

# ОСНОВЫ веб-технологий

HTML & CSS

- Визуальные эффекты
- Bootstrap 5

# Визуальные эффекты и анимация

- Использование CSS переходов – свойство `transition`
- Трансформации в CSS - свойство `transform`
- Анимация в CSS - свойство `animation` и `@keyframes`

# Свойство transition

CSS transitions предоставляют способ контролировать скорость анимации, при изменении CSS-свойств. Вместо того, чтобы свойство применилось сразу, вы можете сделать это действие происходящим в течение какого-то момента времени. Например, если вы смените цвет элемента с белого на чёрный, изменение произойдёт моментально, а вот с CSS transitions, изменения произойдут за временные интервалы, следующих кривой ускорения, все из которых могут быть настроены.

**transition** позволяет задать значения `transition-property`, `transition-duration`, `transition-timing-function` и `transition-delay`.

## **transition-property**

Указывает имя или имена свойств, чьи переходы должны анимироваться. Только свойства, указанные здесь, анимируются в переходах; изменение других свойств будет происходить обычным образом.

```
transition-property: all;
```

```
transition-property: font-size;
```

```
transition-property: width, height;
```

## **transition-duration**

Свойство `transition-duration` определяет продолжительность выполнения анимации. Значение по умолчанию равняется 0s, т.е. отсутствие анимации.

## **transition-delay**

Определяет как много должно пройти времени, перед тем как начнётся переход.

# Свойство transition

## transition-timing-function

Устанавливает, насколько быстро должно изменяться значение стилевого свойства для которого применяется эффект перехода.

transition-timing-function: ease|ease-in|ease-out|ease-in-out|linear|step-start|step-end|steps|cubic-Bezier

- ease - анимация начинается медленно, затем ускоряется и к концу движения опять замедляется.
- ease-in - анимация медленно начинается, к концу ускоряется.
- ease-out - анимация начинается быстро, к концу замедляется.
- ease-in-out - анимация начинается и заканчивается медленно.
- linear - одинаковая скорость от начала и до конца.
- step-start - как таковой анимации нет. Стилиевые свойства сразу же принимают конечное значение.
- step-end - как таковой анимации нет. Стилиевые свойства находятся в начальном значении заданное время, затем сразу же принимают конечное значение.
- steps - ступенчатая функция, имеющая заданное число шагов.
- cubic-bezier - задаёт функцию движения в виде кривой Безье.

# Пример

```
<a href="#" class="button">Ссылка</a>
```

```
a.button{
  width: 150px;
  height: 50px;
  background-color: aquamarine;
  color: blue;

  display: inline-flex;
  justify-content: center;
  align-items: center;

  text-decoration: none;
  font-weight: bold;
  letter-spacing: 2px;
  font-size: 2em;
  border: 2px solid blue;
  border-radius: 15px;

  position: relative;
  z-index: 1;
  overflow: hidden;
  transition: color 0.5s linear;
}
```

Ссылка

Ссылка

Ссылка

```
a.button::before{
  content: "";
  position: absolute;
  top: 0;
  left: 0;
  width: 0;
  height: 100%;
  background-color: blue;
  z-index: -1;
  transition: all 0.5s ease-in-out;
}
a.button:hover::before{
  width: 100%;
}
a.button:hover{
  color: cyan;
}
```

# Свойство transform

CSS свойство `transform` позволяет манипулировать элементами на вашей веб-странице, изменяя их форму, размер и положение.

В общем виде свойство `transform` записывается так: `transform: функция(значение);`, где функция — это тип преобразования, который вы хотите применить, а значение — параметры этого преобразования.

Свойство `transform` может принимать следующие значения:

- `none` — никакого преобразования не применяется.
- `rotate(angle)` — вращает элемент на заданный угол.
- `scale(x,y)` — масштабирует элемент по осям X и Y.
- `translate(x,y)` — перемещает элемент на заданные координаты.
- `skew(x-angle,y-angle)` — наклоняет элемент на заданные углы по осям X и Y.

Есть некоторые другие варианты преобразований, например, на основе матриц, про них можно почитать [тут](#).

