Every Matrix

Everything is possible.

Everything is possible.

SASS Code Standards

sales@everymatrix.com

UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Why do we need SASS standards?

- Switching projects (when we need to, when we want to)
- Maximizing efficiency (finding stuff fast)
- Fast bug-fixing
- Fast extension
- Everybody understands what you're doing

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

What is CSS and SASS?

- CSS is a collection of:
  - SELECTORS,
  - RULES,
  - ...in a certain cascading ORDER,
  - ...with certain context OVERRIDES.
- SASS generates CSS. It automates and eases its use.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #1 “Are you yellin’ at me?”

There will be NO !important rules in our SASS. The only exceptions are fantastically rare cases of overwriting 3rd party content. Offenders will be beheaded.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #2 “No IDs, mo’ classes”

Never use IDs in CSS selectors, unless you absolutely have to, and the respective ID is certain to be singular in the project (not just in the page).

Instead, use classes and other, less imperative selectors.

Reason: easy overriding of rules, no dependency on legacy code.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #3 "Exact > generic"

Try to use as few element / generic selectors as possible. Use class selectors instead (where possible and sane).

E.g. Never use * (except in resets). Rarely use >, element.Class, +, etc.

Reason: generic selectors are slower to parse, class selectors are much faster.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #4 "The fewer, the better"

Do not use more than 2 selectors, unless absolutely necessary. Try to define generic elements with a single (class) selector, and all overrides with a second parent selector for context. 4 selectors and more are forbidden.

E.g. .Context1 .Button {...} .Context2 .Button {...}

Reason: less specificity, very easy overrides.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #5 "Indented 1liners FTW"

Place selectors and properties on a single line. Indent all lines according to the HTML structure they reference. Use tabs for indents.

E.g.

.ParentElement { all rules on 1 line }

.ChildElement { all rules on 1 line }

Reason: make all atoms / molecules / organisms easy to read. Exceptions: sub-selectors, mqs (#6).

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #6 “MQs are great kids”

Always nest Media Queries inside the element they’re modifying, when writing SASS. E.g.

```
.Element {...
        @media (min-width: 728px) {...}
}
```

Reason: modularization & easy tracking of changes for each element.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #7 “Content > devices”

When defining breakpoints, take into consideration the widget architecture and only change its display via Media Queries when relevant. Device-specific breakpoints are irrelevant, due to their volatile nature. Content & features (bandwidth, touch-enabled, etc) must dictate when to change its display.

Reason: atoms, molecules, organisms which can look good at any resolution. Future-proofing.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #8 “Order > chaos”

Properties should be followed by a colon and a space. They should maintain this order, to insure readability and standardization:

1. **BOX & STATUS MODIFIERS** (display, visibility, opacity, float, clear, content, list-style*, marker-offset, box-*, grid-*, zoom, etc),

2. **SIZE & ARRANGEMENT** (width, height, max-/min-*, margin*, padding*, position, top/right/bottom/left, clip, z-index, etc),

3. **COLOR & BG** (background*, color*, border*, outline*),

4. **TEXT** (font*, line-*, text-*, word-*, letter-*, white-space*, column-*, etc),

5. **SPECIAL EFFECTS** (animation*, transition*, transform*, perspective*, *-shadow, border-radius)

6. **OTHERS** (cue*, speak*, mark*, rest*, voice*, ruby*, fit*, page*, appearance, cursor, nav-*, etc)

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #9 “Semantic classes”

Use verbose classes, so anyone can understand the HTML structure and what the CSS styles semantically just by reading the CSS structure. Use ThisCase for all classes. Derive naming from BEM: BlockElementModifier.

E.g. .List {...} .ListItem {...} .ListItemSpecial {...} not .a {...} .b {...} .b.c {...}

Reason: optimizing search, cognition, standards.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #10 "No prefixes, no probs"

Avoid using vendor/browser prefixes (-moz-, -webkit-, -o-, -ms-, etc) in standard CSS. However, use SASS mixins to generate prefixes and avoid writing (and forgetting) them in standard code.

Reason: you might forget or misuse/miswrite prefixes, and mixins help with standardizing the prefixes used.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #11 "Mobile first"

Always start off with default basic mobile styles and add progressively enhanced media queries for different capabilities (e.g. increasing screen resolutions).

