**Official offset ---🡪 Sass-lang.com**

**1) Explain what is Sass? How it can be used?**

Sass stands for Syntactically Awesome Stylesheets and was created by Hampton Catlin. It is an extension of CSS3, adding nested rules, mixins, variables, selector inheritance, placeholder, extend, import etc.

Sass can be used in three ways

* As a command line tool
* As a standalone Ruby module
* As a plugin for any Rack-enabled framework

**2) List out the key features for Sass?**

Key features for Sass include

* Full CSS3-compatible
* Language extensions such as nesting, variables, placeholder, extend, import and mixins
* Many useful functions for manipulating colors and other values
* Advanced features like control directives for libraries
* Well-formatted, customizable output

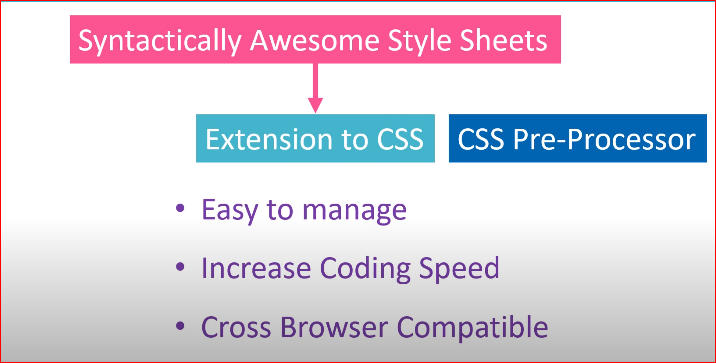
**3) What are the reasons behind using SASS?**

Following are some important reasons behind the popularity of SASS.

* You can write codes easily and efficiently, and they are easy to maintain.
* It is a pre-processing language which provides its syntax for CSS.
* It is a superset of CSS which contains all the features of CSS and is an open source pre-processor, coded in Ruby.
* It is more stable and powerful CSS extension and style documents more clearly and structurally.
* It facilitates reusability methods, logic statements and some of the built-in functions like color manipulation, mathematics, and parameter lists item.
* It is more stable, powerful and fully compatible to CSS3.
* It is time-saving because it facilitates you to write CSS in less code.
* It uses its syntax.
* It is based on the JavaScript and superset of CSS.
* It is an Open source pre-processor that interprets into CSS.
* It contains various functions for manipulating colors and other values.
* It has advanced control directives for libraries.
* It provides well formatted, customizable output.

**4) What are the advantages of Sass?**

* Time saving.
* More efficient and quicker.
* Compatible with all versions of CSS.
* You can use nested syntax and useful functions such as color manipulation, mathematics and other values.
* Write clean CSS in programming construct
* It is the super set of the CSS and using nested and others value.

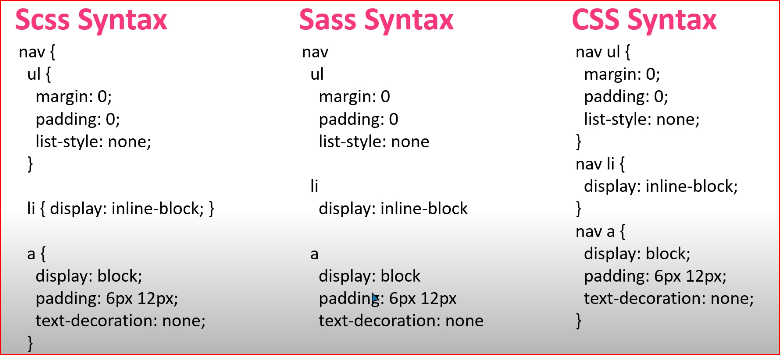




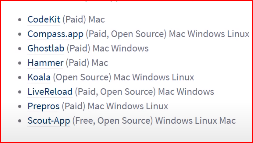
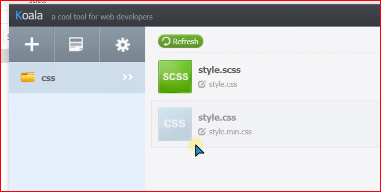
**5) Explain what is the difference between Sass and SCSS?**

The difference between Sass and SCSS is that,

* Sass is a CSS pre-processor with syntax advancements and an extension of CSS3
* Sass has two syntax
* The first syntax is “SCSS” and it uses the .scss extension
* The other syntax is indented syntax or just “Sass” and it uses the .sass extension
* While Sass has loose syntax with white space and no semicolons, the SCSS resembles more to CSS

