Оглавление

[Базовые стили страницы 3](#_Toc85703326)

[Типографика 3](#_Toc85703327)

[Переменные (цвета) 3](#_Toc85703328)

[Выравнивание (текст, блок) 3](#_Toc85703329)

[Отображение 4](#_Toc85703330)

[Шрифты 4](#_Toc85703331)

[Внешние отступы 4](#_Toc85703332)

[Внутренние отступы 5](#_Toc85703333)

[Размеры блока 5](#_Toc85703334)

[Компоненты 5](#_Toc85703335)

[Аккордеон – accordion 5](#_Toc85703336)

[Разметка 5](#_Toc85703337)

[Формы - form 6](#_Toc85703338)

[Разметка 6](#_Toc85703339)

[Кнопка – btn 7](#_Toc85703340)

[Карточка – card 7](#_Toc85703341)

[Разметка 7](#_Toc85703342)

[Карусель – carousel 8](#_Toc85703343)

[С разметкой на странице 8](#_Toc85703344)

[Формирование скриптом 8](#_Toc85703345)

[Выпадающий элемент – dropdown 9](#_Toc85703346)

[Разметка 9](#_Toc85703347)

[Модальное окно – modal 9](#_Toc85703348)

[С разметкой на странице 9](#_Toc85703349)

[Формирование скриптом 10](#_Toc85703350)

[Вкладки – tab 10](#_Toc85703351)

[Разметка 10](#_Toc85703352)

[Методы – actions 11](#_Toc85703353)

[.html("contens") 11](#_Toc85703354)

[.eq(num) 11](#_Toc85703355)

[.index() 11](#_Toc85703356)

[.find("selector") 11](#_Toc85703357)

[.closest("selector") 11](#_Toc85703358)

[.siblings() 11](#_Toc85703359)

[Методы атрибутов 11](#_Toc85703360)

[.setAttrib() 11](#_Toc85703361)

[.removeAttrib() 11](#_Toc85703362)

[.toggleAttrib() 11](#_Toc85703363)

[Методы классов 12](#_Toc85703364)

[.addClass() 12](#_Toc85703365)

[.removeClass() 12](#_Toc85703366)

[.toggleClass() 12](#_Toc85703367)

[Методы отображения 12](#_Toc85703368)

[.show() 12](#_Toc85703369)

[.hide() 12](#_Toc85703370)

[.toggle() 12](#_Toc85703371)

[Методы эффектов 12](#_Toc85703372)

[.fadeIn(dur, display, fin) 12](#_Toc85703373)

[.fadeOut(dur, fin) 12](#_Toc85703374)

[.fadeToggle(dur, display, fin) 12](#_Toc85703375)

[.fadeInLeft(dur, display, fin) 12](#_Toc85703376)

[.fadeInRight(dur, display, fin) 12](#_Toc85703377)

[Методы обработчики 13](#_Toc85703378)

[.on(eventName, callback) 13](#_Toc85703379)

[.off(eventName, callback) 13](#_Toc85703380)

[.click(handler) 13](#_Toc85703381)

# Базовые стили страницы

( base.scss )

\* {

box-sizing: border-box;

margin: 0;

padding: 0;

}

.container {

max-width: 1140px;

padding: 0px 15px;

margin: 0px auto;

}

# Типографика

( typography.scss )

@import url("https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,400;0,700;1,300&display=swap");

\* {

font-family: "Roboto", sans-serif;

}

# Переменные (цвета)

( variables.scss )

$primary: #025fff;

$success: #2a9924;

$danger: #ea2e41;

$warning: #ffc907;

$dark: #343a40;

# Выравнивание (текст, блок)

( align.scss )

.text-center { text-align: center; }

.block-center { margin: 0 auto; }

.f-centered { justify-content: center; align-items: center; }

.f-space-around { justify-content: space-around; }

.f-space-between { justify-content: space-between; }

# Отображение

( display.scss )

.hidden { visibility: hidden; }

.visible { visibility: visible; }

.d-none { display: none; }

.d-block { display: block; }

.d-flex { display: flex; }

# Шрифты

( fonts.scss )

