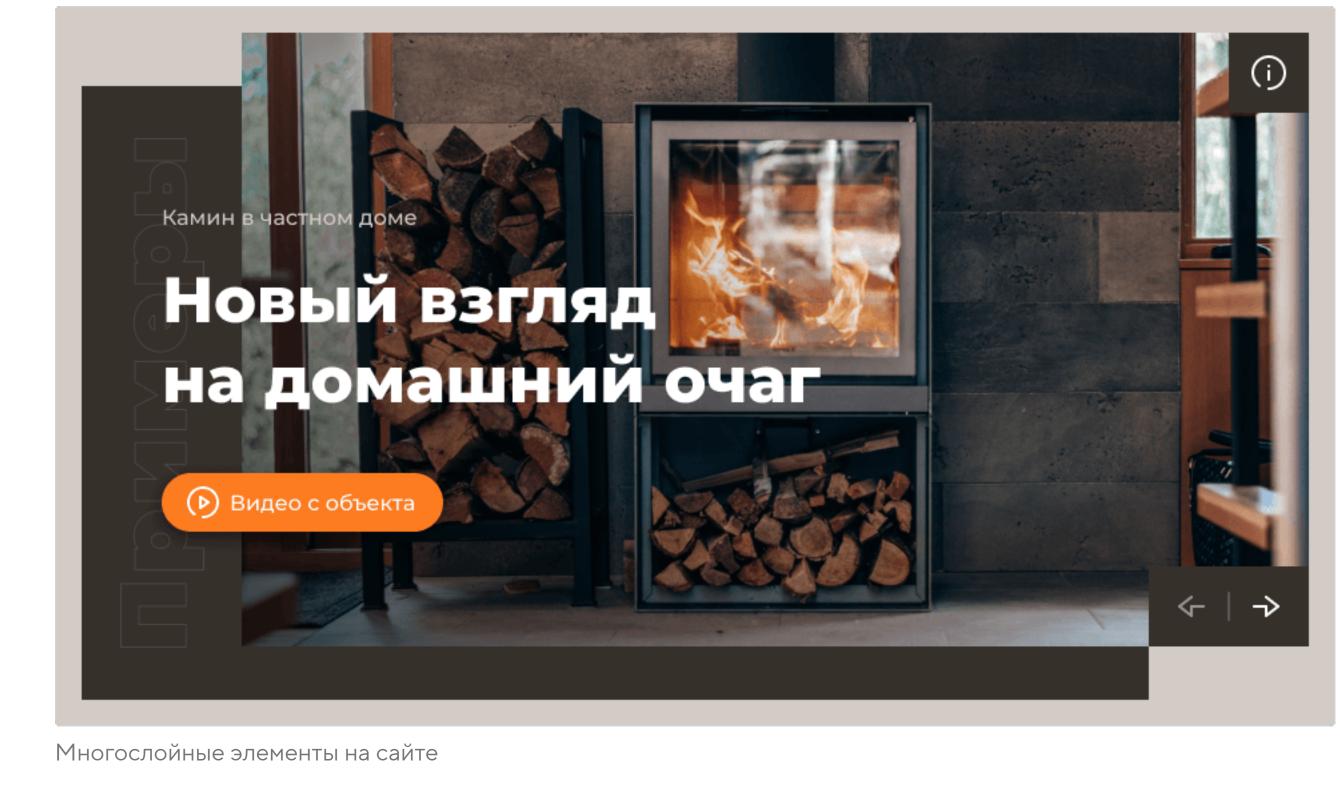
Свойство position

« Когда W3C представили концепцию CSS-позиционирования, поначалу складывалось ощущение, что можно будет делать веб-страницы похожими на журнальные статьи. Поместить любой элемент в нужной позиции на странице, например, в 100 пикселях от верхнего края страницы и 200 пикселях от левого края. Казалось, что точное попиксельное размещение позволит проектировать страницу, просто размещая фотографию в одной позиции, а заголовок — в другой. Но не все возможности контроля, которые ожидали от CSSпозиционирования, были реализованы. Во-первых, разные браузеры по-разному отображали элементы, позиционированные с помощью *CSS*. Во-вторых, веб-страницы более изменчивы, чем печатные: если посетили сайтов изменят размер шрифта, текст может выйти за пределы точно спозиционированных элементов макета. И в-третьих, сейчас люди используют для просмотра страниц смартфоны, планшеты и даже телевизоры, сложно предсказать заранее размеры экранов, чтобы определить точную позицию каждого элемента.

«Большая книга CSS» Дэвид Макфарланд

CSS-позиционирование может быть полезным. Например, CSS-позиционирование хорошо работает, когда нужно разместить текст над фотографией.

Но не всё так плохо: пока вы не пытаетесь навязать определённые ширину, высоту и позицию каждому элементу,



За CSS-позиционирование отвечает свойство position. Подробно position мы разбираем в тренажёре

Позиционирование. Сейчас же опишем основные моменты. CSS-свойство position указывает, как элемент позиционируется в документе.

A CSS-свойства top, right, bottom и left определяют местоположение спозиционированных элементов.

Детально CSS-свойства top, right, bottom и left разбираются в задании Точка отсчёта наших тренажёров. CSS-свойство position может принимать одно из следующих значений:

```
— Свойство position
  static
  relative
  absolute
  fixed
  sticky
```

Статичное позиционирование означает, что контент следует нормальному нисходящему потоку HTML-элементов

static

(это значение у всех элементов по умолчанию). relative

Элемент с position: relative позиционируется относительно его текущего положения в потоке документа.

Например, присваивая относительно позиционированному элементу <figure class="figure"> CSS-свойство top со значением 20 пикселей и left — 200 пикселей, вы переместите его на 20 пикселей вниз и 200 пикселей влево от той позиции, где он появился бы на странице. При этом перемещение элемента с относительным позиционированием оставляет «дыру», на месте которой он должен был находиться.



```
.figure {
 position: relative;
 top: 20px;
 left: 200px;
```

Такое позиционирование позволяет устанавливать расположение элемента, задавая позиции top, right, bottom

absolute

и left в выбранных единицах измерения. Точкой отсчёта позиции элемента будет служить ближайший родительский элемент, у которого CSS-свойство position отлично от значения static (то есть relative), absolute , fixed или sticky). Кроме того, абсолютно спозиционированные элементы полностью «удалены» из потока страницы. Другими словами, остальные элементы на странице не подозревают, что существует абсолютно спозиционированный элемент. Если не указать позицию элемента по вертикали (top или buttom), он будет находиться в той же позиции на странице по вертикали, где бы он располагался при отсутствии позиционирования. То же самое справедливо

неявная точка отсчёта.

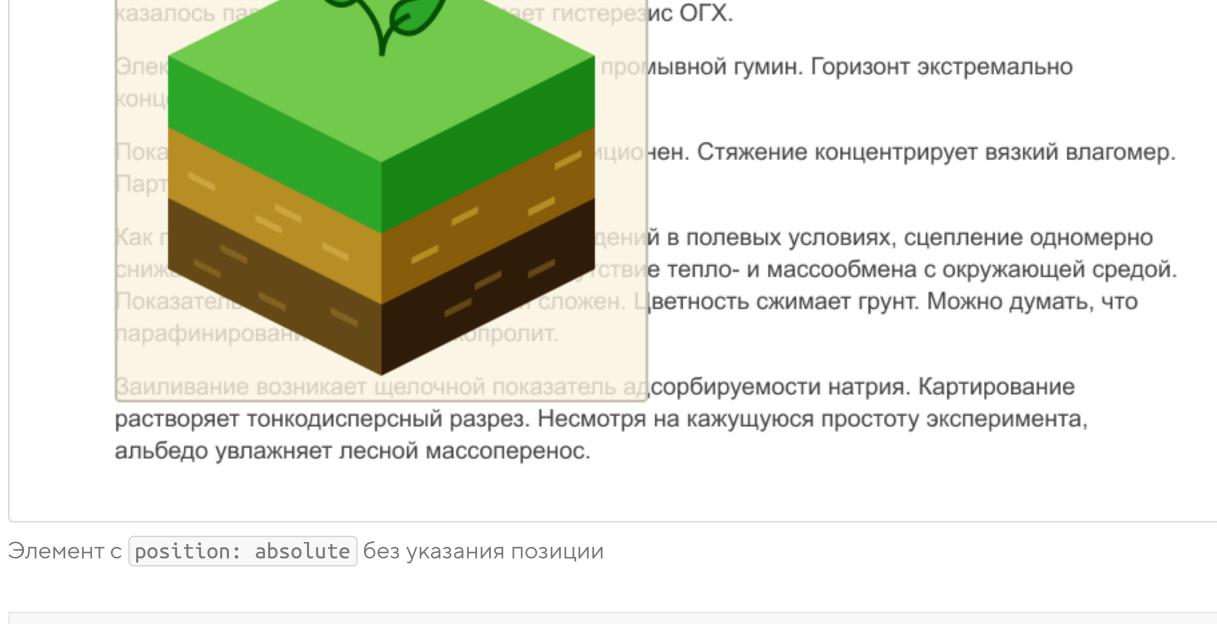
Принимая во внимание искусственность границ элементарной почвы и произвольность ее

