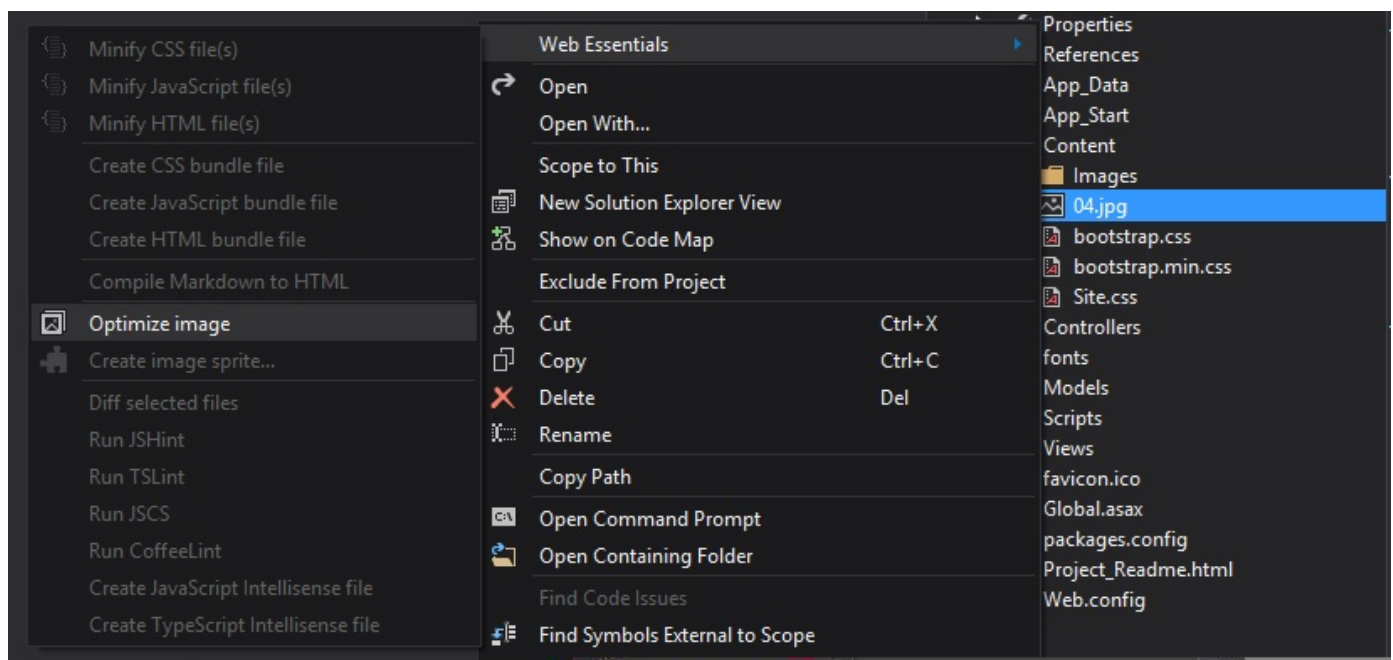


در این مطلب نکات کار با تصاویر را توسط افزونه‌ی Web Essentials بررسی می‌کنیم. این افزونه قابلیت‌های زیر را در کار با تصاویر در اختیار شما قرار می‌دهد: **بهینه‌سازی تصاویر**

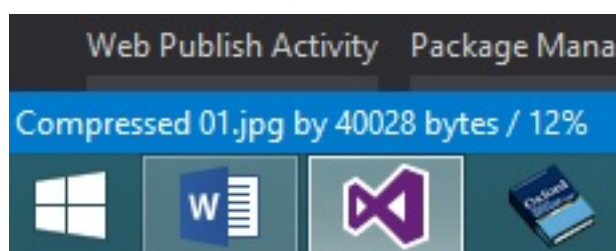
یکی از موارد مهمی که باید مورد توجه قرار بگیرد، استفاده از تصاویر کم حجم در وب‌سایت می‌باشد. روش‌های مختلفی جهت بهینه‌سازی تصاویر مورد استفاده در سایت وجود دارند، به طور مثال جهت بهینه‌سازی تصاویر PNG می‌توانید از ابزار PNGGauntlet استفاده کنید. همچنین [اینجا](#) نیز یک ابزار آنلاین موجود می‌باشد. افزونه‌ی Web Essentials این قابلیت را به آسانی در اختیار شما قرار می‌دهد؛ اینکار را می‌توانید توسط این افزونه به روش‌های زیر انجام دهید:

کلیک راست بر روی تصویر

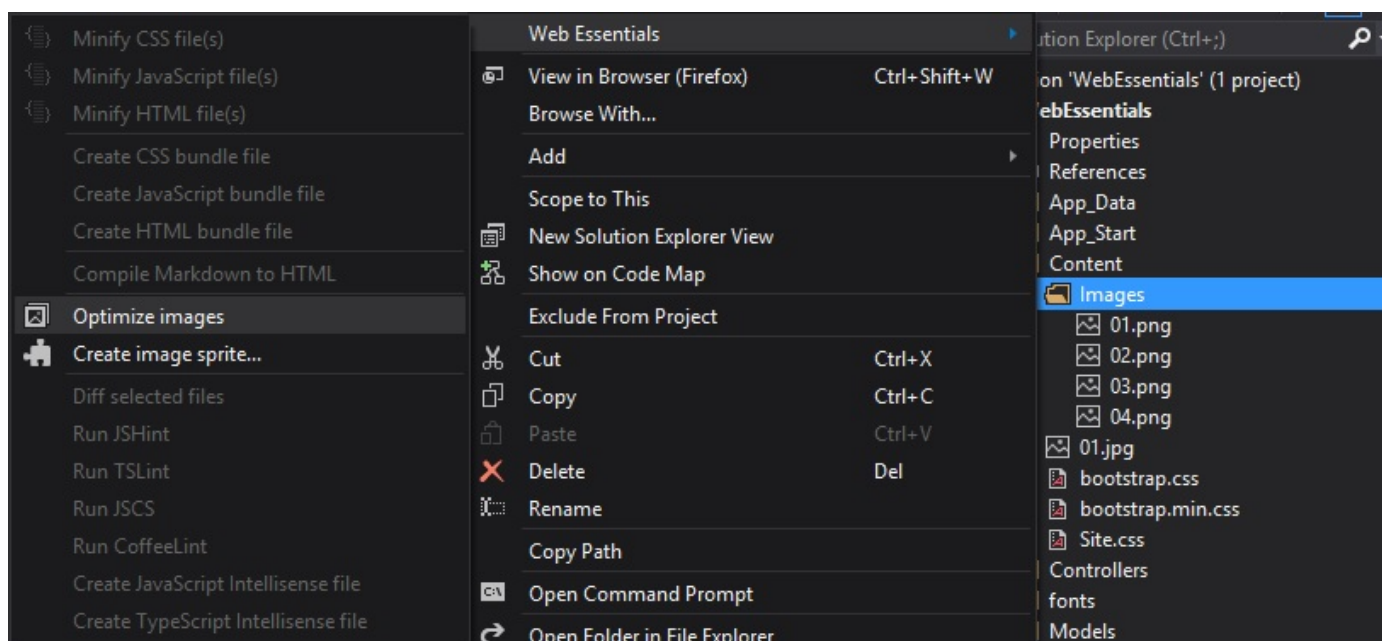
برای اینکار بر روی فایل‌ی که می‌خواهید optimize کنید، کلیک راست کرده و از منوی ظاهر شده گزینه Web Essentials و سپس Optimize Image را انتخاب کنید:



در قسمت status bar نیز می‌توانید نتیجه را مشاهده کنید:

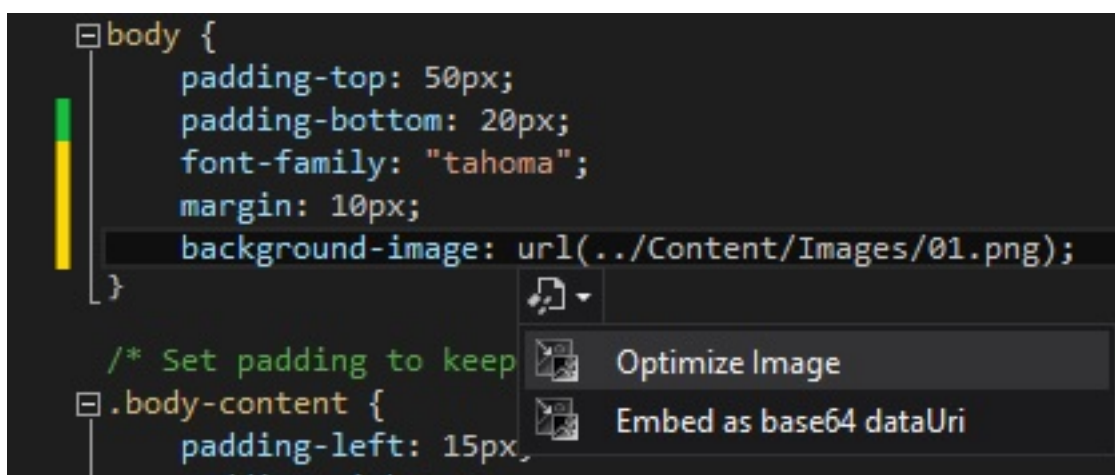


روال قبلی را می‌توانید برای چندین فایل انتخاب شده و یا یک پوشه تکرار کنید:



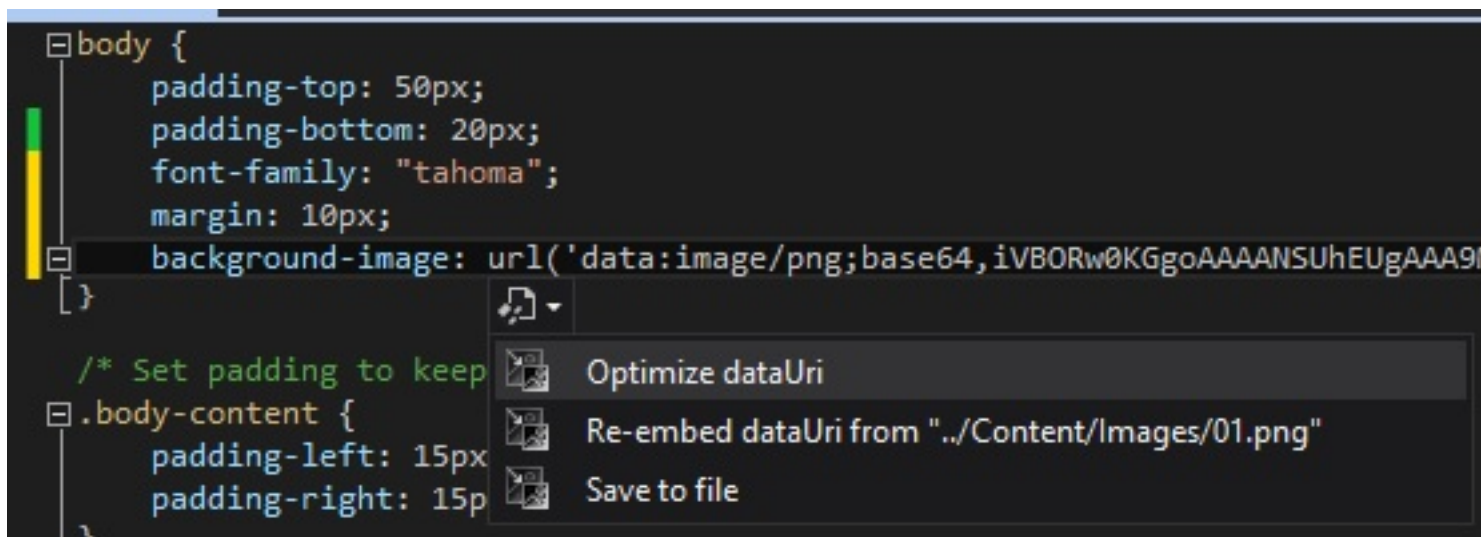
بهینه‌سازی تصاویر موجود در فایل‌های CSS

همچنین امکان بهینه‌سازی تصاویر داخل فایل‌های CSS نیز توسط این افزونه امکان پذیر است:



بهینه سازی تصاویر Base64 Encode

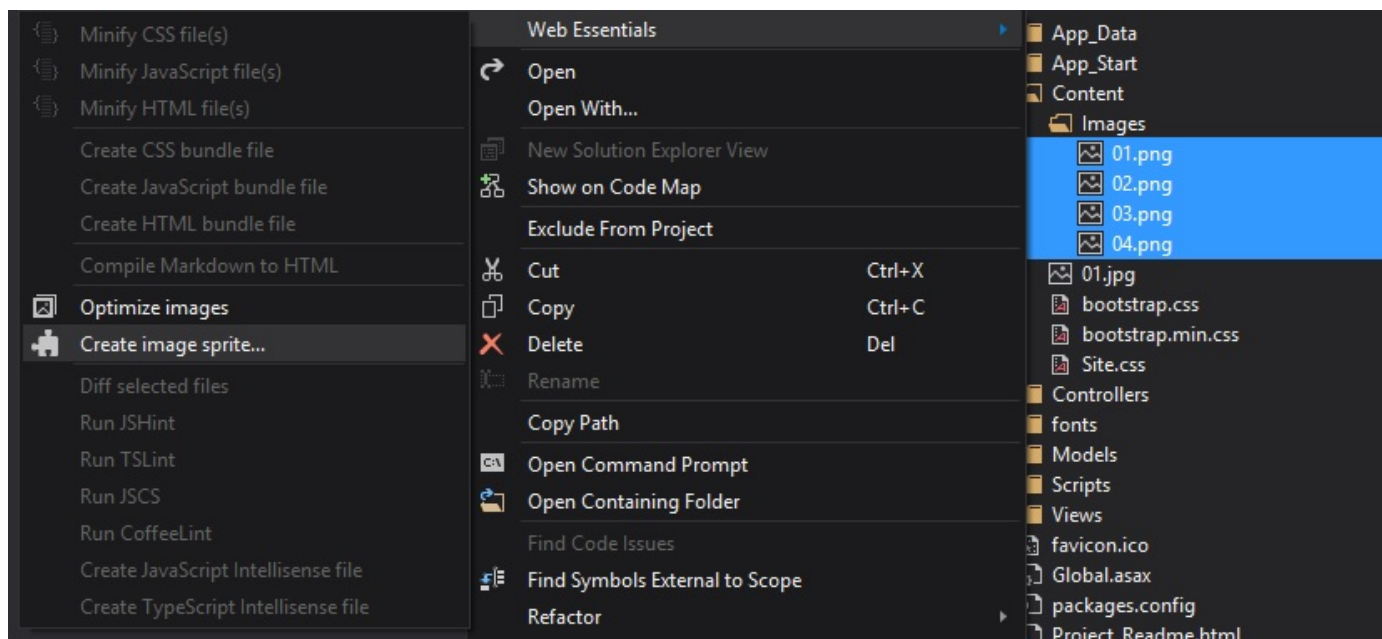
توسط این افزونه می‌توانیم تصاویر [Data Uri](#) را نیز بهینه سازی کنیم:



همانطور که در تصویر فوق مشاهده می‌کنید می‌توانیم تصاویری که به صورت Data Uri درون کد پیوست شده اند را با کلیک بر روی Save to file به صورت یک فایل ذخیره کنیم.

ایجاد تصاویر Sprite

یکی دیگر از قابلیت‌های افزونه Web Essentials امکان تهیه تصاویر به صورت [Sprite](#) می‌باشد. برای اینکار کافی است به این صورت عمل کنید:

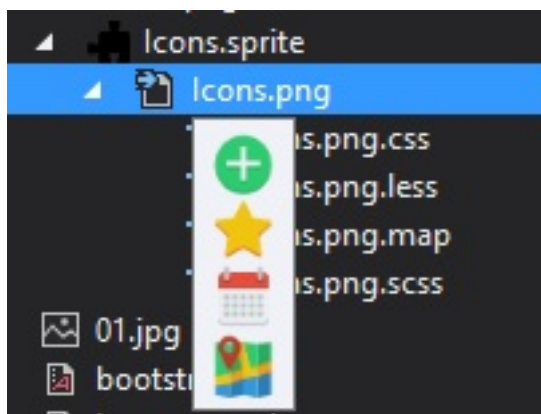


بعد از کلیک بر روی Create image sprite باید یک نام برای آن تعیین کنید و سپس بر روی کلید Save کلیک کنید. با اینکار یک فایل از نوع XML با پسوند sprite برای شما ساخته خواهد شد:

```
<?xml version="1.0" encoding="utf-8"?>
<sprite xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://vswebessentials.com/schemas/v1/sprite.xsd">
  <settings>
    <!--Determines if the sprite image should be automatically optimized after creation/update.-->
    <optimize>true</optimize>
    <!--Determines the orientation of images to form this sprite. The value must be vertical or
```

```
horizontal.-->
<orientation>vertical</orientation>
<!--File extension of sprite image.-->
<outputType>png</outputType>
<!--Determin whether to generate/re-generate this sprite on building the solution.-->
<runOnBuild>false</runOnBuild>
<!--Use full path to generate unique class or mixin name in CSS, LESS and SASS files. Consider
disabling this if you want class names to be filename only.-->
<fullPathForIdentifierName>true</fullPathForIdentifierName>
<!--Use absolute path in the generated CSS-like files. By default, the URLs are relative to sprite
image file (and the location of CSS, LESS and SCSS).-->
<useAbsolutePath>false</useAbsolutePath>
<!--Specifies a custom subfolder to save CSS files to. By default, compiled output will be placed
in the same folder and nested under the original file.-->
<outputDirectoryForCss />
<!--Specifies a custom subfolder to save LESS files to. By default, compiled output will be placed
in the same folder and nested under the original file.-->
<outputDirectoryForLess />
<!--Specifies a custom subfolder to save SCSS files to. By default, compiled output will be placed
in the same folder and nested under the original file.-->
<outputDirectoryForScss />
</settings>
<!--The order of the <file> elements determines the order of the images in the sprite.-->
<files>
<file>/Content/Images/01.png</file>
<file>/Content/Images/02.png</file>
<file>/Content/Images/03.png</file>
<file>/Content/Images/04.png</file>
</files>
</sprite>
```

یکی از زیر مجموعه‌های این فایل، تصویر نهایی می‌باشد، همچنین فایل‌های map، less، css و scss آن نیز تولید می‌شود:



به عنوان مثال فایل CSS تصویر فوق به صورت زیر می‌باشد:

```
/*
This is an example of how to use the image sprite in your own CSS files
*/
.Content-Images-01 {
/* You may have to set 'display: block' */
width: 32px;
height: 32px;
background: url('icons.png') 0 0;
}
.Content-Images-02 {
/* You may have to set 'display: block' */
width: 32px;
height: 32px;
background: url('icons.png') 0 -32px;
}
.Content-Images-03 {
/* You may have to set 'display: block' */
width: 32px;
height: 32px;
background: url('icons.png') 0 -64px;
```

```
}  
.Content-Images-04 {  
/* You may have to set 'display: block' */  
width: 32px;  
height: 32px;  
background: url('icons.png') 0 -96px;  
}
```

هر کدام از کلاس‌های فوق به یک تصویر در فایل مربوطه توسط image position اشاره می‌کند. شما می‌توانید با انتساب هر کدام از کلاس‌های فوق به یک المنت از آن تصویر استفاده نمائید:

```
<div class="Content-Images-01"></div>  
<div class="Content-Images-02"></div>  
<div class="Content-Images-03"></div>  
<div class="Content-Images-04"></div>
```

استفاده از تصاویر Data URIs

یکی دیگر از روش‌های کاهش درخواست‌های HTTP در یک سایت استفاده از [Data URIs](#) می‌باشد، توسط این روش می‌توانید فایل هایتان را درون HTML و یا CSS قرار دهید یا به اصطلاح embed کنید. به طور مثال جهت استفاده از یک تصویر می‌توانید به راحتی با آدرس دهی تصویر درون تگ img، تصویر را درون صفحه نمایش دهید:

```

```

همین کار را می‌توانیم توسط Data URIs انجام دهیم:

```

```

در کد فوق تصویر موردنظر را درون HTML به صورت embed شده قرار داده ایم، در این حالت دیگری نیازی به رفت و برگشت به سرور جهت نمایش تصویر نیست. **سینتکس Data URIs** به طور مثال تگ زیر را در نظر داشته باشید:

```

```

مقدار ویژگی src شامل موارد زیر است:

data: نام schema

image/png: نوع محتوا (content type)

base64: نوع encoding استفاده شده برای encode کردن اطلاعات

iVBOR...: اطلاعات encode شده.

توسط افزونه Web Essentials به راحتی می‌توانید تصویر موردنظرتان را به صورت Data URI تهیه کنید:

