

سوالات متداول

متداولترین سؤالات قالب

ما همیشه از سوالات پشتیبانی که از مشتریان خود می گیریم یاد می گیریم. در حالی که برخی از سؤالات به ما کمک می کنند تا محصول را بهتر کنیم و کدها را بهبود بخشیم تا کار ساده تر شود، برخی دیگر نیاز به توضیح مفصل دارند، بنابراین بهتر متوجه می شوید که چرا مشکل خاصی برای شما اتفاق می افتد یا چرا ما کدگذاری محصول را به این شکل انتخاب کرده ایم.

لطفا توجه داشته باشید. تمام کدهای جاوا اسکریپت در این قالب به گونه ای نوشته شده است که تا حد امکان کوتاه، سریع، آسان برای استفاده و قدرتمند باشد! ما به عملکرد صفحات خود اهمیت زیادی می دهیم و همیشه در تلاش هستیم تا بهترین کیفیت و بزرگترین مجموعه ویژگی را برای بهترین عملکرد ممکن به شما ارائه دهیم!

سوالات رایج

روی هر یک سوالات کلیک کنید تا به آن منتقل شوید.

- ⚠ آیا می توانم از Angular / React / Vue / یا فریمورک های دیگر استفاده کنم؟
- ⚠ وقتی سعی می کنم به یک لینک خارجی بروم، صفحه رفرش می شود. چرا؟
- ⚠ کد جاوا اسکریپت من فقط در صورتی کار می کند که صفحه را رفرش کنم.
- ⚠ چگونه جی کوئری را پیاده سازی کنم؟ من به آن نیاز دارم
- ⚠ پوشه plugins چیست؟
- ⚠ متغیرهای isAJAX / isPWA / pwaName در custom.js چیست؟
- ⚠ وب اپلیکیشن (PWA) نصب نمیشود / جعبه های PWA ظاهر نمی شوند، چرا؟

⚠ منوها سفید یا خالی نشان داده می شوند، چرا؟

توضیحات ساختار SCSS

- i ایجاد طرح رنگی دلخواه
- i چگونه می توانم لوگو یا سایر موارد خود را بدون SCSS تغییر دهم؟
- i حالت تاریک، رنگ ها و کوکی های پس زمینه

جاوا اسکریپت و AJAX

- i چرا ساختار صفحه اینگونه است؟
- i آیا هنوز هم می توانم از المان های پیش فرض بوت استرپ استفاده کنم؟
- i AJAX چگونه در قالب کار می کند؟
- i چگونه سیستم AJAX را غیرفعال کنم؟
- i چگونه سیستم PWA را غیرفعال کنم؟

پلاگین های CSS و جاوا اسکریپت

- i وارد کردن کلاسیک و فراخوانی (توصیه شده برای مبتدیان)
- i ایمپورت فایل ها بر حسب تقاضا و فراخوانی (برای عملکرد بهتر)

عملکرد و سرعت صفحه

- i صفحه من به اندازه کافی سریع نیست! میتونی کمک کنی؟
- i چرا از فایل های CSS و JS کوچکتر استفاده نمی کنید؟ و چرا؟

سوالات رایج

اینها برخی از مشکلات جزئی هستند که می توانند برای شما ظاهر شوند، بیشتر به دلیل دنبال کردن ساختارهای نادرست ظاهر می شوند. در صورتی که همچنان با مشکل مواجه هستید، لطفاً از طریق تیکت در وبسایت راست چین به بنده اطلاع دهید و با کمال میل به شما کمک خواهم کرد.

آیا می توانم از Angular / React / Vue یا فریمورک های دیگر استفاده کنم؟

بله. بسیاری از مشتریان قالب را بدون مشکل به قالب های SPA(Single Page Applications) تبدیل کرده اند. با این حال، برای اینکه با React/Vue/Angular یا هر فریم ورک SPA تداخل نداشته باشید، لطفاً custom.js را باز کنید و پارامتر isAJAX را روی false قرار دهید. در اینجا می توانید درباره این تنظیمات جهانی بیشتر بدانید. سیستم AJAX می تواند با نحوه مدیریت DOM های مجازی در فرم ورک های SPA درگیری ایجاد کند. علاوه بر این، این فریم ورک ها به سیستم AJAX نیاز ندارند، زیرا آنها از قبل دارای مدیریت حالت و روترهایی هستند که مشکلات سرعتی را که سیستم AJAX ما برای آن وجود دارد، مدیریت می کند.

isAJAX را روی false تنظیم کنید، و می توانید قالب را به هر فریم ورکی که می خواهید تبدیل کنید.