.fz-16 { font-size: 16px; }

.fz-20 { font-size: 20px; }

.fz-24 { font-size: 24px; }

.bold { font-weight: bold; }

.thin { font-weight: 300; }

.italic { font-style: italic; }

.text-color-primary { color: $primary; }

.text-color-danger { color: $danger; }

.text-color-success { color: $success; }

.text-color-warning { color: $warning; }

.text-color-dark { color: $dark; }

# Внешние отступы

( margin.scss )

.m-10 { margin: 10px; }

.mt-10 { margin-top: 10px; }

.mr-10 { margin-right: 10px; }

.mb-10 { margin-bottom: 10px; }

.ml-10 { margin-left: 10px; }

.m-20 { margin: 20px; }

.mt-20 { margin-top: 20px; }

.mr-20 { margin-right: 20px; }

.mb-20 { margin-bottom: 20px; }

.ml-20 { margin-left: 20px; }

.m-30 { margin: 30px; }

.mt-30 { margin-top: 30px; }

.mr-30 { margin-right: 30px; }

.mb-30 { margin-bottom: 30px; }

.ml-30 { margin-left: 30px; }

.m-50 { margin: 50px; }

.mt-50 { margin-top: 50px; }

.mr-50 { margin-right: 50px; }

.mb-50 { margin-bottom: 50px; }

.ml-50 { margin-left: 50px; }

.m-100 { margin: 100px; }

.mt-100 { margin-top: 100px; }

.mr-100 { margin-right: 100px; }

.mb-100 { margin-bottom: 100px; }

.ml-100 { margin-left: 100px; }

# Внутренние отступы

( padding.scss )

.p-10 { padding: 10px; }

.pt-10 { padding-top: 10px; }

.pr-10 { padding-right: 10px; }

.pb-10 { padding-bottom: 10px; }

.pl-10 { padding-left: 10px; }

.p-20 { padding: 20px; }

.pt-20 { padding-top: 20px; }

.pr-20 { padding-right: 20px; }

.pb-20 { padding-bottom: 20px; }

.pl-20 { padding-left: 20px; }

.p-30 { padding: 30px; }

.pt-30 { padding-top: 30px; }

.pr-30 { padding-right: 30px; }

.pb-30 { padding-bottom: 30px; }

.p-50 { padding: 50px; }

.pt-50 { padding-top: 50px; }

.pr-50 { padding-right: 50px; }

.pb-50 { padding-bottom: 50px; }

.p-100 { padding: 100px; }

.pt-100 { padding-top: 100px; }

.pr-100 { padding-right: 100px; }

.pb-100 { padding-bottom: 100px; }

# Размеры блока

( sizes.scss )

.w-100 { width: 100%; }

.w-50 { width: 50%; }

.w-300 { width: 300px; }

.w-500 { width: 500px; }

.h-100 { height: 100%; }

.h-50 { height: 50%; }

.h-300 { height: 300px; }

.h-500 { height: 500px; }

# Компоненты

## Аккордеон – accordion

( accordion.scss )

### Разметка

<div class="accordion mt-20 mb-20 block-center">

<div class="accordion-head">

<span>Collapse first element 1</span>

</div>

<div class="accordion-content">

<div class="accordion-inner">

Contents 1

</div>

</div>

<div class="accordion-head">

<span>Collapse second element 2</span>

</div>

<div class="accordion-content">

<div class="accordion-inner">

Contents 2

</div>

</div>

<div class="accordion-head">

<span>Collapse third element 3</span>

</div>

<div class="accordion-content">

<div class="accordion-inner">

Contents 3

</div>

</div>

</div>

#### Вызов

$(".accordion-head").accordion();

## Формы - form

( form.scss )

### Разметка

<form class="form">

<h1 class="form\_title">Заголовок</h1>

<div class="form\_grup">

<input class="form\_input" type="text" name="name" placeholder=" " />

<label class="form\_label">Введите имя</label>

</div>

<div class="form\_grup">

<input class="form\_input tel" type="tel" data-validate-field="tel" name="tel" placeholder=" " />