E.g.

```
.Element { styles for mobile basic
        @media (min-width: 320px) { styles for content reflow 1 }
        @media (min-width: 495px) { styles for content reflow 2 }
        ...
        @media (min-width: 1600px) { styles for large res content reflow }
}
```

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #12 “Say my name ,SASS”

In addition to Rule #6 “Semantic classes”, we should strive to achieve a semantic order in our SASS variables and mixin names. All similar variables and mixins should be named with a semantic prefix (namespacing), and order from generic to specific.

E.g. $color-blue, $color-blue-darker, fontSize(), etc.

Reason: it’s much easier to grasp the logic behind the advanced SASS features when they’re named properly.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #13 “Mark it up, dev!”

When building modules of any kind (atoms, molecules, organisms) always mark them up with a comment above. Indenting with Rule #4 will take care of the rest (so you needn’t comment below the module). If that doesn’t help, add a comment below as well. E.g.

```
/*
* @type module
* @name LoginForm
* @author Gelu Crocochiftelu
* @require Label, Input, Button
* @param $color-EveryMatrix
*/
$color-EveryMatrix: #00b74f !default;
.LoginForm {...}
        .LoginLabel {... color: $color-EveryMatrix; ...}
        .LoginInput {...}
        .LoginButton {...}
                .LoginButtonSymbol {...}
```

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #14 “Standards 4 life!”

- Classes shall be forevermore written like this:
  - .ClassNameOne
- Functions and mixins shall be written like this:
  - @mixin mixinNameTwo
- Variables shall be written like this:
  - $varNameThree
  - $nameSpace-varNameFour

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #15 “Swoosh!”

When animating elements in CSS, always animate only these properties: TRANSFORM (translate, scale, rotate) or OPACITY. When doing so, always create a compositing layer by specifying translateZ or translate3D. When animating, never create more than 3 layers. This means either animating 1 element and changing 3 properties, animating 2 elements one of which using 2 changing properties, or animating 3 elements changing one single property. The consequences of not respecting these rules is a FPS under 30. That is not acceptable. The minimum FPS we need to aim for is around 40, ideally 60, absolute minimum 24.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Rule #16 "\nei\na\na\na"

- Each front end developer needs to set his/her IDE to only produce **UTF-8** files.
- The IDEs must also be set to have **UNIX** line endings (LF, or /n or 0x0A). That's because our servers are Linux-based.
- All file upload / transfer tools, plus all GIT / SVN tools must be set to transfer files in a **binary** fashion.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Suggestion #1 “Pretty Sheets”

Ideally, the CSS/SASS spacing conventions would be:

```
.Parent { property1: value1; property2: value2; }

        .Child { property3: value3; property4: value4; }
```

Note that there are spaces before { , } , property*. There are no spaces before (just after) : , ; . Also note that the tab is 4 spaces long. All of you have large enough monitors, I doubt 4 spaces tabs is excessive, and it provides for excellent readability.

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Suggestion #2 “Make my day!”

Comment everything. As verbose as you can. Before atoms / molecules / organisms. For example, you can use comments for qualifying selectors.

E.g. instead of

html.no-csstransforms .selector {...} use

/*html*/.no-csstransforms .selector {...}

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix
Everything is possible.

Any suggestions?

- Color definition preference: #hex in standard cases, rgba() in opacity cases, hsl() in mixins.
- Quote styles: double.
- @extends at the top of the declaration list; @includes in context.
- Pseudo elements after main element (.A {...} .A:hover {...}).
- Prefer shorthands: background, border, font, margin, padding, etc.
- Use source maps!
- Lint your SASS!

sales@everymatrix.com
UK Office: +44 (0) 20 861 769 396

Every Matrix

Everything is possible.

SASS Structure & CEX integration

- [PROJECT] Meta
- Resets
- Generic Fonts & Icon Fonts
- Generic Language Variations
- Generic SASS Variables
- Tools: mixins, clearfix, hiding elements, box-sizing
- Generic Print Styles
- Generic Shame & Lame: IE, Safari, Opera, etc.
- [PROJECT] Fonts & Icon Fonts
- [PROJECT] Language Variations
- [PROJECT] SASS Variables
- General Styling: Anchors, Headings, Buttons, etc.
- Modules: Atoms, Molecules, Organisms, Pages
- [PROJECT] Print Styles
- [PROJECT] Shame & Lame: IE, Safari, Opera, etc.

sales@everymatrix.com

UK Office: +44 (0) 20 861 769 396