



Let's create our project's file and folder structure.

We need a source folder with a sass folder, the images folder and an index.html file. In the index.html file just type a regular HTML structure:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Personal Portfolio</title>
</head>
<body>

</body>
</html>
```

In the sass folder create a styles.css file:

```
.test {
  font-size: 1em;
  display: flex;
}
```

Then we need a dist folder with empty css and images folders. Gulp will save the website's files here.

Now we have to structure the actual workflow in Gulp. Gulp can run single tasks that run once or run tasks that watch for changes in files, and then run a task or a series of tasks, every time. We'll first create each task, and then decide which ones have to be in a watch task.

Add the following code to gulpfile.js:

```
gulp.task('css', function() {
  return gulp.src('src/sass/**/*.scss')
    .pipe(sourcemaps.init())
    .pipe(sass({outputStyle: 'compressed'}).on('error', sass.logError))
    .pipe(autoprefixer({
      browsers: ['last 2 versions']
    }))
    .pipe(sourcemaps.write('./maps'))
    .pipe(gulp.dest('dist/css'))
    .pipe(browserSync.stream())
});
```

```

gulp.task('images', function(){
  return gulp.src('src/images/*')
    .pipe(imagemin())
    .pipe(gulp.dest('dist/images'))
});

gulp.task('copy', function() {
  return gulp.src('src/*.html')
    .pipe(gulp.dest('dist'))
    .pipe(browserSync.stream())
});

gulp.task('browserSync', function() {
  browserSync.init({
    server: {
      baseDir: 'dist'
    },
  })
});

gulp.task('watch', ['browserSync', 'css'], function(){
  gulp.watch('src/sass/**/*.scss', ['css']);
  gulp.watch('src/*.html', ['copy']);
});

```

That's a lot of code, but as you can see, once this is setup, the code can be copied to other projects easily, you do not need to worry anymore about reloading the page, configuring Sass, or adding those pesky vendor prefixes. This basic front-end workflow will save us a lot of time when developing the website.