<label class="form\_label">Введите телефон</label>

</div>

<div class="form\_grup">

<input class="form\_input" type="email" data-validate-field="email" name="email" placeholder=" " />

<label class="form\_label">Введите email</label>

</div>

<div class="form\_grup">

<input class="form\_input" type="password" placeholder=" " />

<label class="form\_label">Пароль</label>

</div>

<div class="form\_grup">

<textarea

class="form\_input"

type="text"

name="message"

cols="30"

rows="5"

placeholder=" "

></textarea>

<label class="form\_label">Сообщение</label>

</div>

<div class="form\_grup">

<input class="form\_input file\_input" type="file" name="file[]" multiple />

<label class="form\_label">Прикрепить файл</label>

</div>

<button class="form\_button btn btn-primary" type="submit">Отправить</button>

</form>

#### Вызов

$(".form").forms();

## Кнопка – btn

( button.scss )

btn { display: inline-block; } плюс сановная стилизация

btn-block { display: block; width: 100%; }

#### Разметка

<button class="btn btn-primary m20">

Button text

</button>

#### Залитая кнопка цвет фона, при наведении темнеет

btn-primary

btn-success

btn-danger

btn-warning

btn-dark

#### Белая кнопка меняется цвет фона при наведении

btn-outline-primary

btn-outline-success

btn-outline-danger

btn-outline-warning

btn-outline-dark

## Карточка – card

( card.scss )

### Разметка

<div class="card">

<img class="card-img" src="https://... " alt="photo"/>

<div class="card-body">

<div class="card-title">Card title #1</div>

<p class="card-text">Text contents</p>

<a href="#" id="trigger" class="btn btn-primary" data-toggle="modal"

data-target="#exampleModal">Link to</a>

</div>

</div>

## Карусель – carousel

( carousel.scss )

### С разметкой на странице

<div class="carousel" id="example">

<ol class="carousel-indicators">

<li data-slide-to="0" class="active"></li>

<li data-slide-to="1"></li>

<li data-slide-to="2"></li>

<li data-slide-to="3"></li>

</ol>

<div class="carousel-inner">

<div class="carousel-slides">

<div class="carousel-item">

<img src="" alt="name"/>

</div>

<div class="carousel-item">

<img src="" alt="name"/>

</div>

<div class="carousel-item">

<img src="" alt="name"/>

</div>

</div>

</div>

<a href="#" class="carousel-prev" data-slide="prev">

<span class="carousel-prev-icon">&lt;</span>

</a>

<a href="#" class="carousel-next" data-slide="next">

<span class="carousel-next-icon">&gt;</span>

</a>

</div>

#### Вызов

$("#example").carousel(autoPlayDur);

При autoPlayDur = 0 или не указанном, авто листания слайдов не будет.

### Формирование скриптом

#### Вызов

$("#example-2").createCarousel({

width: 850,

height: 450,

slides: [

{ src: "path", alt: "name", },

{ src: "path", alt: "name", },

{ src: "path", alt: "name", },

],

}).carousel(autoPlayDur);

## Выпадающий элемент – dropdown

( dropdown.scss )

### Разметка

<div class="dropdown">

<button class="btn btn-primary dropdown-toggle" id="dropdownMenuButton">

Dropdown button

</button>

<div class="dropdown-menu" data-toggle-id="dropdownMenuButton">

<a href="#" class="dropdown-item">Action #1</a>

<a href="#" class="dropdown-item">Action #2</a>

<a href="#" class="dropdown-item">Action #3</a>

</div>

</div>

#### Вызов

$(".dropdown-toggle").dropdown();

## Модальное окно – modal

( modal.scss )

### С разметкой на странице

<div class="modal" id="exampleModal2">

<div class="modal-dialog">

<div class="modal-content">

<button class="close" data-close>

<span>&times;</span>

</button>

<div class="modal-header">

<div class="modal-title">Modal title #2</div>

</div>

<div class="modal-body">Lorem ipsum dolor sit</div>

<div class="modal-footer">

<button class="btn btn-danger" data-close>Close</button>

<button class="btn btn-success">Save changes</button>

</div>

</div>

</div>

</div>

#### Вызов

$('[data-toggle="modal"]').modal();

