

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with faint, lighter blue diagonal stripes.

Sass - a better css



# Table of contents

1. Sass - What and why
2. Sass or Scss
3. Features
4. Do it yourself



# 1. Sass - Why and What

What is Sass:

- A css preprocessor
- Compiles sass and scss syntax into css syntax

Why you should use Sass:

- Maintainability and readability of your css code.



## 2. Sass vs Scss (syntax)

Sass:

- indentation instead of brackets
- newlines instead of semicolons

Scss:

- new version of the sass syntax (but both have the same features)
- stays true to css basic syntax



## 3. Features

- **Variables**

- Define and reuse variables in your code with the \$ sign

Scss:

```
$primary-color: #3bbfce;
$margin: 16px;

.content-navigation {
  border-color: $primary-color;
  color: darken($primary-color, 10%);
}

.border {
  padding: $margin / 2;
  margin: $margin / 2;
  border-color: $primary-color;
}
```

Compiles to:

```
.content-navigation {
  border-color: #3bbfce;
  color: #2b9eab;
}

.border {
  padding: 8px;
  margin: 8px;
  border-color: #3bbfce;
}
```



## 3. Features

- **Nesting**

- assign style rules to all child elements in one go

Scss:

```
table.hl {  
  margin: 2em 0;  
  td.ln {  
    text-align: right;  
  }  
}  
  
li {  
  font: {  
    family: serif;  
    weight: bold;  
    size: 1.3em;  
  }  
}
```

Compiles to:

```
table.hl {  
  margin: 2em 0;  
}  
table.hl td.ln {  
  text-align: right;  
}  
  
li {  
  font-family: serif;  
  font-weight: bold;  
  font-size: 1.3em;  
}
```



## 3. Features

- **Mixins**

- reuse parts of your code as will

Scss:

```
@mixin table-base {  
  th {  
    text-align: center;  
    font-weight: bold;  
  }  
  td, th {  
    padding: 2px;  
  }  
}  
  
#data {  
  @include table-base;  
}
```

Compiles to:

```
#data th {  
  text-align: center;  
  font-weight: bold;  
}  
#data td, #data th {  
  padding: 2px;  
}
```



## 3. Features

- **Mixins**

- reuse parts of you code at will

Mixin with parameter:

```
@mixin transform($property) {  
  -webkit-transform: $property;  
  -ms-transform: $property;  
  transform: $property;  
}  
  
.box { @include transform(rotate(30deg)); }
```





## 4. Do it yourself

Basic test:

1. Open index.html in your browser
2. Run compile\_scss.bat
3. Reload index.html in your browser - your font should be red now!

Test the other features:

- Try nesting some style rules
- Create a mixin to rotate an html element and use it on multiple objects with different degrees of rotations
- Experiment, clean up an old css file with scss

Check <https://sass-lang.com/guide> and ask me for help!