



Table of Contents >

# Assets

Nuxt uses two directories to handle assets like stylesheets, fonts or images.

- The `public/` directory content is served at the server root as-is.
- The `assets/` directory contains by convention every asset that you want the build tool (Vite or webpack) to process.

## `public/` Directory

The `public/` directory is used as a public server for static assets publicly available at a defined URL of your application.

You can get a file in the `public/` directory from your application's code or from a browser by the root URL `/`.

## Example

For example, referencing an image file in the `public/img/` directory, available at the static URL `/img/nuxt.png` :

```
<template>
  
</template>
```

app.vue

# assets/ Directory

Nuxt uses Vite or webpack to build and bundle your application. The main function of these build tools is to process JavaScript files, but they can be extended through plugins (for Vite) or loaders (for webpack) to process other kind of assets, like stylesheets, fonts or SVG. This step transforms the original file mainly for performance or caching purposes (such as stylesheets minification or browser cache invalidation).

By convention, Nuxt uses the `assets/` directory to store these files but there is no auto-scan functionality for this directory, and you can use any other name for it.

In your application's code, you can reference a file located in the `assets/` directory by using the `~/assets/` path.

## Example

For example, referencing an image file that will be processed if a build tool is configured to handle this file extension:

```
<template>
  
</template>
```

app.vue

💡 Nuxt won't serve files in the `assets/` directory at a static URL like `/assets/my-file.png` . If you need a static URL, use the `public/` directory.

## Global Styles Imports

To globally insert statements in your Nuxt components styles, you can use the `Vite` option at your `nuxt.config` file.

## Example

In this example, there is a sass partial file containing color variables to be used by your Nuxt pages and components

```
$primary: #49240F;  
$secondary: #E4A79D;
```

assets/\_colors.scss

In your `nuxt.config`

SCSS    SASS

```
export default defineNuxtConfig({  
  vite: {  
    css: {  
      preprocessorOptions: {  
        scss: {  
          additionalData: '@use "@/assets/_colors.scss" as *;'  
        }  
      }  
    }  
  }  
})
```

SCSS

[Edit on Github](#)



© 2016-2023 Nuxt - MIT  
License

Enterprise

Design Kit  
Nuxt Studio

NuxtLabs

