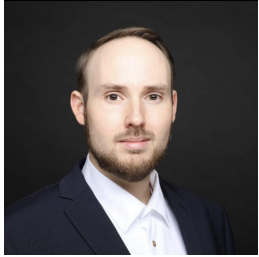


CLEAN CSS

Comprehensible, maintainable and scalable CSS



C'EST MOI



Andreas Gartz
Product Owner
IT-Projektleiter



constructiva
SOLUTIONS



A SHORT STORY

```
/* Idea: increase space below paragraphs */
```

A SHORT STORY

```
/* Idea: increase space below paragraphs */  
p { margin-bottom: 2em; }
```

A SHORT STORY

```
.paragraph p, .paragraph,  
.lead p, .lead,  
.header > .copy,  
.category .widget > p,  
.category .widget .copy { margin-bottom: 2em; }
```

A SHORT STORY

```
/* fixes.css (must be included as last file) */  
.paragraph p, .paragraph,  
.lead p, .lead,  
.header > .copy,  
.category .widget > p,  
.category .widget .copy { margin-bottom: 2em; }
```

A SHORT STORY

```
/* Late findings: */  
#home .hero p { margin-bottom: 1.75em; }  
#form .description { margin-bottom: 1.2em; }
```

A SHORT STORY

```
/*  
 *   new: page-style.css  
 *   set default margin-bottom for paragraph elements on all pages  
 *   Specificity: 1 1 1  
 */  
  
#body .page p,  
#body .page .paragraph { margin-bottom: var(--mb); }
```


A SHORT STORY

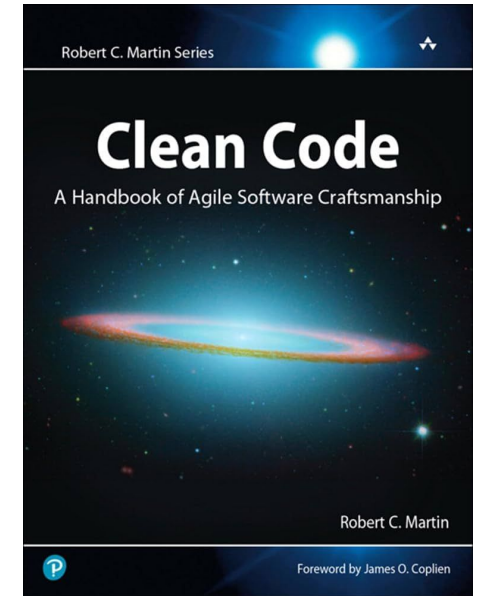
```
/*
 *    new: page-style.css
 *    set default margin-bottom for paragraph elements on all pages
 *    Specificity: 1 1 1
 */
#body .page p,
#body .page .paragraph { margin-bottom: var(--mb); }

/* THE WAR IS ON! */
#body .side-bar > p { margin-bottom: 22px; }
```

A REAL-LIFE EXAMPLE

```
.wrapper_landing .l_header .btns_group .v-menu {  
    display: inline-block;  
}  
.wrapper_landing .l_header .btns_group .v-menu .v-menu__content {  
    max-width: 100%;  
}  
.wrapper_landing .l_header .btns_group .v-menu .v-alert {  
    margin-top: 10px;  
    margin-bottom: 0;  
}  
.wrapper_landing .l_header .btns_group .v-menu .v-btn--round .v-btn__content .v-icon {  
    color: var(--red);  
}  
@media (max-width: 800px) {  
    .wrapper_landing .l_header .v-menu .btns_group { padding: 10px; }  
}
```

CLEAN CODE



CLEAN CODE

“rude”

```
public static String testableHtml(  
    PageData pageData,  
    boolean includeSuiteSetup  
) throws Exception {  
    WikiPage wikiPage = pageData.getWikiPage();  
    StringBuffer buffer = new StringBuffer();  
    if (pageData.hasAttribute("Test")) {  
        if (includeSuiteSetup) {  
            WikiPage suiteSetup =  
                PageCrawlerImpl.getInheritedPage(  
                    SuiteResponder.SUITE_SETUP_NAME, wikiPage  
                );  
            if (suiteSetup != null) {  
                WikiPagePath pagePath =  
                    suiteSetup.getPageCrawler().getFullPath(suiteSetup);  
                String pagePathName = PathParser.render(pagePath);  
                buffer.append("!include -setup .")  
                    .append(pagePathName)  
                    .append("\n");  
            }  
        }  
        WikiPage setup =  
            PageCrawlerImpl.getInheritedPage("SetUp", wikiPage);  
    }  
}
```

4

Source: UnityCoin "Clean Code - Uncle Bob / Lesson 1" on Youtube.com
(<https://youtu.be/7EmboKQH8IM?si=ASanQXUtZXWylrb1>)

*It's **rude to** get it working
and then do **not fix it!***

– “Uncle Bob”

*It's **polite** to allow the
reader **to exit early**.*

– “Uncle Bob”

CLEAN CODE

“rude”

```
public static String testableHtml(  
    PageData pageData,  
    boolean includeSuiteSetup  
) throws Exception {  
    WikiPage wikiPage = pageData.getWikiPage();  
    StringBuffer buffer = new StringBuffer();  
    if (pageData.hasAttribute("Test")) {  
        if (includeSuiteSetup) {  
            WikiPage suiteSetup =  
                PageCrawlerImpl.getInheritedPage(  
                    SuiteResponder.SUITE_SETUP_NAME, wikiPage  
                );  
            if (suiteSetup != null) {  
                WikiPagePath pagePath =  
                    suiteSetup.getPageCrawler().getFullPath(suiteSetup);  
                String pagePathName = PathParser.render(pagePath);  
                buffer.append("!include -setup .")  
                    .append(pagePathName)  
                    .append("\n");  
            }  
        }  
        WikiPage setup =  
            PageCrawlerImpl.getInheritedPage("SetUp", wikiPage);
```

4

SMALLER THAN A SCREENFUL

- Functions should not be 100 lines long.
- Functions should hardly ever be 20 lines long.
- Indeed, the refactored function was too long.
- It should have been:

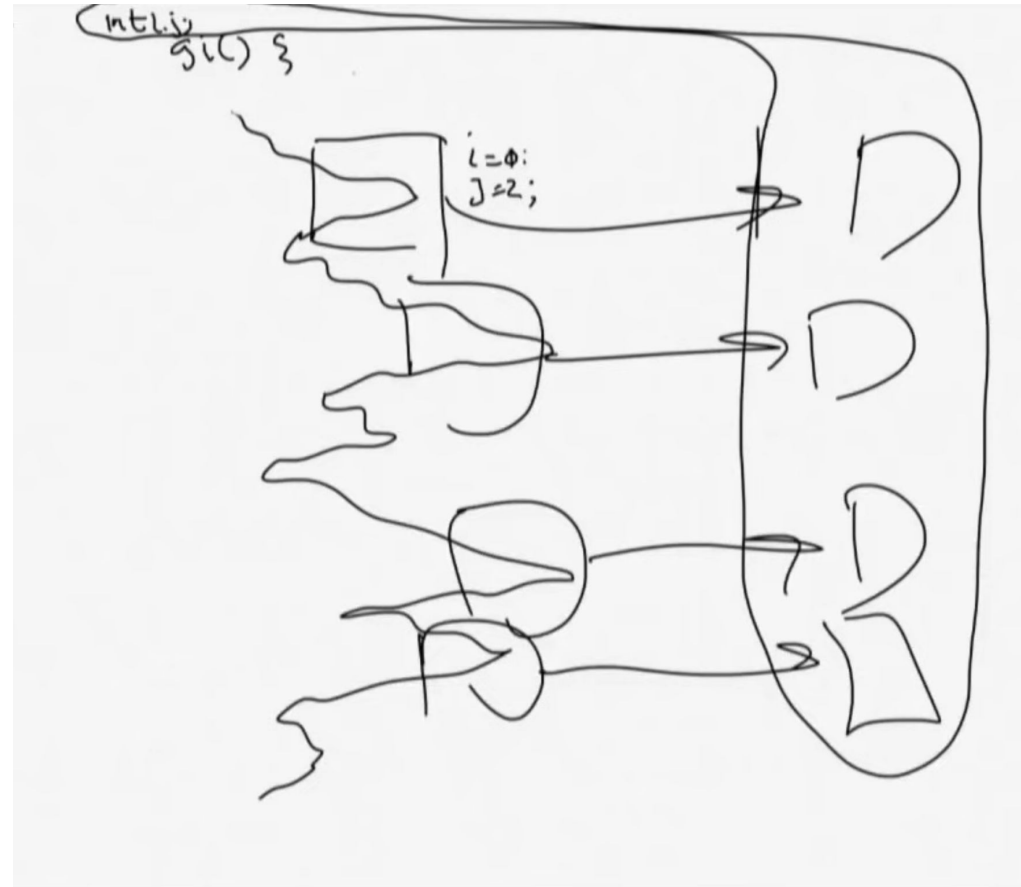
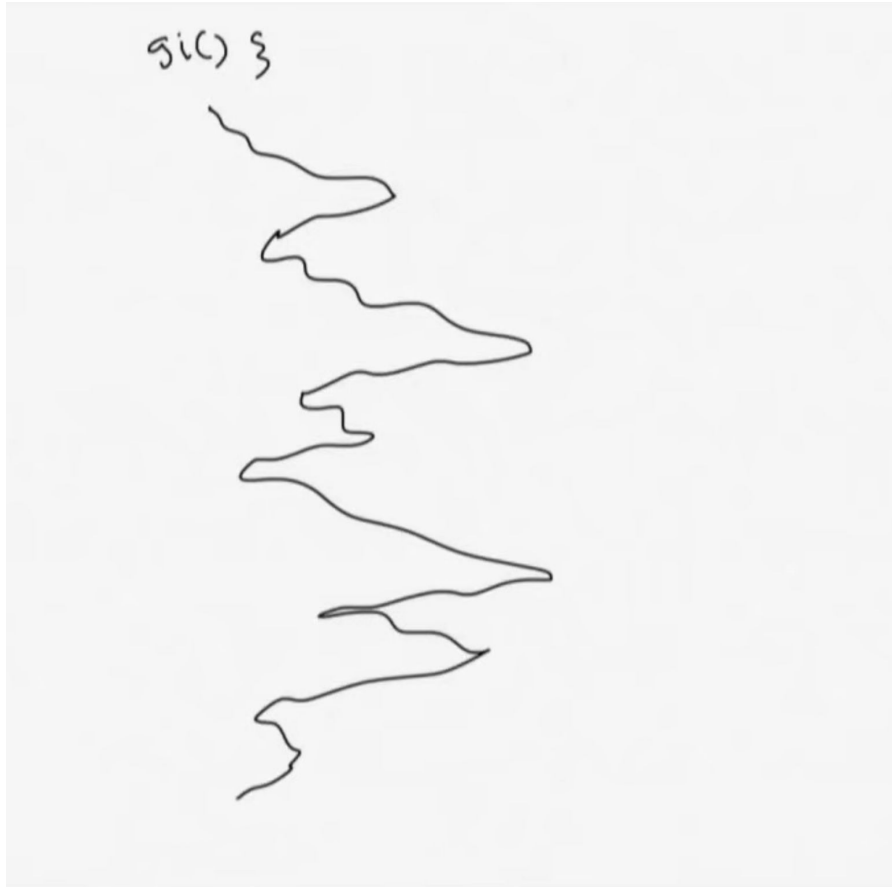
```
public static String renderPageWithSetupsAndTeardowns(  
    PageData pageData, boolean isSuite) throws Exception {  
    if (isTestPage(pageData))  
        includeSetupAndTeardownPages(pageData, isSuite);  
    return pageData.getHtml();  
}
```

“polite”

16

Source: UnityCoin “Clean Code - Uncle Bob / Lesson 1” on Youtube.com
(<https://youtu.be/7EmboKQH8IM?si=ASanQXUtZXWylrb1>)

CLEAN CODE



Source: UnityCoin "Clean Code - Uncle Bob / Lesson 1" on Youtube.com
(<https://youtu.be/7EmboKQH8IM?si=ASanQXUtZXWylrb1>)

CLEAN CSS

A REAL-LIFE EXAMPLE

```
.wrapper_landing .l_header .btns_group .v-menu {  
    display: inline-block;  
}  
.wrapper_landing .l_header .btns_group .v-menu .v-menu__content {  
    max-width: 100%;  
}  
.wrapper_landing .l_header .btns_group .v-menu .v-alert {  
    margin-top: 10px;  
    margin-bottom: 0;  
}  
.wrapper_landing .l_header .btns_group .v-menu .v-btn--round .v-btn__content .v-icon {  
    color: var(--red);  
}  
@media (max-width: 800px) {  
    .wrapper_landing .l_header .v-menu .btns_group { padding: 10px; }  
}
```

A REAL-LIFE EXAMPLE

```
.wrapper_landing {  
  .l_header {  
    .btns_group {  
      .v-menu {  
        display: inline-block;  
      }  
    }  
  }  
}  
  
.wrapper_landing {  
  .l_header {  
    .btns_group {  
      .v-menu {  
        .v-menu__content {  
          max-width: 100%;  
        }  
      }  
    }  
  }  
}
```

REDUCE SELECTOR LEVELS

```
.wrapper_landing {  
    ...  
}  
  
.v-menu {  
    display: inline-block;  
}  
  
.v-menu__content {  
    max-width: 100%;  
}
```

REDUCE SELECTOR LEVELS

```
.wrapper_landing {  
    ...  
}  
  
.wrapper_landing .v-menu {  
    display: inline-block;  
}  
  
.wrapper_landing .v-menu__content {  
    max-width: 100%;  
}
```

MEANINGFUL NAMES

```
/* microsite.css */

.microsite {
    ...
}

.microsite__nav {
    display: inline-block;
}

.microsite__nav-content {
    max-width: 100%;
}
```

MEANINGFUL NAMES

```
/* microsite.css
 * specific styles for the marketing microsite */
.microsite {
    ...
}

.microsite__nav {
    display: inline-block;
}

.microsite__nav-content {
    max-width: 100%;
}
```

MEANINGFUL NAMES

```
/* microsite.css
 * specific styles for the marketing microsite */
.microsite {
    ...
}

.microsite__nav {
    display: inline-block;
}

.microsite__nav-content {
    max-width: 100%;
}
```



A SHORT SELECTOR IS A GOOD SELECTOR

Keep selector combinations
within 2 class-based levels!
Avoid element selectors.



CSS METHODOLOGIES

Choose and enforce a **common methodology**.

Build your own or use:

1. BEM Block__Element--Modifier
2. Cube CSS Composition Utility Block Exception CSS
3. Atomic Design
4. SMACSS Scalable and Modular Architecture for CSS
5. ...



REAL-WORLD EXAMPLE

```
<div class="mdl lg stack">  
  
  <h1 class="ttl cntr huge">Sign into your account</h1>  
  
  <div class="job mdl cnt">  
    <form class="login frm ">...</form>  
  </div>  
</div>
```

BEM EXAMPLE

```
<div class="modal modal--large">  
  
  <h1 class="modal__title">Sign into your account</h1>  
  
  <div class="modal__content">  
    <form class="form-login">...</form>  
  </div>  
</div>
```

BEM EXAMPLE

```
<div class="modal modal--large"> /* block "modal", modifier "large" */  
  
  <h1 class="modal__title">Sign into your account</h1> /* child element */  
  
  <div class="modal__content"> /* child element */  
    <form class="form-login">...</form> /* Block "form-login" */  
  </div>  
</div>
```

BEM EXAMPLE

```
<div class="modal modal--large">  
  
  <h1 class="modal__title">Sign into your account</h1>  
  
  <div class="modal__content">  
    <form class="form-login">...</form>  
  </div>  
</div>
```



MAKE USE OF MEANINGFUL NAMES

For classes, variables, files & mixins.

Use prefixes to reduce selector combinations!



POLITE VARIABLE NAMES

```
:root {  
    --color: black;  
    --border: 1px solid red;  
}
```

```
/* home.css */  
.home {  
    --border: red;  
}
```


POLITE VARIABLE NAMES

```
:root {  
    --color: black;  
    --border: 1px solid red;  
}
```

```
/* home.css */  
.home {  
    --border: red;  
}
```

```
:root {  
    --root-color-text: black;  
    --root-border-width: 1px;  
    --root-border-style: solid;  
    --root-border-color: red;  
}
```

```
/* much better @ home.css: */  
.home {  
    --root-border-color: red;  
}
```

POLITE VARIABLE NAMES

```
.job {  
  --job-text-color: darkgray;  
  --job-border-color: gray;  
  
  color: var(--job-text-color);  
  border: 1px solid var(--job-border-color);  
}
```

POLITE VARIABLE NAMES

```
.job {  
    --job-text-color: darkgray;  
    --job-border-color: gray;  
  
    color: var(--job-text-color);  
    border: 1px solid var(--job-border-color);  
}  
  
.job--inactive      { --job-text-color: lightgray; }  
.job--highlighted   { --job-border-color: orange; }
```

POLITE VARIABLE NAMES

```
.job {  
  --job-text-color: darkgray;  
  --job-border-color: gray;  
  
  color: var(--job-text-color);  
  border: 1px solid var(--job-border-color);  
}
```

```
.job--inactive      { --job-text-color: lightgray; }  
.job--highlighted   { --job-border-color: orange; }
```



ANOTHER EXAMPLE

```
.content {  
  display: grid;  
  grid-template-columns:  
    [full-width-start] minmax(1.5rem, 1fr) [breakout-start]  
    minmax(0, calc(1440px - 1100px) / 2)) [main-start]  
    min(100% - (1.5rem * 2), 1440px)  
    [main-end] minmax(0, calc(1440px - 1100px) / 2))  
    [breakout-end] minmax(1.5rem, 1fr) [full-width-end];  
}
```

COMPREHENSIBLE

```
/* Layout: 1 Column with breakout
 *  |- padding | breakout-col | main-col | breakout-col | padding -|
 */
.layout-column--breakout {
  display: grid;
  grid-template-columns:
    [full-width-start] minmax(1.5rem, 1fr) [breakout-start]
    minmax(0, calc(1440px - 1100px) / 2)) [main-start]
    min(100% - (1.5rem * 2), 1440px)
    [main-end] minmax(0, calc(1440px - 1100px) / 2))
    [breakout-end] minmax(1.5rem, 1fr) [full-width-end];
}
```

ENCAPSULATED COMPLEXITY, NO REDUNDANCY

```
/*  Layout: 1 Column
*   |- padding | main-col | padding -|
*/
.layout-column {
    @include create_one_column_grid_with_breakout(1.5rem, 1440px);
}

/*  Layout: 1 Column with breakouts
*   |- padding | breakout-col | main-col | breakout-col | padding -|
*/
.layout-column--breakout {
    @include create_one_column_grid_with_breakout(1.5rem, 1440px, 1100px);
}
```

