БАЗОВЫЙ СИНТАКСИС

Используйте для отступов везде табуляцию размером 2. Это важно для читаемости кода, строк которого могут быть тысячи.  
Для улучшения читаемости ставьте пробел перед открывающей фигурной скобкой.  
Оставляйте пробел после двоеточия в каждом свойстве.  
Между каждым селектором оставляйте строку отступа.  
Каждое свойство на новой строке.  
В url() обязательно пишите кавычки.  
Обязательно использовать точку с запятой после каждого свойства.  
Если в свойстве где-то нужна запятая - после нее обязателен пробел.  
Однако в color: rgb(0,0,0) пробелы не нужны.  
Все 16-е значение (например цвета) пишутся в нижнем регистре, и где возможно, используются сокращения (#fff вместо #ffffff)  
Если нужен альфа-канал (прозрачность) - используется rgba.  
Если используете селектор по атрибуту - пишите двойные кавычки (input[type="checkbox"])  
Если используется нулевое значение (padding: 0) - единицы измерения не пишутся (кроме частных случаев поддержки браузеров).

/\* Правильно: \*/

.selector {

**padding**: 15px;

**margin**: 0 0 15px;

**background-color**: rgba(0, 0, 0, 0.5);

**box-shadow**: 0 1px 2px #ccc, inset 0 1px 0 #fff;

**top**: 0;

}

/\* Неправильно: \*/

.selector{

**padding**:15px;

**margin**:0px 0px 15px

background-color:rgba(0, 0, 0, 0.5);

**box-shadow**:0 1px 2px #CCC,inset 0 1px 0 #FFFFFF

top:0px;

СБРОС БРАУЗЕРНЫХ СТИЛЕЙ

Пользуйтесь сбросом стилей осторожно и осознанно. Это поможет избежать непредвиденных проблем со стилями по умолчанию. Рекомендуется делать базовый сброс box-sizing, а остальное делать только напрямую у элементов по классу.

/\* Пример минимального сброса стилей \*/

/\* Селектор \* - обращение ко всем элементам в документе \*/

\* {

**box-sizing**: border-box;

}

/\* Представим, что класс .title у заголовка h1, и нам нужно сбросить отступ сверху, а снизу поставить свой \*/

.title {

**margin**: 0;

**margin-bottom**: 30px;

}

ПОРЯДОК СВОЙСТВ

Соблюдайте определенный порядок стилей для удобства и читаемости кода. Сперва идут свойства позиционирования, затем блочной модели, потом - шрифты. В самом конце - прочее оформление и анимации.

.element {

  /\* Позиционирование \*/

  position: relative;

  left: 0;

  right: 0;

  top: 0;

  bottom: 0;

  /\* Блочная модель \*/

  display: flex;

  align-items: center;

  margin: 10px;

  padding: 10px 20px;

  border: 1px solid red;

  width: 200px;

  height: 100px;

  box-sizing: border-box;

  /\* Типографика \*/

  font-family: Arial;

  font-size: 25px;

  font-style: italic;

  text-decoration: none;

  color: red;

  /\* Оформление \*/

  background: red;

  opacity: 1;

  /\* Анимации \*/

  transform: translateX(5px);

  animation: shake 0.3s infinite;

}

ИМЕНОВАНИЕ КЛАССОВ

Запрещено использовать camelCase. Лучше .header или .header-btn. Использовать нижнее подчеркивание можно только при именовании по БЭМ.  
Если это кнопка - называть ее .knopka недопустимо. Используйте английский язык. Обязательно использовать осмысленные имена, чтобы было понятно, что это за элемент.

/\* Правильно: \*/

.header { ... }

.edit-button { ... }

.time { ... }

/\* Неправильно: \*/

.t { ... }

.ssylka { ... }

.asd { ... }

УПРОЩЕНИЕ СЕЛЕКТОРОВ

Старайтесь максимально упрощать селекторы\вложенность. Это влияет на скорость загрузки сайта и в целом на производительность.

/\* Хороший пример \*/

.nav { ... }

.list { ... }

/\* Плохой пример \*/

**nav**.nav { ... }

**ul**.list { ... }

ИСПОЛЬЗОВАНИЕ !IMPORTANT

Использовать !important можно только в САМЫХ крайних случаях. В остальном - категорически запрещено, так как вы потом не сможете переопределить свойство, если потребуется.

результат операции числовым, нужно использовать Number.isNaN.  
Запрещено использовать case без break или return в блоках switch.  
В коде не используется оставленных выводов в консоль.  
В коде нет объявленных, но неиспользуемых переменных.  
В коде нет выражений, значения которых не записываются в переменные, параметры функций или свойства объектов и не передаются как параметры функций.