Sass using math

SASS allows for mathematical operations such as addition, subtraction, multiplication and division. You cannot use incompatible units such as *px * px* or while adding number with *px* and *em* leads to produce invalid CSS. Therefore, SASS will display an error if you use invalid units in CSS. SASS supports relational operators like *<, >, <=, >=* and equality

Example:

- It can used in identifier such as *font-size* and SASS allows only valid identifiers.
- It can be used with two numbers without space i.e, 10-5 is similar to 10 - 5.
- It can be used as beginning of a negative number (-5).
- It can be used without considering space such as 5 -$myval is similar to 5 - $myval.
- It can be used as unary negation operator (-$myval).

_card.sass file

```sass
@use 'sass:math';

.card {
  display: block;
  padding: $base-padding;
  border: $base-border-thickness solid #ddd;
  box-shadow: $base-box-shadow;
  border-radius: math.div($base-border-radius ,3);
  margin: 20px;
  width: 20rem;

  .card-title {
    font-size: $base-font-size;
    padding-bottom: $base-padding;
    font-weight: bold;
    font-size: $font-size-lg;
  }
  .card-body {
    font-size: $base-font-size;

    a {
      text-decoration: underline;
    }
  }
}
```

```sass
@debug math.max(20px, 40px, 30px, 90px);
@debug math.min(20px, 40px, 30px, 90px);
@debug math.random(10000);
```

_variable.sass file

```sass
// theme colors
$primary: #326dee !default;
$secondary: #1ac886 !default;
$error: #d32752 !default;
$info: #f6c31c !default;

// spacing
$base-padding: 0.75rem;
$base-margin: 0.75rem;

// borders
$base-border-thickness: 1px;
$base-border-radius: 20px;

// box-shadow
$base-box-shadow: 1px 3px 5px rgba(0,0,0,0.1);

// font sizes
$base-font-size: 1rem;

// font sizes
$base-font-size: 1rem;
$font-size-sm: $base-font-size * 0.75; // 16 * 0.75
=  28  ;              1rem = 16 px;
$font-size-lg: $base-font-size * 1.5;
$font-size-xl: $base-font-size * 2;
$font-size-xxl: $base-font-size * 3;
```