ENCAPSULATED COMPLEXITY, NO REDUNDANCY

```
@mixin create_one_column_grid_with_breakout (  
  $padding-inline: 1rem,  
  $main-max-width: 1100px,  
  $breakout-max-width: $main-max-width  
) {  
  display: grid;  
  $breakout-size: calc($breakout-max-width - $main-max-width) / 2);  
  grid-template-columns:  
    [full-width-start] minmax($padding-inline, 1fr) [breakout-start]  
    minmax(0, $breakout-size) [main-start]  
    min(100% - ($padding-inline * 2), $main-max-width)  
    [main-end] minmax(0, $breakout-size)  
    [breakout-end] minmax($padding-inline, 1fr) [full-width-end];  
}
```


COMPREHENSIBLE, NO REDUNDANCY

```
/*  Layout: 1 Column
*   |- padding | main-col | padding -| */
.layout-column {
    --padding-inline: 1.5rem;
    --main-max-width: 1100px;
    --breakout-max-width: --main-max-width;

    display: grid;
    --breakout-size: calc((var(--breakout-max-width) - var(--main-max-width)) / 2);
    grid-template-columns: ...
}

/*  Layout: 1 Column with breakouts
*   |- padding | breakout-col | main-col | breakout-col | padding -| */
.layout-column.has-breakout { --breakout-max-width: 1440px; }
```

AVOID REDUNDANCY

DRY: Don't repeat yourself.



AVOID REDUNDANCY

```
/* job-section.css */  
  
.job-section {}  
.job-section__wrapper {}  
.job-section__container {}  
.job-section__list {}  
.job-section__list-label {}  
.job-section__list-item {}  
.job-section__job {}  
.job-section__job-title {}  
.job-section__job-description {}
```

```
/* benefits.css */  
  
.benefits {}  
.benefits__wrapper {}  
.benefits__container {}  
.benefits__list {}  
.benefits__list-label {}  
.benefits__list-item {}  
.benefits__benefit-title {}  
.benefits__benefit-icon {}
```

AVOID REDUNDANCY

```
/* job-section.css */

.job-section {} | LAYOUT
.job-section__wrapper {} |
.job-section__container {} |
.job-section__list {} | LISTING
.job-section__list-label {} |
.job-section__list-item {} |
.job-section__job {} | JOB
.job-section__job-title {} |
.job-section__job-description {} |
```

AVOID REDUNDANCY

```
/* section.css */  
  
.section {}  
.section__wrapper {}  
.section__container {}
```

```
/* popping-list.css */  
  
.popping-list {}  
.popping-list__content {}  
.popping-list__item {}
```

```
/* job.css */  
  
.job {}  
.job__title {}  
.job__description {}
```

```
/* benefit.css */  
  
.benefit {}  
.benefits__title {}  
.benefits__icon {}
```

AVOID REDUNDANCY

```
/* section.css */  
  
.section {}  
.section__wrapper {}  
.section__container {}
```

```
/* popping-list.css */  
  
.popping-list {}  
.popping-list__content {}  
.popping-list__item {}
```

```
/* job.css */  
  
.job {}  
.job__title {}  
.job__description {}
```

```
/* benefit.css */  
  
.benefit {}  
.benefits__title {}  
.benefits__icon {}
```



REVIEW AND REFACTOR CONSTANTLY

Make use of common standards &
CSS guidelines in your team.



CLEAN CSS

Comprehensible, maintainable and scalable CSS



CLEAN CODE

Excerpt of helpful CLEAN CODE principles:

1. **Keep selectors small** *no war on specificity!*
2. **Use meaningful names** *to understand context and scope*
3. **Avoid redundancy** *DRY-principle*
4. **Review & refactor constantly** *use CSS guidelines in your team*
5. ...

LINKS

1. Nicolas Gallagher's Idiomatic CSS principles

<https://github.com/necolas/idiomatic-css>

2. Harry Roberts's CSS guidelines

<http://cssguidelin.es/>

3. BEM

<https://getbem.com/>

4. Cube CSS

<https://cube.fyi/>



CONTACT



Andreas Gartz
Projektleiter / Product Owner
a.gartz@constructiva.de

www.constructiva.de



vcard

THANK YOU!

I'm ready for questions!