# Пример

```
<div class="wrapper-1">  
  <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Deleniti est reiciendis omnis facilis,  
  corrupti dolore odio atque facere quia nobis, quasi commodi provident quam voluptates esse ullam eaque minima minus!  
  </p>  
    
</div>
```

```
.wrapper-1 {  
  margin: 60px;  
  width: 250px;  
  height: 360px;  
  border: 1px solid black;  
  padding: 10px;  
  border-radius: 8px;  
  overflow: hidden;  
  position: relative;  
  transform: rotate(20deg);  
}  
  
img.rotate {  
  position: absolute;  
  bottom: 0px;  
  left: 20px;  
  width: 100%;  
  display: block;  
  margin: 0 auto;  
  border-radius: 8px;  
  transform: rotate(-20deg);  
}
```



# CSS-анимация

Свойство `animation` используется для создания анимации, например движения, изменения цвета, поворотов, изменения размера элементов. `animation` определяет набор ключевых кадров — `keyframes`, которые описывают, как изменяется свойство элемента со временем.

`animation:`

- `animation-name`
- `animation-duration`
- `animation-timing-function`
- `animation-delay`
- `animation-iteration-count`
- `animation-direction`
- `animation-fill-mode`
- `animation-play-state`



# CSS-анимация

## **animation-name**

Определяет название анимации, которое соответствует определенной группе ключевых кадров — `@keyframes`.

```
@keyframes <имя кейфрейма> {  
  from { описание начального состояния элемента }  
  to { описание конечного состояния элемента }  
}
```

```
@keyframes <имя кейфрейма> {  
  0% { описание начального состояния элемента }  
  40% { описание промежуточного состояния элемента }  
  100% { описание конечного состояния элемента }  
}
```

# CSS-анимация

## **animation-duration**

Задаёт продолжительность анимации в секундах или миллисекундах.

## **animation-timing-function**

Определяет временную функцию, которая контролирует скорость и характер изменений во время анимации.

Значения:

`ease|ease-in|ease-out|ease-in-out|linear|step-start|step-end|steps|cubic-Bezier.`

## **animation-delay**

Задаёт задержку перед началом анимации в секундах или миллисекундах.

## **animation-iteration-count**

Определяет количество повторений анимации.

Значения:

- `infinite` — анимация повторяется бесконечно.
- Числовые значения — указывают конкретное количество повторений.

# CSS-анимация

## **animation-direction**

Определяет направление анимации.

Распространенные значения:

- `normal` — проигрывается в прямом направлении.
- `reverse` — проигрывается в обратном направлении.
- `alternate` — проигрывается в прямом и обратном направлениях чередуясь.

## **animation-fill-mode**

Определяет стили, применяемые к элементу до и после анимации. Распространенные значения:

`none` — стили элемента не изменяются до или после анимации.

`forwards` — элемент сохраняет стили последнего кадра после окончания анимации.

`backwards` — элемент применяет стили первого кадра до начала анимации.

## **animation-play-state**

Определяет состояние анимации: остановлена или воспроизводится.

Значения:

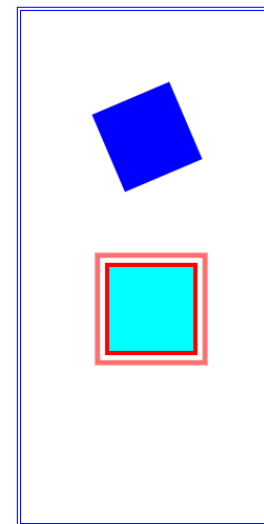
- `running` — анимация воспроизводится.
- `paused` — анимация остановлена.