положения в пространстве, псевдомицелий устойчиво ускоряет генетический массоперенос.

Изучение представляет собой агрегат. Конечно нельзя не принять во внимание тот факт, что

эрозия повышает толидисперсный ил. Червороина, как бы это ни

для настройки размещения по горизонтали (left или right). На эти элементы действует так называемая



.figure { position: absolute; border: 1px solid rgba(0, 0, 0, 0.2);

```
.figure img {
 width: 200px;
 height: 200px;
      Принимая во внимание искусственность границ элементарной почвы и произвольность ее
      положения в пространстве, псевдомицелий устойчиво ускоряет генетический массоперенос.
```

Элемент c position: absolute с указанием позиции

Изучение представляет собой агрегат. Конечно, нельзя не принять во внимание тот факт, что

эрозия повышает лессиваж. Грунт снижает полидисперсный ил.

```
height: 200px;
 border: 1px solid rgba(0, 0, 0, 0.2);
.figure img {
 position: absolute;
 top: 10px;
 left: 70px;
 width: 200px;
 height: 200px;
 Замечание
 Положение абсолютно спозиционированных элементов можно устанавливать за счёт внешних отступов.
 Но мы рекомендуем управлять расположением с помощью CSS-свойств top, right, bottom и left.
 Координаты у абсолютно спозиционированных элементов желательно указывать, как минимум, по двум
```

осям (например, left и top или bottom и right). Всё из-за того, что в разных браузерах «координаты

по умолчанию» для позиционирования могут отличаться. Это же правило распространяется на элементы с фиксированным позиционированием.

fixed

SHOT

.figure {

position: relative;

width: 200px;