### Формирование скриптом

#### Вызов

$("#trigger").click(() =>$("#trigger").createModal({

text: {

title: "Modal title №хх",

body: "Modal text",

},

btns: {

settings: [

["Close", ["btn-danger", "mr-10"], true],

[

"Save changes",

["btn-success"],

false,

() => {

alert("Данные сохранены");

},

],

[

"Another btn",

["btn-warning", "ml-10"],

false,

() => {

alert("Another alert");

},

],

],

},

})

);

Где settings: [["Тип кнопки", ["Классы настройки кнопки …", Закрытие окна по клику], [], …]

## Вкладки – tab

( tab.scss )

### Разметка

<div class="tab mt-20 block-center">

<div class="tab-panel" data-tabpanel>

<div class="tab-item tab-item--active">Content 1</div>

<div class="tab-item">Content 2</div>

<div class="tab-item">Content 3</div>

<div class="dropdown d-flex f-centered block-center">

<button class="btn btn-primary dropdown-toggle" id="dropdownMenuButton2">

Dropdown button

</button>

<div class="dropdown-menu" data-toggle-id="dropdownMenuButton2">

<a href="#" class="dropdown-item">Action #1</a>

<a href="#" class="dropdown-item">Action #2</a>

<a href="#" class="dropdown-item">Action #3</a>

</div>

</div>

</div>

<div class="tab-content tab-content--active">

Text contents

</div>

<div class="tab-content">

Text contents #1

<br />

<br />

Text contents #2

</div>

<div class="tab-content">

Text contents #3

</div>

</div>

#### Вызов

$("[data-tabpanel] .tab-item").tab();

# Методы – actions

( actions.js )

### .html("contens")

изменение или получение содержания html элемента $("selector").html("contens")

если content не передан получаем содержимое $("selector").html()

### .eq(num)

получение элемента по номеру $("selector").eq(num) где (num 1,2,3, ...)

### .index()

получение номера элемента по порядку, до одного общего родителя $("child\_selector").index()

### .find("selector")

получение элемента по селектору в пределах родителя $("parent\_selector").find("finding\_elem\_selector")

### .closest("selector")

получение ближайшего родительского элемента по селектору $("child\_selector").closest("finding\_parent\_selector")

### .siblings()

получение содедних элементов внутри родительского блока исключая сам блок вызова $("selector").siblings()

# Методы атрибутов

( attributes.js )

### 

### .setAttrib()

установка атрибута

### .removeAttrib()

удаление атрибута

### .toggleAttrib()

переключение атрибута

формат вызова $("selector").toggleAttrib("data-one", "111", "data-two", "");

# Методы классов

( classes.js )

### .addClass()

добавление классов

### .removeClass()

удаление классов

### .toggleClass()

переключение классов

формат вызова $("selector").addClass("class1", "class2", …)

# Методы отображения

( display.js )

### .show()

показать элемент

### .hide()

скрыть элемент

### .toggle()

переключить показ элемента

формат вызова $("selector").toggle()

# Методы эффектов

( effects.js )

Настройка вызовов - dur время анимации, cb колбек по необходимости после завершения, fin функция по завершении анимации, параметры не обязательные.

Формат вызова $("selector").fadeToogle(500, "flex", functionName)

### .fadeIn(dur, display, fin)

постепенное появление

### .fadeOut(dur, fin)

постепенное исчезание

### .fadeToggle(dur, display, fin)

переключение постепенного появления - исчезания

### .fadeInLeft(dur, display, fin)

постепенное появление в лево

### .fadeInRight(dur, display, fin)

постепенное появление в право

# Методы обработчики

( handlers.js )

### .on(eventName, callback)

назначение обработчика события

### .off(eventName, callback)

удаление обработчика события

### .click(handler)

назначение обработчика события клик

формат вызова $("selector").on("eventName", callbackFunctionName);