وقتی سعی می کنم به یک لینک خارجی بروم، صفحه رفرش می شود. چرا؟

ما عملکرد تغییر صفحه AJAX را در این سوال شرح می دهیم. اما، برای کوتاه کردن همه چیز، صفحه سعی می کند لینک های خارجی را در داخل صفحه شناسه div بارگیری کند و باعث بارگیری مجدد صفحه شود. برای جلوگیری از این امر، لینک پیش فرض کلاس را به هر لینکی که به سمت یک صفحه خارجی هدف قرار می دهید، اضافه کنید. لینک شما یک `href=Your-external-link class=default-link` خواهد بود.

کد جاوا اسکریپت من فقط در صورتی کار می کند که صفحه را رفرش کنم

همه پلاگین های جاوا اسکریپت خود را در داخل همه فایل های html بالای custom.js اضافه کنید و تابع فراخوانی خود را در داخل custom.js داخل تابع `init_template()` قرار دهید، نه داخل HTML. حتماً واردات افزونه های JS و CSS را نیز بررسی کنید. ۲ راه برای فراخوانی آنها وجود دارد، یکی از موارد ذکر شده در اینجا و یکی سیستم بارگذاری مدولار(modular) که برای عملکرد بهتر است.

چگونه جی کوئری را پیاده سازی کنم؟ من به آن نیاز دارم

jquery.js خود را به پایین صفحه، درست بالای bootstrap.min.js اضافه کنید. مطمئن شوید که آن را به همه صفحات اضافه کنید. بعد از اینکه jquery.js را به پایین صفحه خود اضافه کردید، کدهای فراخوانی jQuery خود را در داخل custom.js داخل تابع `init_template()` قرار دهید. اگر می خواهید jQuery را در فایل HTML خود اضافه کنید، باید با دنبال کردن این سوال، بارگذاری صفحه AJAX را غیرفعال کنید. استفاده از AJAX برای سرعت صفحه مهم است مگر اینکه صفحاتی را در Angular یا Vue یا React ایجاد کنید، جایی که روترها می توانند استفاده شوند. توصیه می کنیم همه کدهای فراخوانی سفارشی را در custom.js نگه دارید. اگر می خواهید از jQuery استفاده کنید، از مثال زیر استفاده کنید.

```
1. <script type="text/javascript" src="js/jquery.js"></script>
2. <script type="text/javascript" src="js/your-jquery-plugins-go-here.js"></script>
3. <script type="text/javascript" src="js/bootstrap.min.js"></script>
4. <script type="text/javascript" src="js/custom.js"></script>
```

پوشه plugins چیست؟

پوشه پلاگین ها پلاگین هایی را که ما از طریق لودر مدولار خود استفاده می کنیم را نگه می دارد. (آموزش در اینجا). این سیستم باعث می شود که صفحه بسیار سریعتر از حالت عادی بارگذاری شود، بدون اینکه کارایی کاهش یابد. برای اطلاعات بیشتر آموزش را بررسی کنید

متغیرهای isAJAX / isPWA / pwaName در custom.js چیست؟

اینها مقادیر boolean هستند که می توانند برای اصلاح عملکرد برخی از جنبه های قالب استفاده شوند. بیایید آنها را یکی یکی روشن کنیم.

```
var isPWA = true;
```

این متغیر، قالب را از pwa بودن یا غیرفعال شدن آن، فعال یا غیرفعال می کند، به این معنی که اگر این متغیر روی false تنظیم شود، service-worker بارگذاری نمی شود، بنابراین، صفحه را به یک قالب قدیمی HTML تبدیل می کند.

var isAJAX = true;

صفحات قالب از انتقال صفحه و سیستم AJAX استفاده می کند که در این سؤال متداول به طور مفصل توضیح داده شده است. در صورتی که می خواهید از منطق Router خود استفاده کنید یا React، Angular، Vue را پیاده سازی کنید و اسکریپت ها بارگیری نمی شوند، می توانید این متغیر را روی false تنظیم کنید. یا اگر قصد دارید یک فریم ورک SPA مانند React یا Vue را پیاده سازی کنید، که در آن سیستم AJAX قدرت کمتری نسبت به سیستم های ارائه شده توسط این فریمورک دارد، به شدت توصیه می شود. در غیر این صورت سیستم isAJAX نباید حذف شود زیرا سرعت قالب های قدیمی HTML را افزایش می دهد.

var remind = ۱;

