



Знакомство с PHP

Свойства CSS - Шрифт

- **font:** Универсальное свойство, позволяющее определить все свойства шрифта за одну декларацию
- **font-family:** Задаёт семейство шрифта, которое будет использоваться для оформления текста содержимого
- **font-size:** Определяет размер шрифта.
- **font-size-adjust:** Позволяет контролировать размер шрифта, когда первый выбранный шрифт не доступен
- **font-stretch:** Устанавливает более широкое или узкое начертание шрифта
- **font-style:** Устанавливает стиль начертания шрифта - обычное, курсивное или наклонное
- **font-variant:** Определяет, как следует представлять строчные буквы - делать их все прописными уменьшенного размера или оставить без изменений
- **font-weight:** Устанавливает насыщенность или "жирность" начертания шрифта

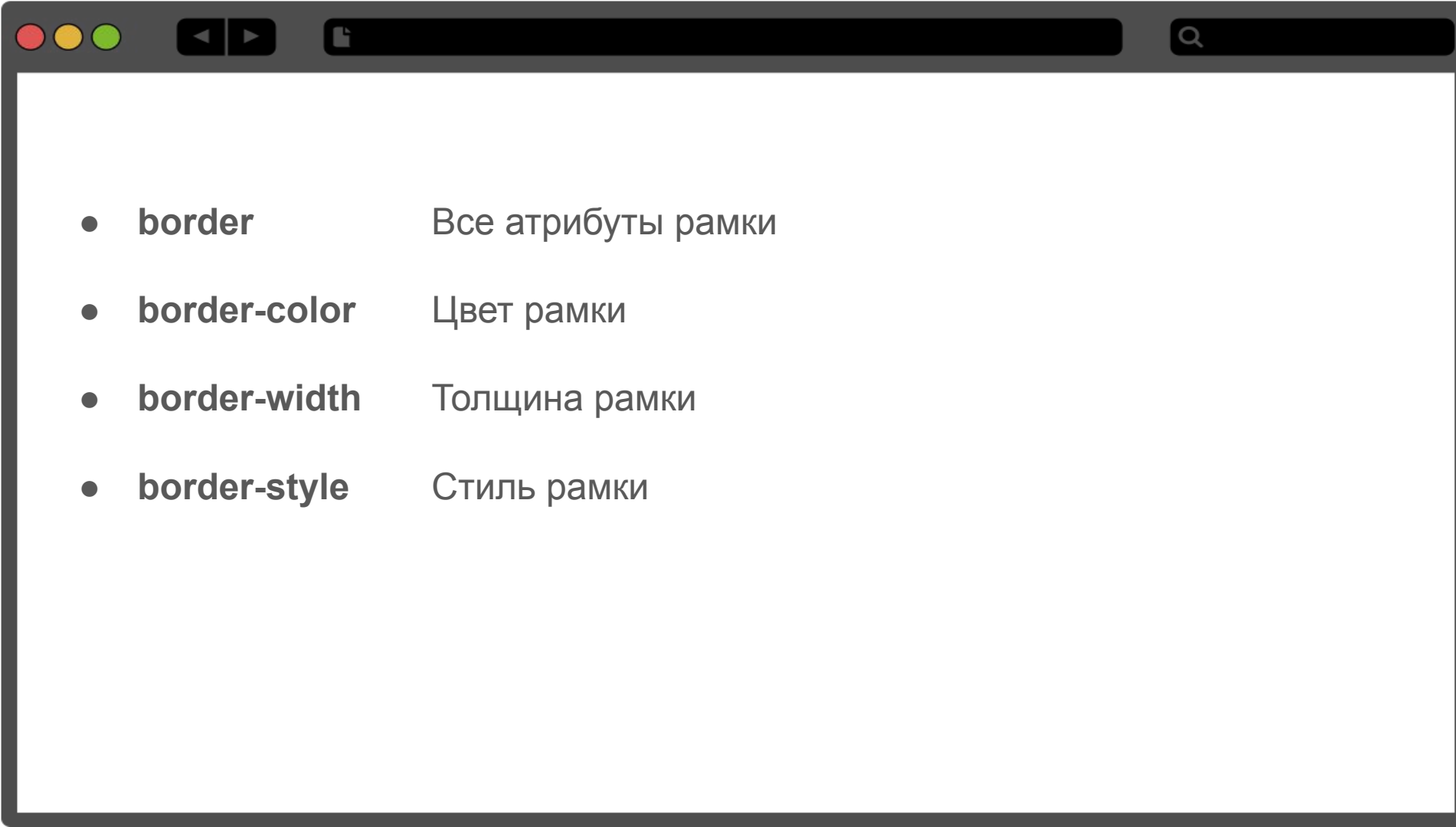
Свойства CSS - Цвет и фон

- **background** Устанавливает все свойства фона элемента
- **background-color** Определяет фоновый цвет элемента
- **background-image** Фоновое изображение элемента
- **background-position** Положение фонового изображения
- **background-repeat** Повторение изображения фона
- **background-size** Размер фонового изображения
- **color** Цвет текста элемента

Внешние и внутренние отступы

- **margin** Размер всех отступов элемента
- **margin-bottom** Размер нижнего отступа элемента
- **margin-left** Размер левого отступа элемента
- **margin-right** Размер правого отступа элемента
- **margin-top** Размер верхнего отступа элемента
- **padding** Размер всех полей элемента за одну декларацию
- **padding-bottom** Размер нижнего поля элемента
- **padding-left** Размер левого поля элемента
- **padding-right** Размер правого поля элемента
- **padding-top** Размер верхнего поля элемента

- **width** Устанавливает ширину элементов
- **height** Устанавливает высоту элементов
- **max-height** Устанавливает максимальную высоту элемента
- **max-width** Устанавливает максимальную ширину элемента
- **min-height** Устанавливает минимальную высоту элемента