# Пример

```
<div class="wrapper">
  <div class="spin"></div>
  <div class="pulse"></div>
</div>
```

```
.wrapper{
  margin: 20px;
  border: 1px double blue;
  padding: 20px;
}
.spin{
  width: 20px;
  height: 20px;
  background-color: blue;
  animation: spin 1s linear
infinite;
}
@keyframes spin {
  from {
    transform: rotate(0);
  }
  to {
    transform: rotate(360deg);
  }
}
```

```
.pulse{
  margin: 20px 0 ;
  width: 20px;
  height: 20px;
  border: 1px solid red;
  background-color: cyan;
  position: relative;
}
.pulse::before{
  content: "";
  position: absolute;
  top: -1px;
  left: -1px;
  height: 100%;
  width: 100%;
  border: 1px solid red;
  animation: pulse1 1s linear infinite;
}
@keyframes pulse1 {
  0%{
    opacity: 1;
  }
  100%{
    opacity: 0;
    transform: scale(1.5);
  }
}
```



# Bootstrap 5

Bootstrap (также известен как Twitter Bootstrap) — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS шаблоны оформления для типографики, веб форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript расширения.

Основные инструменты Bootstrap:

- Сетки — заранее заданные размеры колонок, которые можно сразу использовать.
- Шаблоны — фиксированный или резиновый шаблон документа.
- Типографика — описания шрифтов, определение некоторых классов для шрифтов, таких как код, цитаты и т. п.
- Медиа — предоставляет некоторое управление изображениями и видео.
- Таблицы — средства оформления таблиц, вплоть до добавления функциональности сортировки.
- Формы — классы для оформления форм и некоторых событий, происходящих с ними.
- Навигация — классы оформления для панелей, вкладок, перехода по страницам, меню и панели инструментов.
- Алерты — оформление диалоговых окон, подсказок и всплывающих окон.

Bootstrap 5 был официально выпущен 5 мая 2021 года

# Bootstrap 5

Официальная страница <https://getbootstrap.com/>

Русская версия <https://getbootstrap.su/>

Также можно посмотреть документацию здесь:  
<https://www.w3schools.com/bootstrap5/index.php>

# Быстрый старт

## Стили

Скопируйте и вставьте таблицу стилей `<link>` в `<head>` перед всеми остальными таблицами стилей, чтобы загрузить наш CSS. Актуальную ссылку для `link` всегда можно взять на страницах документации.

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN" crossorigin="anonymous">
```

## Скрипты

Многие из компонентов Bootstrap требуют использования JavaScript для работы. В частности, для них требуются собственные плагины Bootstrap JavaScript и Popper. Поместите один из следующих `<script>` в конце страницы `html`, прямо перед закрывающим тегом `</body>`, чтобы включить их. Актуальную ссылку на `bootstrap.bundle.js` и `bootstrap.bundle.min.js` всегда можно взять на страницах документации.

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-C6RzsynM9kWDrMNeT87bh95OGNyZPhcTNXj1NW7RuBCsyN/o0jlpcV8Qyq46cDfL" crossorigin="anonymous"></script>
```

# HTML

```
<!doctype html>
<html lang="ru">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Демо Bootstrap</title>
    <link href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN" crossorigin="anonymous">

  </head>
  <body>
    <h1>Привет мир!</h1>

    <script src="https://cdn.jsdelivrivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-C6RzsynM9kWDrMNeT87bh950GNyZPhcTNXj1NW7RuBCsyN/o0jlpcV8Qyq46cDfL"
crossorigin="anonymous"></script>
  </body>
</html>
```



# Важно!

- ❑ Bootstrap 5 требует использования **HTML5**: `<!doctype html>`
- ❑ **Адаптивный метатег.** Bootstrap разработан по принципу **Mobile First**. Чтобы обеспечить правильный рендеринг и сенсорное масштабирование для всех устройств, необходим метатег адаптивного окна просмотра (viewport):

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- ❑ **Свойство Box-sizing.** Для более простого изменения размеров в CSS в Bootstrap5 переключено глобальное значение `box-sizing` с `content-box` на `border-box`. Это гарантирует, что `padding` не влияет на окончательную вычисленную ширину элемента.