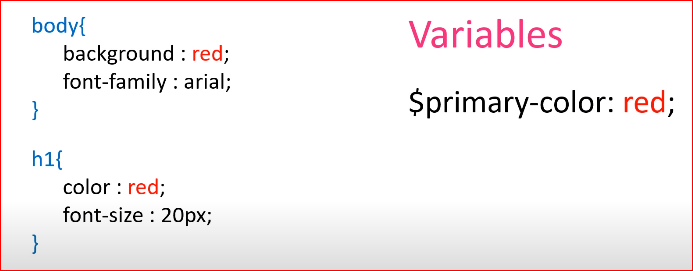


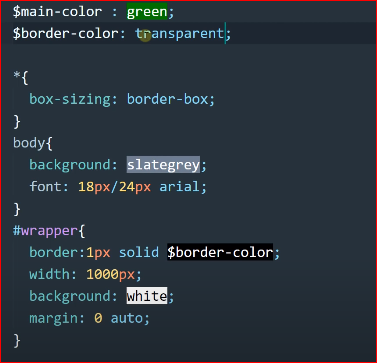


**6) Which variable is used to define SASS?**

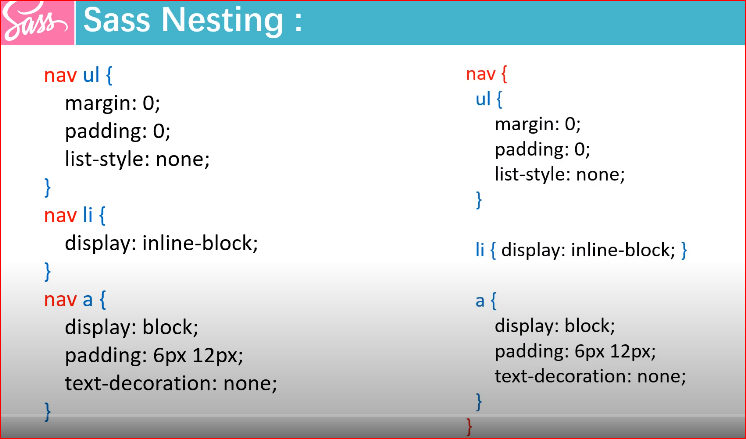
A variable begins with a dollar ($) sign, and the assignment of the variable is completed with a semicolon (;) sign.

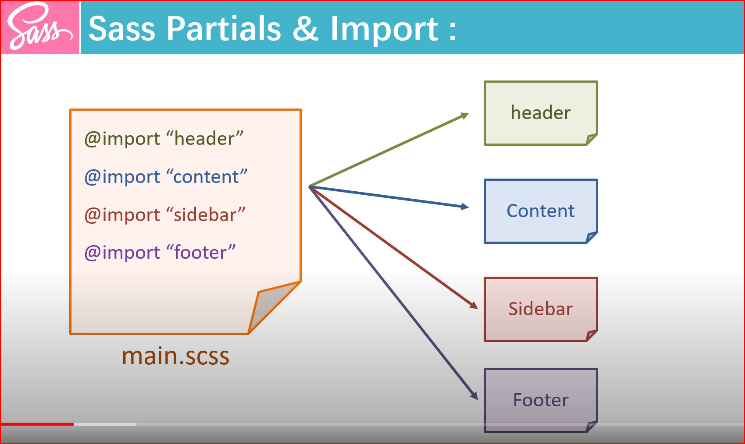




**7) What are nested rules in Sass?**

Nesting is a method of combining multiple logic structures within one another. In Sass, various CSS rules are connected to one another.



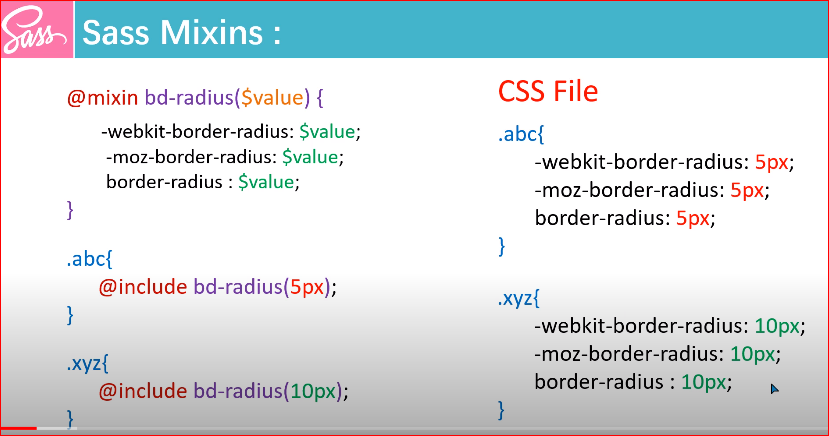




All part like nav, footer, or pages file are separate then we get benefit when we modify those code

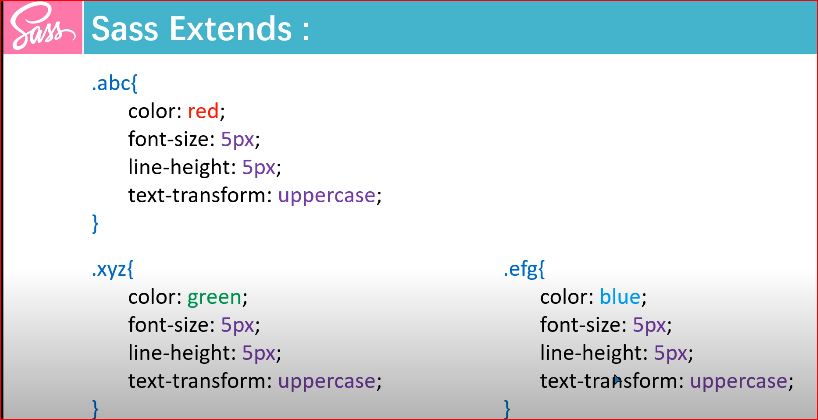
**8) Define the use of Sass Mixin function?**

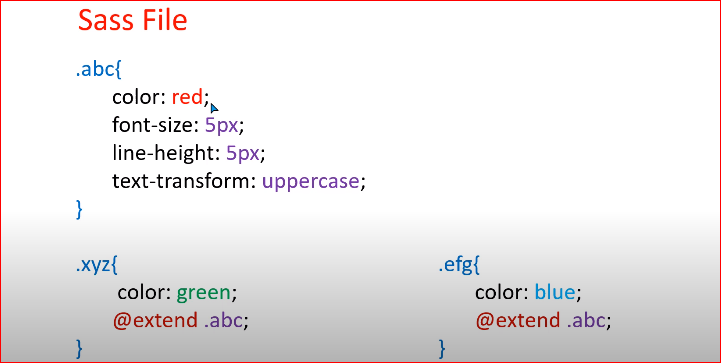
The Mixin function is used to define styles. Functions and Mixins are very similar. You can re-use this style throughout the style sheet. To re-use it you do not need to resort the non-semantic classes like .float-left. The Mixin can store multiple values or parameters and call a function to avoid writing repetitive codes. It names can use underscores and hyphens interchangeably.



**9) What is the use of @extend directive in SASS?**

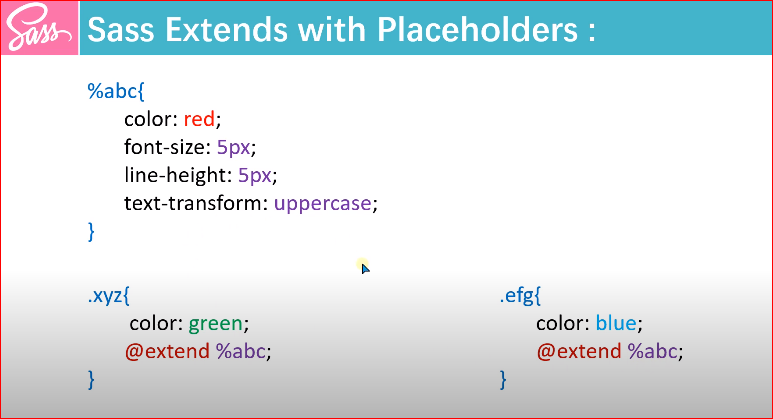
The SASS @extend directive is used to share a set of CSS properties from one selector to another. It is a very important and useful feature of Sass. It allows classes to share a set of properties with one another. It makes your code less and facilitates you to rewrite it repeatedly.

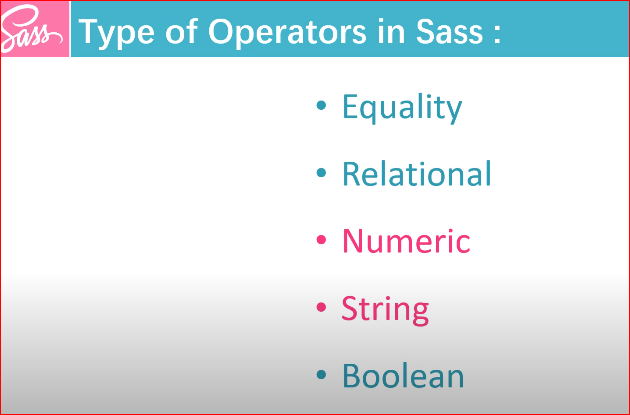




**10) What is the way to write a placeholder selector in Sass?**

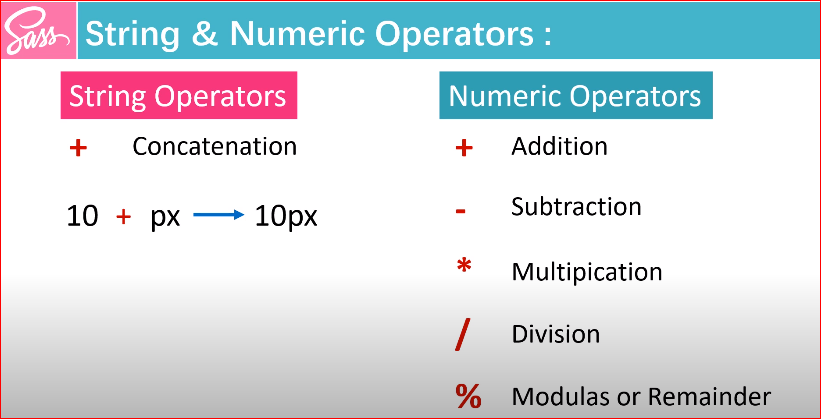
In Sass, the placeholder selectors can be used with class or id selector. In standard CSS, these are specified with "#" or ".", but in SASS they are replaced with "%". It uses @extend directive to display the result in CSS.

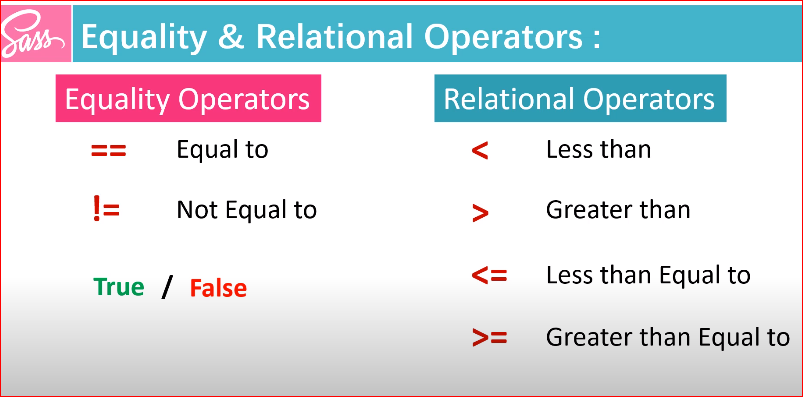




**11) What are number operations in Sass?**

In Sass, the number operations are used for mathematical operations like addition, subtraction, multiplication and division. The Sass number operation will do something like take pixel values and convert them to percentages without much hassle.