- **min-width** Устанавливает минимальную ширину элемента



- **border** Все атрибуты рамки
- **border-color** Цвет рамки
- **border-width** Толщина рамки
- **border-style** Стил ь рамки

Синтаксис PHP

- PHP-скрипт может быть размещен в любом месте документа.
- PHP-скрипт начинается с `<?php` и заканчивается `?>`:

```
<?php
```

```
// PHP код находится здесь
```

```
?>
```

PHP Комментарии

```
<?php
```

```
// Однострочный комментарий
```

```
# Это тоже однострочный комментарий
```

```
/*
```

```
Это многострочный блок комментариев  
охватывает несколько линий
```

```
*/
```

```
// Можно также использовать комментарии, чтобы исключить части строки кода
```

```
$x = 5 /* + 15 */ + 5;
```

```
echo $x;
```

```
?>
```


PHP Чувствительность к регистру

В PHP, все ключевые слова (например: if, else, while, echo и т.п.), классы и функции определенные пользователем не чувствительны к регистру.

```
<?php
```

```
    ECHO "Привет Мир!<br>";
```

```
    echo "Привет Мир!<br>";
```

```
    EcHo "Привет Мир!<br>";
```

```
?>
```

Переменные

- Переменные являются контейнерами для хранения информации.
- Все имена переменных чувствительны к регистру.
- В PHP, переменная начинается со знака \$, за которым следует имя переменной:

```
<?php
```

```
$txt = "Привет Мир!";
```

```
$x = 5;
```

```
$y = 10.5;
```

```
echo $txt;
```

```
?>
```

Переменные - правила

- Переменная начинается со знака \$, за которым следует имя переменной
- Имя переменной должно начинаться с буквы или символа подчеркивания
- Имя переменной не может начинаться с цифры
- Имя переменной может содержать только буквенно-цифровые символы и знак подчеркивания A-z, 0-9, и _
- Имена переменных чувствительны к регистру \$age и \$AGE, две разные переменные

PHP Вывод переменных

Следующий пример показывает, как вывести текст и переменную:

```
<?php
    $txt = "PHP";
    echo "Мне нравится $txt!";
?>
```

Следующий пример точно имеет, такой же результат, что и предыдущий пример:

```
<?php
    $txt = "PHP";
    echo "Мне нравится " . $txt . "!";
?>
```