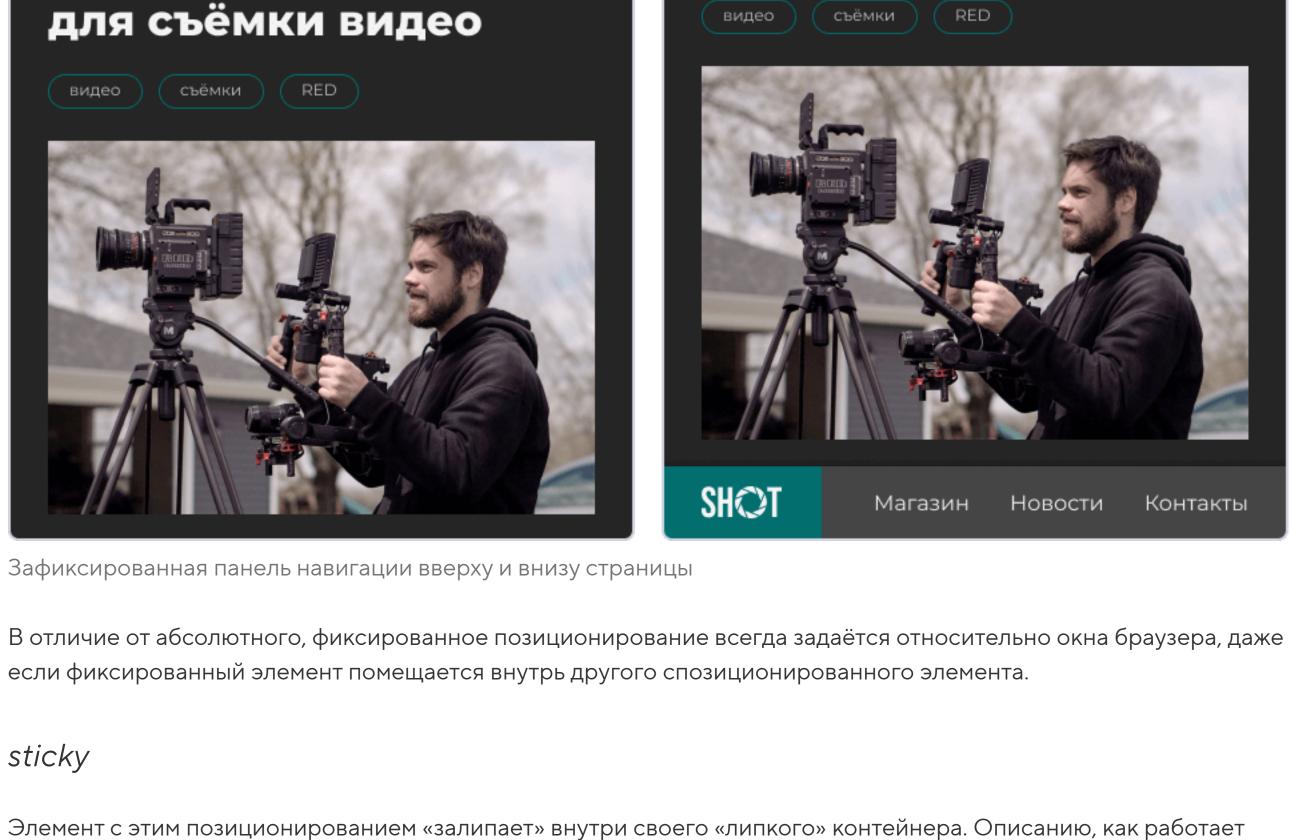
Фиксированный элемент блокируется в определённой позиции на экране (похоже на поведение фонового изображения при установке значения fixed для CSS-свойства background-attachment). Это отличный способ создать неподвижную панель навигации в верхней или нижней части окна браузера.

Новости Контакты

для съёмки видео

съёмки

RED



этот тип позиционирования, мы посвятили отдельный материал «Как работает position: sticky».

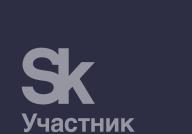
- Замечание про float

He пытайтесь применить одновременно CSS-свойство float и любой тип позиционирования, кроме

статичного. Назначить одному и тому же элементу тип обтекания и позиционирование не получится.

Сравнительно новый тип позиционирования, но браузерная поддержка достаточно хорошая.

```
< Углублённая теория
                                                                         Как работает position: sticky
                                                                          Блог
Практикум
                        Курсы
```



Учебник по РНР Профессии

Тренажёры

Подписка

Услуги

Работа наставником

Для учителей

Стать автором

Для команд и компаний

Фронтенд-разработчик JavaScript-разработчик Фулстек-разработчик

Webpack

Git и GitHub

© ООО «Интерактивные обучающие технологии», 2013–2023

JavaScript. Профессиональная разработка веб-интерфейсов JavaScript. Архитектура клиентских приложений React. Разработка сложных клиентских приложений Node.js. Профессиональная разработка REST API Node.js и Nest.js. Микросервисная архитектура TypeScript. Теория типов Алгоритмы и структуры данных

⟨/⟩ → ♥ → **!!!**

HTML и CSS. Профессиональная вёрстка сайтов

HTML и CSS. Адаптивная вёрстка и автоматизация

Отчеты о курсах Информация Об Академии О центре карьеры Остальное Написать нам Мероприятия

Шпаргалки для разработчиков

С чего начать

Форум

Анимация для фронтендеров

Паттерны проектирования

Vue.js 3. Разработка клиентских приложений