اگر جعبه های pwa-install را در صفحه اصلی صفحه خود اضافه کرده اید که از کاربر می خواهد برنامه را نصب کند، ۲ گزینه دارند، نصب یا شاید بعداً. هنگامی که کاربر ممکن است روی بعداً کلیک کند، مقدار اینجا نشان می دهد که چند روز قبل از درخواست مجدداً در صورت بازدید دوباره نشان داده می شود. اگر کاربر نصب برنامه را در صفحه اصلی خود رد کند، اعلان نصب pwa دوباره به مدت ۲۴ ساعت ظاهر نمی شود.

var pwaNoCache = false;

این تضمین می کند که PWA شما همیشه داده های تازه را ارائه می دهد. هر زمان که PWA خود را پس از بستن کامل آن بر روی تلفن خود راه اندازی می کنید به طور خودکار تمام حافظه های پنهان را پاک می کند و محتوای تازه را دریافت می کند. لطفاً توجه داشته باشید که دستگاه های iOS ممکن است این ویژگی را نادیده بگیرند، اما iOS به طور خودکار کش PWA را هر ۲۴ ساعت یا بیشتر پاک می کند. این محدودیتی از جانب اپل است که ما هیچ کنترلی روی آن نداریم.

```
var pwaScope = "https://yourserver.com/your-folder/_server-worker.js";
```

مکان PWA است که فایل server-worker شما قرار می گیرد. آدرس کامل فایل مورد نیاز است. لطفاً فایل Service Worker را در همان پوشه با فایل HTML خود نگه دارید، در غیر این صورت، PWA شما ثبت نمی شود.

وب اپلیکیشن (PWA) نصب نمیشود / جعبه های PWA ظاهر نمی شوند، چرا؟

فقط ۳ دلیل می تواند وجود داشته باشد که PWA درخواست نصب PWA را نشان نمی دهد، که به راحتی قابل رفع هستند، بنابراین بیایید به آنها بپردازیم.

HTTPS/SSL - برای اینکه وب اپلیکیشن کار کند سرور شما باید ایمن باشد. این یک الزام برای همه وب اپلیکیشن ها است و بدون گواهی SSL نمی توان آنها را نصب کرد. لطفاً قبل از انجام هر یک از موارد زیر مطمئن شوید که سرور شما دارای گواهی SSL است.

service-worker یا manifest.json هیچ محدوده ای ندارند - فایل server-worker قلب یک وب اپلیکیشن است. این فایل باید وجود داشته باشد تا سایت شما به PWA تبدیل شود. لطفاً در ابتدای فایل custom.js، `var pwaScope` و `var pwaLocation` را پیدا کنید. لطفاً آدرس های موجود در آن را با پوشه اصلی که فایل server-worker شما در آن قرار دارد و همچنین مقصد فایل service-worker.js جایگزین کنید. پس از انجام این کار، manifest.json را باز کنید و محدوده و start_url خود را تنظیم کنید. پس از انجام این کار، PWA شما اکنون به درستی راه اندازی شده است. لطفاً توجه داشته باشید، فایل های server-worker یا manifest را به پوشه اسکریپت منتقل نکنید. این فایل ها باید در همان مکان با فایل index.html شما باشد.

نصب به صورت لوکال - وب اپلیکیشن برای نصب به سرور نیاز دارد. برای نصب وب اپلیکیشن باید از یک سرور محلی یا وب سایت واقعی خود با شرایط فوق استفاده کنید.

جعبه های نصب وجود ندارد - جعبه های نصب را می توانید در پایین فایل index.html پیدا کنید، به نام های menu-pwa-install-android و menu-pwa-install-ios. این جعبه ها باید در صفحه ای باشند که می خواهید نشان داده شوند. این جعبه ها نمی توانند به صورت دستی یا کدنویسی توسط کاربر فعال شوند و فقط در صورت رعایت شرایط فوق نشان داده می شوند.

منوها سفید یا خالی نشان داده می شوند، چرا؟

لطفا توجه داشته باشید. منوهای خارجی از طریق جاوا اسکریپت بارگیری می شوند. این بدان معنی است که آنها برای مشاهده به یک محیط سرور، مانند AMPPS، XAMP یا WAMP نیاز دارند. اگر منو، سفید بارگذاری شده است، دلیل این امر است.

ساختار SCSS

اولا، اگر در SCSS مبتدی هستید، نگران نباشید. این بیشتر شبیه CSS با متغیرهای قابل استفاده مجدد است.

اگر قبلاً به SCSS عادت کرده اید، هنگام ویرایش صفحات، می توانید امان ها را در style.css را با استفاده از جستجوی برای پیدا کردن کلاس مورد نظر خود یا با استفاده از SCSS ویرایش کنید. ما ساختار SCSS را طوری ساختیم که فایل های SCSS Bootstrap را در آنجا و دست نخورده نگه دارد.

این ساختار پوشه SCSS است

../bootstrap/

../components/

../pages/

../plugins/

../highlights/

../productName/

../_bootstrap.scss/

../style.scss

بیایید توضیح دهیم.

`/bootstrap/..` - شامل سیستم پیش فرض بوت استرپ SCSS، بدون تغییرات است. می توانید برای حذف عناصری که نیاز ندارید به `bootstrap/bootstrap.scss/` بروید یا از منطق Bootstrap SCSS برای ویرایش این فایل پیروی کنید.

`/components/..` - شامل تمام افزونه های ساخته شده سفارشی ما است که قبلاً توسط بوت استرپ ارائه شده است.

`/pages/..` - شامل تمام صفحات ساخته شده سفارشی ما است

`/productName/..` - شامل تمام عناصری است که محصول ما برای اجرای موثر به آنها وابسته است، از جمله حالت تاریک، سرصفحه، پاورقی، منو، لوگوها و همچنین ساختار و کلاس های کاربردی. این همچنین شامل متغیرهایی است که برای فونت ها، حاشیه ها، سایه ها و رنگ حالت تاریک استفاده می کنیم.

ساخت طرح رنگی دلخواه

برای ایجاد طرح رنگی خود، توصیه می کنیم از SCSS استفاده کنید. در حالی که ممکن است پیچیده به نظر برسد، اما فقط ۵ مرحله برای ایجاد یک رنگ جدید لازم است.

۱. برنامه کامپایلر (scout-app Compiler) را در [اینجا](#) دانلود و نصب کنید. این برنامه ۱۰۰٪ رایگان و ساده ترین کامپایلر برای تبدیل SCSS به CSS است.

۲. پس از نصب برنامه آن را باز کنید. اکنون که آن را باز کردید، فقط پوشه اصلی (کل پوشه) را که قالب را در داخل برنامه دارید بکشید و رها کنید. توجه داشته باشید که برنامه به طور خودکار فایل های ورودی و خروجی SCSS را اضافه می کند. مطمئن

شوید که روی آیکون پخش در نوار کناری سمت چپ برنامه Scout-app کلیک کرده اید تا محصول زمانی که تشخیص دهد شما هر فایل scss را ویرایش کرده اید شروع به ایجاد فایل های css کند. این بدان معناست که اکنون، هنگامی که پوشه محصول/scss را باز می‌کنید و هر گونه ویرایشی روی آن فایل‌ها انجام می‌دهید، تغییرات به‌طور خودکار روی فایل‌های CSS موجود اعمال می‌شود.

۳. به `product-folder/scss/highlights` بروید و همه فایل‌ها را از پوشه کپی کنید. مطمئن شوید که همین نام را حفظ کنید. به عنوان مثال اگر می‌خواهید یک رنگ awesomeBuyer ایجاد کنید، فایل شما **حتماً** باید `highlight_awesomeBuyer.scss` نام داشته باشد.

۴. `highlight_awesomeBuyer.scss` را در ویرایشگر کد خود باز کنید (مطمئن شوید که برنامه باز شده است، پوشه محصول کشیده شده و در آن رها شده است، و شما روی نماد پخش کلیک کرده اید). هنگامی که `highlight_awesomeBuyer` خود را در بالای کد باز کردید، رنگ‌های موجود در `$colorHighlightLight` و `$colorHighlightDark` را با رنگ‌های دلخواه خود جایگزین کنید. مطمئن شوید که این رنگ‌ها HEX هستند. همین! پس از ذخیره، طرح رنگ جدید به‌طور خودکار وارد می‌شود.

۵. در فایل HTML خود، در تگ `body`، اکنون می‌توانید `data-highlight="highlight-wesomeBuyer"` را تنظیم کنید و قالب مطمئن می‌شود که هر عنصر با کلاس `color-highlight`, `bg-highlight`, `border-highlight`, `gradient-highlight` به رنگی که اضافه کردید احترام می‌گذارد.

نکته: هنگام استفاده از برنامه، می‌توانید Environment را روی LF - Compressed - Production تنظیم کنید که باعث می‌شود فایل‌های CSS شما بسیار بهینه‌تر و سریع‌تر بارگذاری شوند.