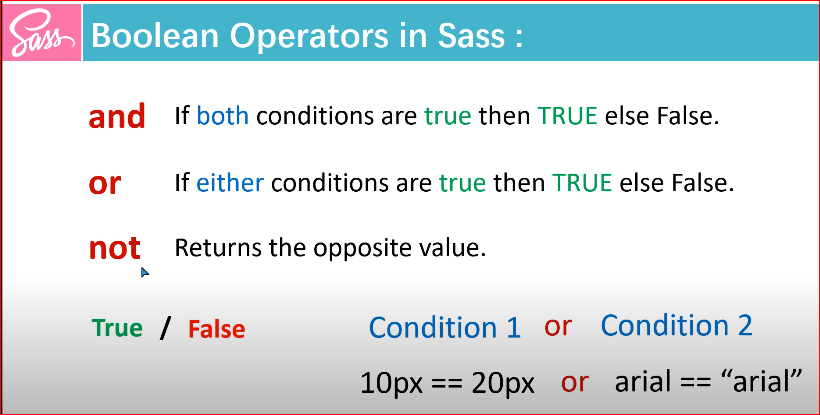


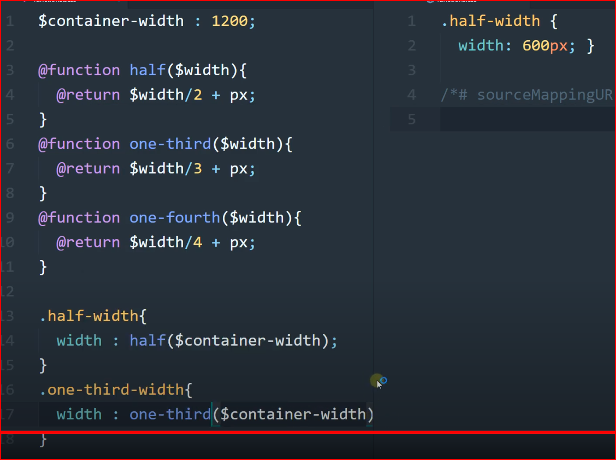


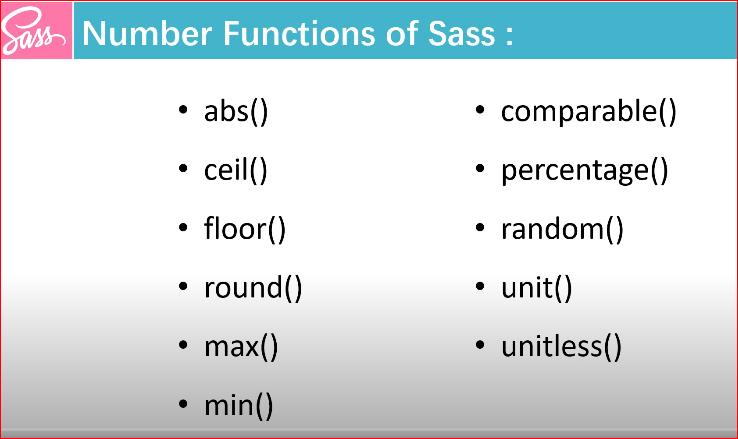


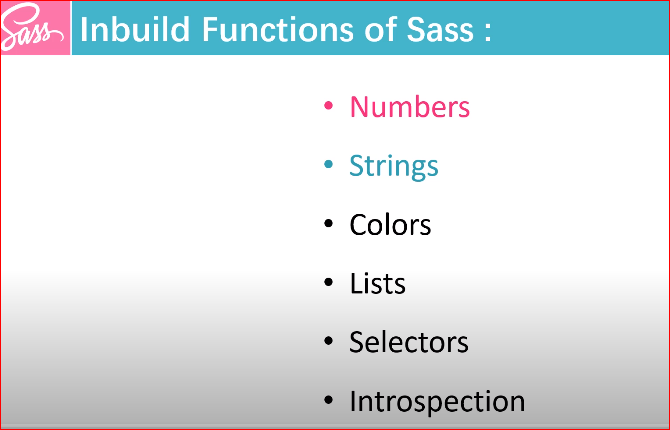
**12) How can we perform Boolean operations in Sass?**

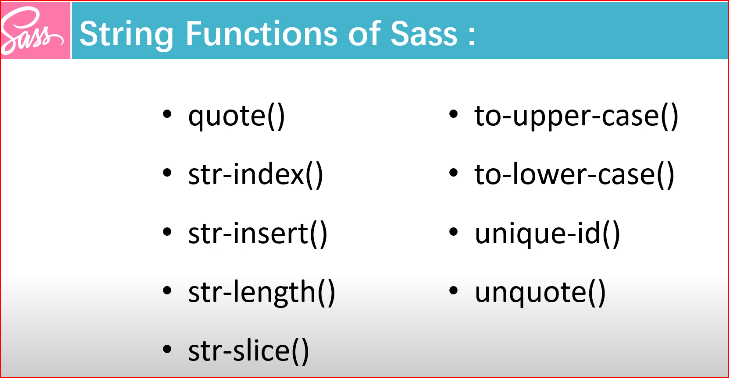
The Boolean operations can be performed on Sass script by using and, & and not operators.

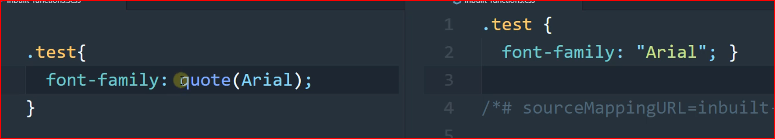






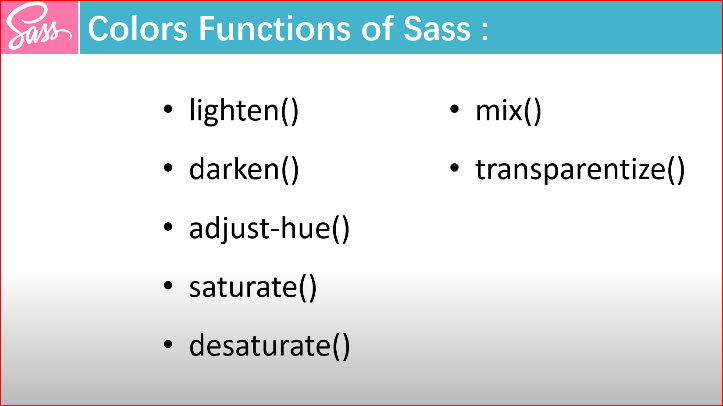


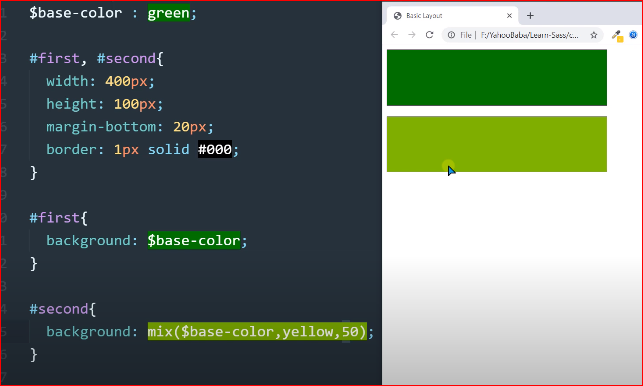


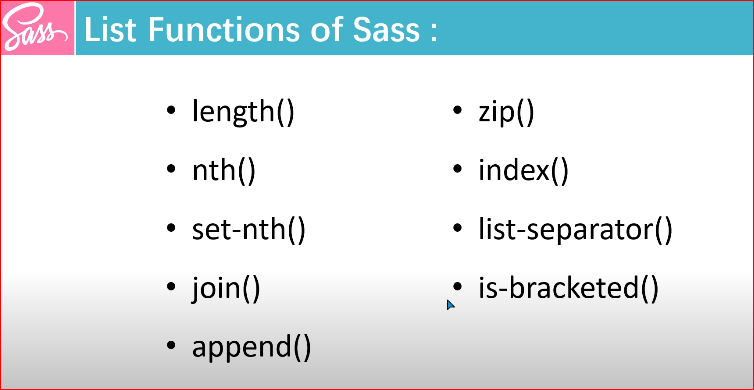


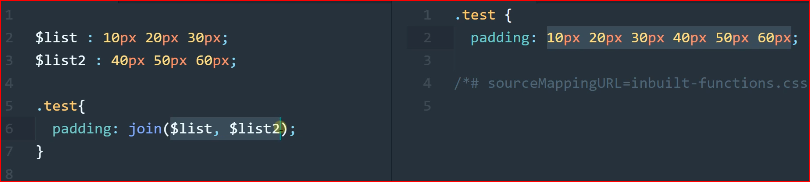
**13) What are the color operations in Sass?**

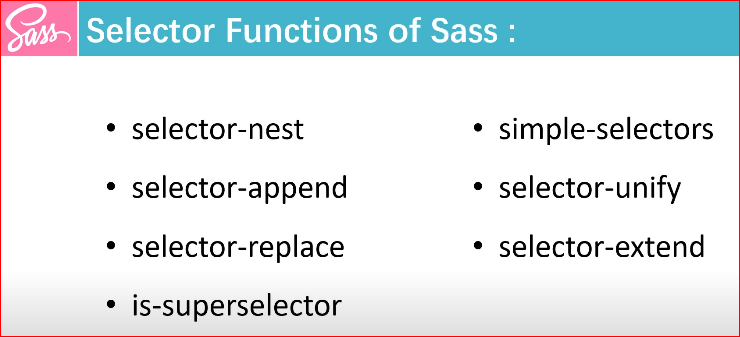
In Sass, color operation allows to use color In Sass, color operation allows to use color components along with arithmetic operations

