

# Работа с формой. HTML. CSS

## HTML

### input

```
<label for="formName" class="form__label">Имя*:</label>
<input id="formName" type="text" name="name" class="form__input_req">
```

### options

```
<div class="options">
  <div class="options__item">
    <input id="formRightHanded" checked type="radio" value="right" name="hand" class="options__input">
    <label for="formRightHanded" class="options__label"><span>Правша</span></label>
  </div>
  <div class="options__item">
    <input id="formLeftHanded" type="radio" value="left" name="hand" class="options__input">
    <label for="formLeftHanded" class="options__label"><span>Левша</span></label>
  </div>
</div>
```

### textarea

```
<label for="formMessage" class="form__label">Сообщение:</label>
<textarea name="message" id="formMessage" class="form__input"></textarea>
```

### select

```
<select name="age" class="form__select">
  <option value="от 18 до 25" selected>Молодой</option>
  <option value="от 25 до 45">Старый</option>
  <option value="от 45">Супер стар</option>
</select>
```

### file

```
<div class="form__item">
  <div class="form__label">Прикрепить фото</div>
  <div class="file">
    <div class="file__item">
      <input id="formImage" accept=".jpg, .png, .gif" type="file" name="image" class="file__input">
      <div class="file__button">Выбрать</div>
    </div>
    <div id="formPreview" class="file__preview"></div> //для просмотра
  </div>
</div>
```

### checkbox

```
<div class="checkbox">
  <input id="formAgreement" checked type="checkbox" name="agreement" class="checkbox__input_req">
  <label for="formAgreement" class="checkbox__label"><span>Я даю свое согласие на обработку персональных
    данных в соответствии с <a href="">Условиями</a>*</span></label>
</div>
```

Другой вариант: <https://myrusakov.ru/css3-custom-checkbox.html>

```
<label class="container">Один
  <input type="checkbox" checked="checked">
  <span class="checkmark"></span>
</label>
```

```
<label class="container">Два
  <input type="checkbox">
  <span class="checkmark"></span>
```

```
</label>
```

```
<label class="container">Три  
  <input type="checkbox">  
  <span class="checkmark"></span>  
</label>
```

### submit

```
<button type="submit" class="form__button">Отправить</button>
```

## Стилизация формы на CSS

### input

```
.form__input {  
  height: 50px;  
  padding: 0px 20px;  
  border-radius: 5px;  
  width: 100%;  
  font-size: 18px;  
  transition: all 0.5s ease 0s; // для своего стиля при фокусе  
}
```

### options

Скрываем стандартный и заменяем своим при помощи псевдоэлемента

```
.options {  
  padding: 10px 0px 0px 0px;  
}  
// скрываем стандартный  
.options__input {  
  display: none;  
}  
.options__input:checked + .options__label::after { // + смежные селекторы  
  transform: scale(1); //рисовать кружок внутри  
}  
.options__label {  
  display: inline-flex;  
  font-size: 16px;  
  line-height: 140%;  
  align-items: center; //для многострочного текста  
  position: relative; //для кружка внутри  
  cursor: pointer;  
}  
.options__label::before {  
  content: "";  
  align-self: flex-start;  
  flex: 0 0 24px;  
  height: 24px;  
  background-color: #fff;  
  border-radius: 50%;  
  margin: 0px 10px 0px 0px;  
}  
.options__label::after { //рисует кружок внутри
```

```
transition: transform 0.5s ease 0s;
content: "";
position: absolute;
top: 4px;
left: 4px;
width: 16px;
height: 16px;
background-color: #7a956b;
border-radius: 50%;
transform: scale(0); //скрытие кружка
}
```

### **textarea**

```
textarea.form__input {
    min-height: 120px;
    resize: vertical;
    padding: 20px;
}
```

### **select**

Можно стилизовать само окошко выбора, но стилизовать выпадающий список средствами CSS очень сложно. Подключаем готовый JS. Скрываем оригинальный select и строим свой со своими стилями. Делаем отдельный CSS файл и подключаем его.