Арифметические операторы

Оператор	Имя	Пример	Результат
+	Сложение	$\$x + \y	Сумма $\$x$ и $\$y$
-	Вычитание	$\$x - \y	Разница $\$x$ и $\$y$
*	Умножение	$\$x * \y	Результат $\$x$ и $\$y$
/	Деление	$\$x / \y	Коэффициент $\$x$ и $\$y$
%	Модуль	$\$x \% \y	Остаток при делении $\$x$ и $\$y$
**	Возведение в степень	$\$x ** \y	Результат $\$x$ к $\$y$ 'th право (введен в PHP 5.6)

Операторы присваивания

Присваивание	То же ...	Описание
$x = y$	$x = y$	Левый операнд получает значение выражения назначения справа.
$x += y$	$x = x + y$	Сложение
$x -= y$	$x = x - y$	Вычитание
$x *= y$	$x = x * y$	Умножение
$x /= y$	$x = x / y$	Деление
$x \% = y$	$x = x \% y$	Модуль

Типы данных

- string (строки) `$string = 'Строка';`
- integer (целые числа) `$integer = 10;`
- double (дробные числа) `$double = 10.15;`
- boolean (логический тип) `$boolean = true; // false`
- NULL `$null = null;`
- array (массивы) `$array = [10, 'Привет', 11.2, false, null];`

Условный оператор if, else, elseif

```
if ($a > $b) {  
    echo "a больше, чем b";  
} elseif ($a == $b) {  
    echo "a равен b";  
} else {  
    echo "a меньше, чем b";  
}
```


Операторы сравнения

Оператор	Имя	Пример	Результат
==	Равно	<code>\$x == \$y</code>	Возвращает true если <code>\$x</code> равно <code>\$y</code>
===	Идентичны	<code>\$x === \$y</code>	Возвращает true если <code>\$x</code> идентично <code>\$y</code> , и они того же типа
!=	Не равно	<code>\$x != \$y</code>	Возвращает true если <code>\$x</code> не равно <code>\$y</code>
<>	Не равно	<code>\$x <> \$y</code>	Возвращает true если <code>\$x</code> не равно <code>\$y</code>
!==	Не идентичны	<code>\$x !== \$y</code>	Возвращает true если <code>\$x</code> не идентичны <code>\$y</code> , или они не одного типа
>	Больше чем	<code>\$x > \$y</code>	Возвращает true если <code>\$x</code> больше чем <code>\$y</code>
<	Меньше чем	<code>\$x < \$y</code>	Возвращает true если <code>\$x</code> меньше чем <code>\$y</code>
>=	Больше чем или равно	<code>\$x >= \$y</code>	Возвращает true если <code>\$x</code> больше чем или равно <code>\$y</code>
<=	Меньше чем или равно	<code>\$x <= \$y</code>	Возвращает true если <code>\$x</code> меньше чем или равно <code>\$y</code>

Инкремент/Декремент операторы

RНР Инкремент оператор, используются для увеличения значения переменной.

RНР Декремент оператор, используются для уменьшения значения переменной.

Оператор	Имя	Описание
<code>++\$x</code>	Перед-инкрементом	Увеличивает <code>\$x</code> на один, возвращает <code>\$x</code>
<code>\$x++</code>	После-инкремента	Возвращает <code>\$x</code> , увеличивает <code>\$x</code> на один
<code>--\$x</code>	Перед-декрементом	Уменьшает <code>\$x</code> на один, возвращает <code>\$x</code>
<code>\$x--</code>	После-декремента	Возвращает <code>\$x</code> , уменьшает <code>\$x</code> на один

PHP пользовательские функции

- Функция - это блок операторов, которые могут использоваться повторно в программе.
- Функция не будет выполняться сразу при загрузке страницы.
- Функция будет выполняться при вызове функции.

```
<?php
    function имя_функции() {
        код для выполнения;
    }
?>
```

Вызов функции

```
<?php
    function writeMsg() {
        echo "Привет Мир!";
    }

    writeMsg(); // вызов функции
?>
```