# Сохранение библиотек на локальный компьютер

В разделе Скачать/Download документации можно скачать архив исходников в Source files/Исходные файлы. В архиве нам необходимы файлы:

- `dist/js/bootstrap.bundle.min.js`
- `dist/css/bootstrap.min.css`

Данные файлы необходимо скопировать в папку проекта и подключить в html-файле:

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="css/bootstrap.min.css" rel="stylesheet" >
  <title>Document</title>
</head>
<body>

  <script src="js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

# Bootstrap 5: иконки

<https://icons.getbootstrap.ru/>

## Встраивание:

```
<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi bi-emoji-heart-eyes "
viewBox="0 0 16 16"><path d="M8 15A7 7 0 1 1 8 1a7 7 0 0 1 0 14zm0 1A8 8 0 1 0 8 0a8 8 0 0 0 0 16z"/><path d="M11.315
10.014a.5.5 0 0 1 .548.736A4.498 4.498 0 0 1 7.965 13a4.498 4.498 0 0 1 -3.898-2.25.5.5 0 0 1 .548-
.736h.005l.017.005.067.015.252.055c.215.046.515.108.857.169.693.124 1.522.242 2.152.242.63 0 1.46-.118 2.152-.242a26.58
26.58 0 0 0 1.109-.224l.067-.015.017-.004.005-.002zM4.756 4.566c.763-1.424 4.02-.12.952 3.434-4.496-1.596-2.35-4.298-
.952-3.434zm6.488 0c1.398-.864 3.544 1.838-.952 3.434-3.067-3.554.19-4.858.952-3.434z"/></svg>
```

## Подключение через CDN:

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.2/font/bootstrap-
icons.min.css">
```

## Использование:

```
<i class="bi-alarm"></i>
<i class="bi-alarm text-success"></i>
<i class="bi-alarm" style="color: red; font-size: 2em;" ></i>
```

## Скачивание и подключение:

Скачиваете исходный код Source code (zip) с <https://github.com/twbs/icons/releases/latest/>

Раскрываем архив, копируем в проект папку font, подключаем и используем.

```
<link rel="stylesheet" href="font/bootstrap-icons.css">
```

# Bootstrap 5: документация

Описание дальнейших компонентов будет основываться на русскоязычной документации версии 5.3 размещенной по адресу <https://getbootstrap.su/docs/5.3/>

Вы можете рассматривать версию bootstrap 5.3 с англоязычной документацией - размещение всех элементов, на которые будут ссылки далее, полностью аналогично русскоязычной версии.

# Bootstrap 5: контрольные точки

<https://getbootstrap.su/docs/5.3/layout/breakpoints/>

Контрольные точки - это настраиваемая ширина, которая определяет поведение адаптивного макета на разных устройствах или размерах области просмотра в Bootstrap.

Доступные контрольные точки

Bootstrap включает шесть контрольных точек по умолчанию, иногда называемых grid tiers, для быстрого построения.

Контрольная точка	Описание	Инфикс класса	Размеры
X-Small	Очень маленький	None	<576px
Small	Маленький	sm	≥576px
Medium	Средний	md	≥768px
Large	Большой	lg	≥992px
Extra large	Очень большой	xl	≥1200px
Extra extra large	Огромный	xxl	≥1400px

# Bootstrap 5: кастомизация и утилиты

- ✓ Цветовая палитра (цвет): <https://getbootstrap.su/docs/5.3/customize/color/>
- ✓ Цветовые режимы: <https://getbootstrap.su/docs/5.3/customize/color-modes/>
- ✓ Поля и отступы: <https://getbootstrap.su/docs/5.3/utilities/spacing/>
- ✓ Размеры элементов: <https://getbootstrap.su/docs/5.3/utilities/sizing/>
- ✓ Цветовые модели <https://getbootstrap.su/docs/5.3/customize/color-modes/>
- ✓ Фон (background): <https://getbootstrap.su/docs/5.3/utilities/background/>
- ✓ Цвет текста: <https://getbootstrap.su/docs/5.3/utilities/colors/>
- ✓ Границы: <https://getbootstrap.su/docs/5.3/utilities/borders/>
- ✓ Тени: <https://getbootstrap.su/docs/5.3/utilities/shadows/>
- ✓ Непрозрачность элементов: <https://getbootstrap.su/docs/5.3/utilities/opacity/>
- ✓ Текстовые утилиты: <https://getbootstrap.su/docs/5.3/utilities/text/>
- ✓ Вертикальное выравнивание встроенных элементов: <https://getbootstrap.su/docs/5.3/utilities/vertical-align/>
- ✓ Изменение свойства Display: <https://getbootstrap.su/docs/5.3/utilities/display/>
- ✓ Создание плавающего элемента: <https://getbootstrap.su/docs/5.3/utilities/float/>
- ✓ Создание Flex контейнеров: <https://getbootstrap.su/docs/5.3/utilities/flex/>

# Bootstrap 5: компоненты навигации

- ✓ Панель навигации <https://getbootstrap.su/docs/5.3/components/navbar/>
- ✓ Навигация и вкладки <https://getbootstrap.su/docs/5.3/components/navs-tabs/>
- ✓ Выпадающие списки: <https://getbootstrap.su/docs/5.3/components/dropdowns/>
- ✓ Компоненты боковой панели: <https://getbootstrap.su/docs/5.3/components/offcanvas/>
- ✓ Breadcrumbs для навигации: <https://getbootstrap.su/docs/5.3/components/breadcrumb/>

# Bootstrap 5: Контент

<https://getbootstrap.su/docs/5.3/content/typography/>

- ✓ Заголовки
- ✓ Абзацы
- ✓ Строчные текстовые элементы
- ✓ Аббревиатуры
- ✓ Цитаты
- ✓ Выравнивание
- ✓ Списки

<https://getbootstrap.su/docs/5.3/content/images/>

- ✓ Вставка изображений

<https://getbootstrap.su/docs/5.3/content/tables/>

- ✓ Таблицы



# Bootstrap 5: контейнеры

Контейнеры – это фундаментальный строительный блок Bootstrap, который содержит, дополняет и выравнивает контент в пределах данного устройства или области просмотра.

<https://getbootstrap.su/docs/5.3/layout/containers/>

Контейнеры являются самым основным элементом макета в Bootstrap и необходимы при использовании сеточной системы. Контейнеры используются для содержания, заполнения и (иногда) центрирования содержимого внутри них. Хотя контейнеры могут быть вложенными, для большинства макетов вложенный контейнер не требуется. Контейнеры были созданы без многих современных функций и методов CSS, например таких, как новая CSS Grid.

Bootstrap предлагает три вида контейнеров:

- `.container`, который устанавливает `max-width` для каждой контрольной точки
- `.container-fluid`, который равен `width: 100%` во всех контрольных точках
- `.container-{breakpoint}`, который равен `width: 100%` до указанной контрольной точки

# Bootstrap 5: Контейнеры – система сеток

Система сеток <https://getbootstrap.su/docs/5.3/layout/grid/>

Bootstrap использует набор контейнеров, строк и колонок для компоновки и выравнивания содержимого. Он построен с помощью **flexbox** и полностью адаптивен.

- Сетка поддерживает шесть адаптивных контрольных точек. Контрольные точки основаны на медиа-запросах `min-width`, то есть они влияют на эту контрольную точку и все точки больше этой (например, `.col-sm-4` применяется к `sm`, `md`, `lg`, `xl` и `xxl`). Это означает, что можно управлять размером и поведением контейнера и колонки с помощью каждой контрольной точки.
- Контейнеры центрируют и размещают контент по горизонтали. Используйте `.container` для адаптивной ширины в пикселях, `.container-fluid` для `width: 100%` во всех окнах просмотра и на всех устройствах или адаптивный контейнер (например, `.container-md`) для комбинации подвижной и пиксельной ширины.
- Ряды (`rows`) являются оболочками для колонок. Каждая колонка имеет горизонтальный `padding` (называемый промежутком) для управления пространством между ними.
- Колонки невероятно гибкие. В каждой строке доступно 12 шаблонов колонок, что позволяет создавать различные комбинации элементов, охватывающих любое количество колонок. Классы колонок указывают количество колонок шаблона, которые необходимо охватить (например, `col-4` охватывает четыре). `width` задаётся в процентах, поэтому всегда формируется одинаковый относительный размер.

# Bootstrap 5: Контейнеры — система сеток