### **file**

Также скрываем оригинальный и заменяем своим. Рисуем свою кнопку file\_\_button. Поверх неё накладываем file\_\_input, прозрачный с размером шрифта 0 (без этого иногда пропадает стиль для pointer).

```
.file__item { //оболочка элемента
    position: relative;
}
.file__input { // скрытый элемент
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    opacity: 0;
    font-size: 0;
    cursor: pointer;
}
.file__button { // наша кнопка
    display: inline-flex;
    min-height: 40px;
    border-radius: 30px;
    justify-content: center;
    align-items: center;
    background-color: #77608d;
    padding: 0px 20px;
}
.file__preview {
    display: inline-block;
    max-width: 100px;
    padding: 10px 0px 0px 0px;
}
.file__preview img {
```

```

    max-width: 100%;
}
checkbox
Скрываем оригинальную и рисуем свою кнопку. Все аналогично радиокнопке.
.checkbox__input { //скрываем оригинальную
    display: none;
}
.checkbox__input:checked + .checkbox__label::after { // + смежные селекторы
    transform: scale(1); //нарисовать внутри
}
.checkbox__label {
    font-size: 16px;
    line-height: 140%;
    display: inline-flex;
    align-items: center;
    position: relative;
    cursor: pointer;
}

.checkbox__label::before { //белый квадрат
    content: "";
    align-self: flex-start;
    flex: 0 0 24px;
    height: 24px;
    background-color: #fff;
    border-radius: 4px;
    margin: 0px 10px 0px 0px;
}

.checkbox__label::after { // маленький квадрат внутри ( можем вместо него вставить свою картинку)
    transition: transform 0.5s ease 0s;
    content: "";
    width: 16px;
    height: 16px;
    position: absolute; //рисуем внутри
    top: 4px;
    left: 4px;
    background-color: #7a956b;
    border-radius: 4px;
    transform: scale(0); // скрываем изначально
}

```

**Другой вариант:** <https://myrusakov.ru/css3-custom-checkbox.html>

Строчный `ter label`, заменим на блочный (`display: block`), чтобы чекбоксы, встали друг под другом, `user-select: none` – запрещает пользователю выделять элемент.

```

.container {
    display: block;
    user-select: none;
}

```

Этот код прячет дефолтные браузерные стили для чекбокса. Мы делаем элемент полностью прозрачным за счет `opacity`, `width` и `height` с нулевым значением и на их месте, задаем кастомные стили для чекбокса.

```

.container input {
    opacity: 0;
}

```

```
height: 0;
width: 0;
}
```

Создаем кастомные чекбоксы. Меняем размеры и цвет фона.

```
.checkmark {
height: 23px;
width: 22px;
background-color: #eec321;
}
```

При наведении курсора, делаем цвет фона немного темнее.

```
.container:hover input ~ .checkmark {
background-color: #ccc678;
}
```

Для отмеченного чекбокса, задаем другой цвет для фона.

```
.container input:checked ~ .checkmark {
background-color: #2196f3;
}
```

Прячем галочку для не отмеченных чекбоксов, применив псевдоэлемент after.

```
.checkmark:after {
content: "";
position: absolute;
display: none;
}
```

Делаем видимой галочку, только для отмеченных чекбоксов.

```
.container input:checked ~ .checkmark:after {

display: block;

}
```

## **submit**

```
.form__button {
width: 100%;
display: flex;
justify-content: center;
align-items: center;
min-height: 60px;
background-color: #7a956b;
color: #fff;
font-size: 18px;
text-transform: uppercase;
letter-spacing: 2px;
font-weight: 700;
cursor: pointer;
border-radius: 20px;
box-shadow: 0 5px 0 #161b13;
transition: background-color 0.5s ease 0s;
position: relative;
top: 0;
```

```

}
.form__button:hover {
    background-color: #4f6146;
}
.form__button:active { // сдвиг кнопки при нажатии
    top: 3px;
    box-shadow: 0 2px 0 #161b13;
}

```

## Стилизация состояния при фокусе

Скрываем стандартный outline:

```

.form * {
    outline: none;
}

```

Заменяем, где нужно своим:

```

.form__input:focus {
    box-shadow: 0 0 15px #7a956b;
}

```

## Стилизация ошибок

```

.form__input._error {
    box-shadow: 0 0 15px red;
}
.checkbox._error .checkbox__label::before {
    box-shadow: 0 0 15px red;
}

```

## При отправке формы

```

.form__body {
    position: relative;
}
.form__body::after { //накрываем всю форму
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: rgba(51, 51, 51, 0.9) url("../img/loading.gif") center / 50px no-repeat;
    opacity: 0;
    visibility: hidden;
    transition: all 0.5s ease 0s;
}
.form__body._sending::after {
    opacity: 1;
    visibility: visible;
}

```

## Стилизация placeholder

```
::placeholder {  
  color: blue;  
  font-size: 1.5em;  
}
```

В ::placeholder можно использовать только те CSS-свойства, которые разрешены в псевдоэлементе ::first-line:

<https://developer.mozilla.org/ru/docs/Web/CSS/::first-line>