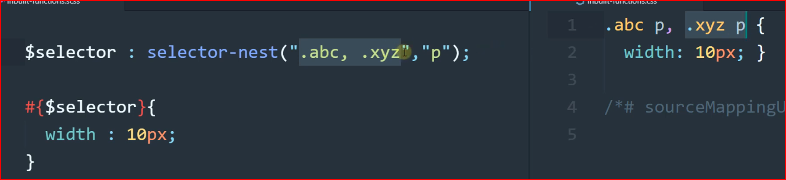


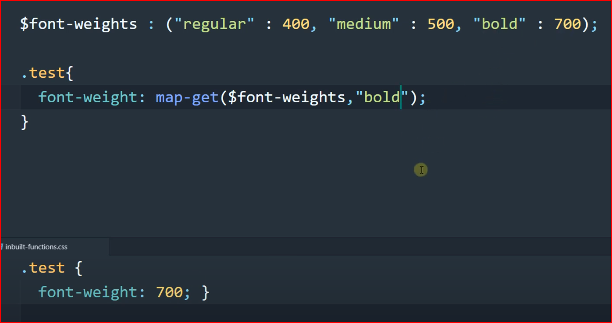


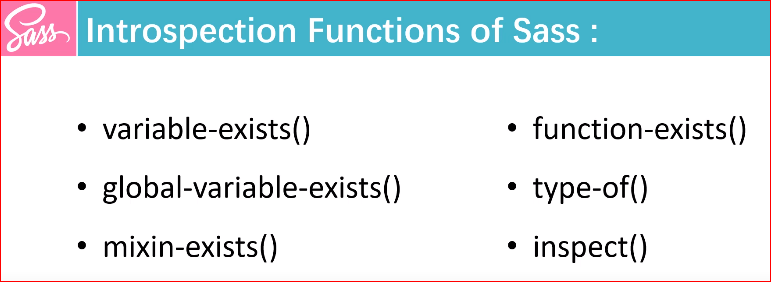


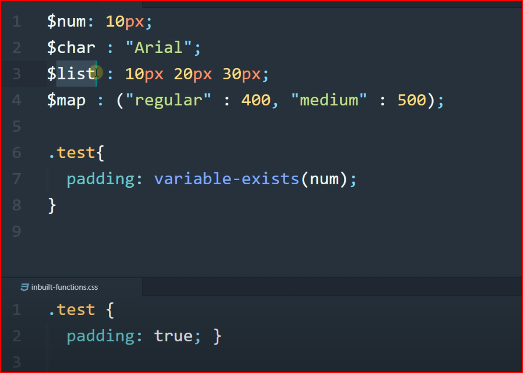


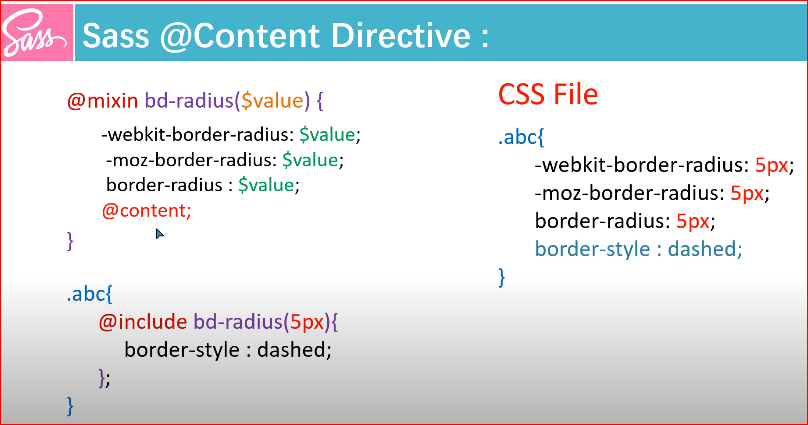


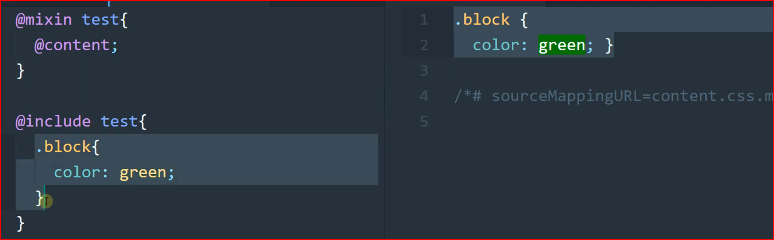




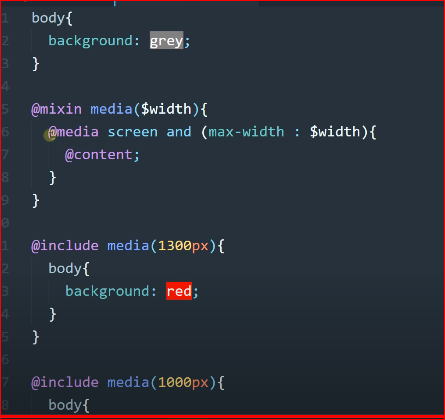


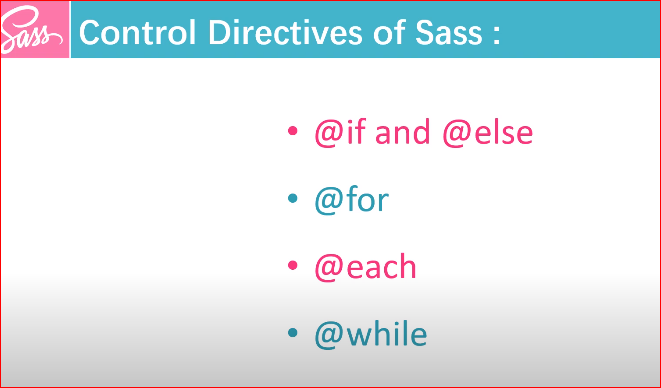


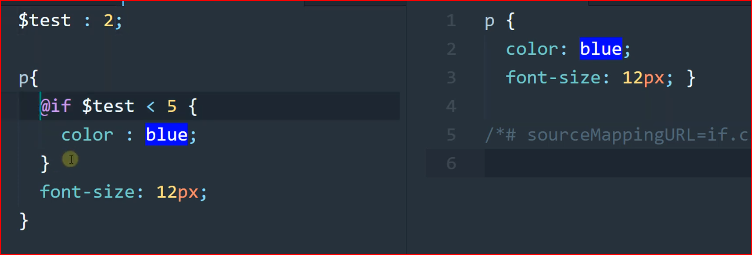


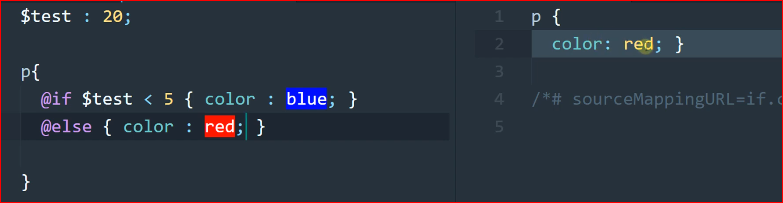


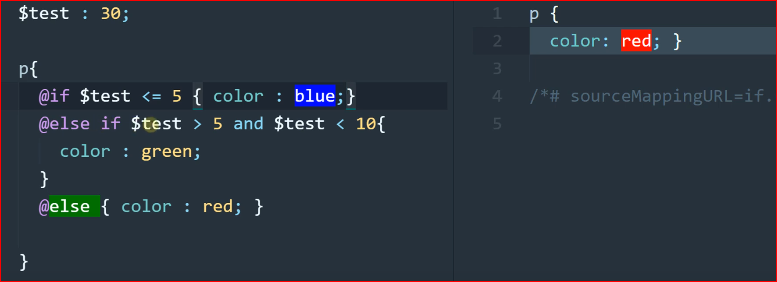
@content best and more uses in Media Query.

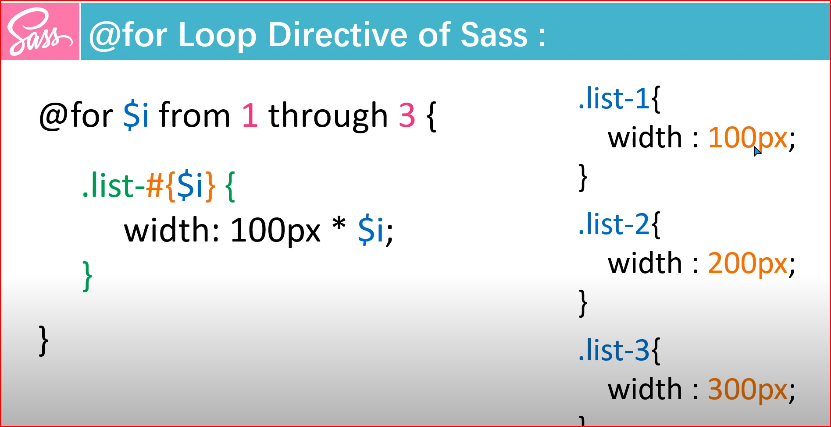


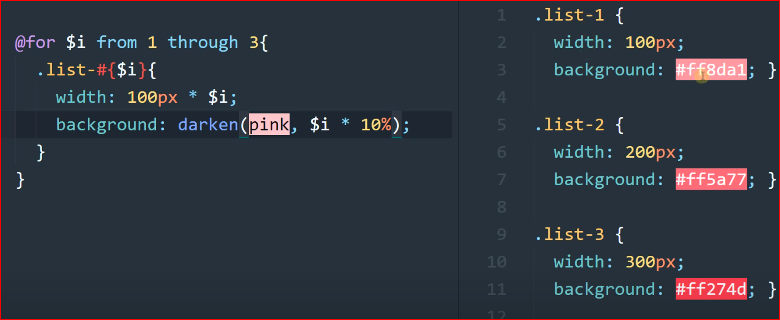


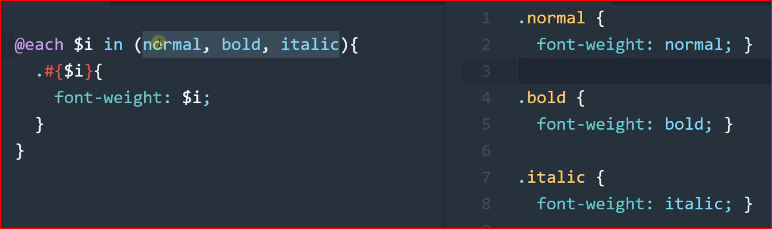


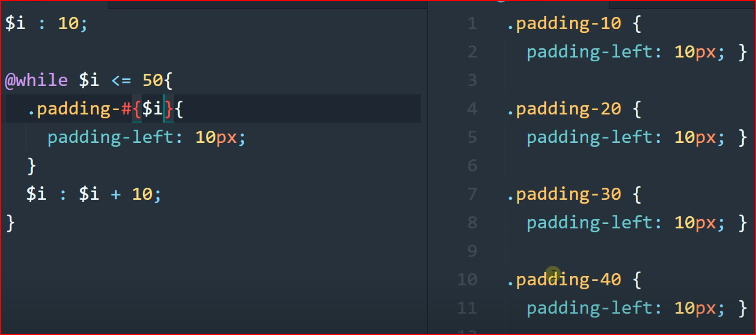


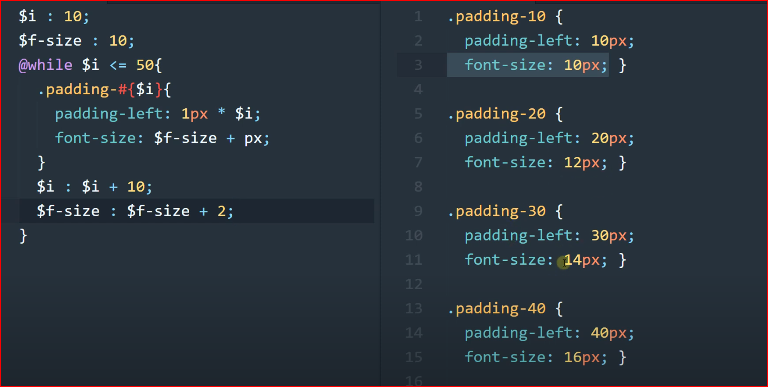












**14) List out the Data Types that SassScript supports?**

SassScript supports seven main data types

1. Numbers ( eg; 1,5 ,10px)
2. Strings of texts ( g., “foo”, ‘bar’, etc.)
3. Colors (blue, #04a3f9)
4. Booleans (true or false)
5. Nulls (e.g; null)
6. List of values, separated by space or commas (g., 1.5em, Arial, Helvetica etc.)
7. Maps from one value to another (g., ( key 1: value1, key 2: Value 2))

**15) How many ways can we use SASS?**

We can use SASS in three different ways:

* As a command line tool.
* As a standalone Ruby module.
* As a plug-in for any Rack-enabled framework.

**16) Which one is better, Sass or Less?**

Due to the following reasons, Sass is better than less:

* Sass provides the facilities to use logical statements like loops, conditions and also facilitates you to write reusable methods.
* The user of Sass can access the library of the company. Sass users can also use some awesome features like cross-browser support, legacy browser hacks, and dynamic sprite map generation.
* Compass also provides the facilities to add an external framework like Bootstrap on top, Blueprint.
* Sass provides you the facility to write your handy functions

**17) What are parentheses in Sass?**

Parentheses are used to provide a symbolic logic that affects the order of the operation. It is a pair of signs which are usually marked off by round () brackets or square [] brackets.

**18) What is the use of DRY-ing out a Mixin function in Sass?**

DDRY-ing out a Mixin function splits into two parts: the static part and dynamic parts.

The static Mixin contains the pieces of information that would otherwise get duplicated and the dynamic Mixin is the function that the user going to call.

**19) Describe the difference between Sass comment and regular CSS comment?**

Comments in regular CSS starts with /\* \*/ and Sass contains two commands. The single line comment with // and multiple CSS comments with /\* \*/.

**20) Which directive is used to detect the errors in SASS?**

Sass @debug directive is used to detect the errors and display the Sass Script expressions values to the standard error output stream. For example:

 $font-sizes: 10px + 20px;

$style: (

  color: #bdc3c7

);

.container{

  @debug $style;

  @debug $font-sizes;

}

**21) What are the requirements of SASS system?**

1. .Operating System - Cross platform
2. .Browser support - Internet Explorer, Google Chrome, Safari, Opera.
3. .Programming language - Ruby.

**22) What is the role of @media directive in SASS?**

The Sass @media directive is used to set style rules to different media types. It supports and extends the @media rules. This directive can be nested inside the selector SASS but the main impact is displayed to the top level of the style sheet. For example:-

 h2{

color: violet;

}

.style{

width: 500px;

@media screen and (orientation: portrait){

width:200px;

margin-left: 80px;

}

}

**23) What is the use of at-root directive in SASS?**

The Sass @at-root directive is a collection of nested rules that are used to style block at the root of the document. For example:-

 h2{

color: blue;

background-color: pink;

@at-root {

.style{

 font-size: 20px;

 font-style: bold;

 color: violet;

 }

 }

}

[**24) Is bootstrap LESS or SASS? Explain**](https://www.bestinterviewquestion.com/question/is-bootstrap-less-or-sass-explain-mmqvg958)

SASS and LESS both are the preprocessor of CSS that enhance the development of CSS but SASS is powerful as compared to LESS. Bootstrap provides a whole framework for creating the user interfaces that include the widgets, the grid layout system, the typographic styling, the Interactive Menu bars and many more. Bootstrap have used many preprocessors in the past for the CSS source code but out of SASS and LESS, Bootstrap uses LESS pre-processor and it has also included various customisations by using the fewer variables and the fewer components.

***But Recently BOOTSTRAP 3 has also given the SASS codebase as an alternate option but BOOTSTRAP 4 is being built completely on SASS codebase and this is the first time Bootstrap has using SASS.***

| **S.no** | **CSS** | **SASS** |
| --- | --- | --- |
| 1. | Codes are Long | Code is Short |
| 2. | Unstructured Document | Structured Document |
| 3. | Nesting:-Code Block are not Nested | Nesting:- Allow the nested code to be included in each other |
| 4. | Suitable for small web applications | Suitable for small web applications as well as large user interfaces |
| 5. | Offers less flexibility | Has an ability to add mixin, functions, variable etc |
| 6. | Multiple stylesheets cannot be joined | Multiple Stylesheets can be joined into only one Style Sheet |
| 7. | Darken and Lighting Functions are not available | Darken and Lighting Functions are available |
| 8. | Mathematical functions are not available | Mathematical functions are available |

| **S.no** | **SCSS** | **SASS** |
| --- | --- | --- |
| 1. | It is a New Syntax | It is an original and the old syntax |
| 2. | The file is saved with the extension .scss | The file is saved with the extension .SASS |
| 3. | In a code the use of semicolon and bracket{} is compulsory. | In a code, the use of semicolon and bracket{} is not allowed |
| 4. | There is no strict indentation | There is a strict indentation |
| 5. | Easier to learn | Difficult to learn as compared to SCSS |

**[25) Who is the father of SASS?](https://www.onlineinterviewquestions.com/sass-interview-questions/" \l "collapseUnfiled2)**

* The SASS language primarily designed by **Hampton Catlin** and developed by **Natalie Weizenbaum**.
* After the primary version, Natalie Weizenbaum and Chris Eppstein are continuing for SASS with a scripting language called SaasScript.

**[26) List some alternative of SASS?](https://www.onlineinterviewquestions.com/sass-interview-questions/" \l "collapseUnfiled4)**

**Some Alternative list of SASS**

* BOOTSTRAP, LESS, NODE-SASS, POSTCSS, COMPASS, Animate.css, STYLED-COMPO, STYLUS

**[27) How many output styles are there in sass?](https://www.onlineinterviewquestions.com/sass-interview-questions/" \l "collapseUnfiled11)**

**The sass** file converted into CSS file automatically, but some condition CSS file is not supported.

**The Sass supports four output style.**

* **Compact output style**: this is pointed on selector not property. It takes less space than an expanded and nested output style.
* **Compressed output style:** it provides whitespace to separate selectors. It takes less space than the other three styles.
* **Expanded output style:** this style to space then the default output tag. Each property expanded CSS style.
* **Nested output style:** this is the default sass output style.

**[28) How comfortable are you with Sass?](https://www.onlineinterviewquestions.com/sass-interview-questions/" \l "collapseUnfiled15)**

**We** know the prerequisite language as HTML, CSS with SASS style. I can work on Sass daily as a UI/UX designer and I am confident about my knowledge to work fast and minimum corrections. I rate for myself the basis of saas technology 8/10

**29) What is less?**

**Less or Leaner Style Sheets** is a CSS extension for backwards-compatible language. Less is an open-source dynamic style sheet language that runs on the client-side or server-side and is cross-browser compatible. It allows a user to modify, manage and reuse style sheets for any webpage.

**30) What is the difference between Sass and Less?**

| Sass | Less |
| --- | --- |
| Sass is based in Ruby. | Less is based on JavaScript. However, it was constructed in Ruby. |
| To assign variables, Sass uses a dollar ($) sign. | To assign variables Less uses an at (@) sign. |
| Less user-friendly documentation. | More user-friendly documentation. |
| The error message and the ability to report syntax errors are not as good as Less. | Better error message and ability to report syntax errors. |
| To integrate mixin, Sass uses the extension Compass. | To integrate mixin, Less uses the extension Preboot.less. |