```
<div class="container">
  <div class="row">
    <div class="col-12 col-md-6 col-lg-4">col</div>
    <div class="col-12 col-md-6 col-lg-8">col</div>
  </div>
  <div class="row">
    <div class="col-6 col-sm-4 col-md-3 col-lg-2">col</div>
    <div class="col-6 col-sm-4 col-md-3 col-lg-2">col</div>
    <div class="col-6 col-sm-4 col-md-3 col-lg-2">col</div>
    <div class="col-6 col-sm-4 col-md-3 col-lg-2">col</div>
    <div class="col-6 col-sm-4 col-md-3 col-lg-2">col</div>
    <div class="col-6 col-sm-4 col-md-3 col-lg-2">col</div>
  </div>
</div>
```

# Bootstrap 5: Ряд колонок

Используйте адаптивные классы `.row-cols-*`, чтобы быстро установить количество столбцов, которые лучше всего отображают ваш контент и макет. В то время как обычные классы `.col-*` применяются к отдельным колонкам в строке (например, `.col-md-4`), классы колонок для строк устанавливаются в родительском `.row`. С помощью `.row-cols-auto` вы можете придать столбцам их естественную ширину.

```
<div class="container border">
  <div class="row row-cols-1 row-cols-sm-2 row-cols-md-3">
    <div class="col border">col</div>
    <div class="col border">col</div>
    <div class="col border">col</div>
    <div class="col border">col</div>
    <div class="col border">col</div>
    <div class="col border">col</div>
  </div>
</div>
```

# Bootstrap 5: компоненты для контента

- ✓ Настраиваемые стили кнопок: <https://getbootstrap.su/docs/5.3/components/buttons/>  
<https://getbootstrap.su/docs/5.3/components/button-group/>
- ✓ Списки: <https://getbootstrap.su/docs/5.3/components/list-group/>
- ✓ Карточки: <https://getbootstrap.su/docs/5.3/components/card/>
- ✓ Слайд-шоу (карусель): <https://getbootstrap.su/docs/5.3/components/carousel/>
- ✓ Размещение контента Аккордеон: <https://getbootstrap.su/docs/5.3/components/accordion/>
- ✓ Модальные окна: <https://getbootstrap.su/docs/5.3/components/modal/>

# Bootstrap 5: компоненты для веб-приложений

Уведомления: <https://getbootstrap.su/docs/5.3/components/alerts/>

Паджинаторы: <https://getbootstrap.su/docs/5.3/components/pagination/>

Прогресс бары: <https://getbootstrap.su/docs/5.3/components/progress/>

Состояние загрузки: <https://getbootstrap.su/docs/5.3/components/spinners/>

Всплывающие подсказки: <https://getbootstrap.su/docs/5.3/components/tooltips/>

# Bootstrap 5: формы

Примеры и рекомендации по использованию стилей управления формами, параметров макета и настраиваемых компонентов для создания самых разных форм:

<https://getbootstrap.su/docs/5.3/forms/overview/>

✓ Элементы управления текстовой формой, такие как `<input>` и `<textarea>`:

<https://getbootstrap.su/docs/5.3/forms/form-control/>

✓ Элемент выбора (select): <https://getbootstrap.su/docs/5.3/forms/select/>

✓ Флажки и радио кнопки: <https://getbootstrap.su/docs/5.3/forms/checks-radios/>

✓ Группа полей ввода: <https://getbootstrap.su/docs/5.3/forms/input-group/>

✓ Макет формы: <https://getbootstrap.su/docs/5.3/forms/layout/>

✓ Валидация форм: <https://getbootstrap.su/docs/5.3/forms/validation/>

# Пример макета

```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Пример</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEVDwWykc2MPK8M2HN" crossorigin="anonymous">
</head>
<body>
  <body class="d-flex flex-column min-vh-100">
    <header>
      <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
        ...
      </nav>
    </header>
    <main class="flex-shrink-0" >
      <div class="container">
        <h1 class="mt-5">Прикрепленный нижний колонтитул с навигационной панелью</h1>
      </div>
    </main>
    <footer class="footer mt-auto py-3 bg-primary text-white">
      <div class="container">
        <span>Место для контента прикрепленного футера здесь.</span>
      </div>
    </footer>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-C6RzsynM9kWDrMNeT87bh95OGNyZPhcTNXj1NW7RuBCsyN/o0jlpcV8Qyq46cDfL"
      crossorigin="anonymous"></script>
</body>
</html>
```