چرا به جای CSS از SCSS استفاده کنیم؟

خیلی ساده است. ما به‌روزرسانی‌های دائمی را برای شما منتشر می‌کنیم و همیشه دقیقاً فایل‌هایی را که تغییر کرده‌اند ذکر می‌کنیم. در بیشتر موارد این فایل‌های SCSS هستند. جایگزین کردن یک فایل SCSS و کامپایل مجدد در CSS به جای ایجاد تغییراتی در خطوط کد که دیگر با کد ویرایش شده شما مطابقت ندارد، بسیار ساده‌تر است.

بیایید بگوییم که آکاردئون را بهبود بخشیم. در `style.css` در کد اصلی، این می‌تواند در خط ۱۰۰۰ باشد. اگر ویرایش‌هایی بالاتر از آن خط انجام داده‌اید، زمانی که یک به‌روزرسانی را منتشر می‌کنیم، این برای شما متفاوت خواهد بود. به جای اینکه شما را با یافتن خطوط بی‌شماری کد که اکنون می‌توانند بر اساس ویرایش‌هایتان متفاوت باشند، تحت فشار قرار دهید، گفتن "فقط فایل `accordions.scss`" را جایگزین کنید، بسیار ساده‌تر است. همچنین، فایل‌های SCSS تکه‌ای هستند، به این معنی که کد کمتری دارند و پیگیری به‌روزرسانی آنها بسیار آسان‌تر است.

من SCSS را دوست ندارم. آیا می‌توانم به جای آن از CSS استفاده کنم؟

بله حتما! اگرچه، ما به شدت توصیه می‌کنیم که SCSS را امتحان کنید! ما محصولات خود را به روز می‌کنیم و ویژگی‌های جدید را به طور مداوم اضافه می‌کنیم، بنابراین این امر برای شما در به روز رسانی در کوتاه‌ترین زمان ممکن با کمترین استرس ضروری خواهد بود. استفاده از CSS از شما می‌خواهد تا کلاس‌ها را به صورت دستی تنظیم کنید و کلاس‌های جدیدی را جایگزین یا اضافه کنید. SCSS راهی بسیار ساده‌تر از آن چیزی است که به نظر می‌رسد! ۳ مرحله اول بالا همه چیزهایی را که برای یک مبتدی در مورد SCSS باید بداند را پوشش می‌دهد و می‌توانید کدهای کلاسیک CSS را بدون داشتن دانش SASS درون آن بنویسید. SCSS فقط CSS های کوچکتر و برنامه‌ای‌تر را در یک فایل CSS معمولی کامپایل می‌کند.

چگونه می‌توانم لوگو یا موارد دیگر خود را بدون SCSS تغییر دهم

`style.css` را باز کنید و از جستجو برای ویرایش `header-logo` یا `footer-logo` استفاده کنید. با کلیک راست در Google Chrome و کلیک بر روی "Inspect Element" می‌توانید نام کلاس‌هایی را که می‌خواهید جستجو کنید پیدا کنید. در پنجره باز شده، نام کلاس را خواهید دید و در چه خطی می‌توان آن را در `style.css` یافت

حالت تاریک، رنگ‌ها و کوکی‌های پس‌زمینه

قالب تنظیماتی را که برای هایلایت رنگ و رنگ پس‌زمینه دارید، در صورتی که آنها را از منوی پایین تنظیم کنید، و حتی برای حالت تاریک یا حالت روشن به خاطر می‌آورد. این کار به طور خودکار با کوکی‌ها بر اساس رنگ برجسته یا پس‌زمینه‌ای که از صفحه انتخاب کرده‌اید انجام می‌شود. برای حذف این کوکی‌ها، کوکی‌ها را از صفحه پاک کنید. اگر صفحه شما از منوها برای تغییر رنگ‌ها و پس‌زمینه‌ها استفاده نمی‌کند، کوکی‌ها ایجاد نمی‌شوند.

جاوا اسکریپت و AJAX

چرا ساختار صفحه اینگونه است؟

هنگام ساخت فریم‌ورک قبلی ما، از یک سیستم بارگذاری بسیار کوچک و سریع استفاده کردیم که لزوماً برای کار کردن به AJAX نیاز نداشت، اما هنگام کار با بوت‌استرپ، عملکرد دستگاه‌های تلفن همراه یک مشکل است.

این ساختار به ۳ دلیل به این صورت طراحی شده است.

۱. برای فعال کردن سیستم انتقال صفحه AJAX برای تغییر محتوای `id=page`
۲. برای ایجاد افکت‌ها هنگام باز کردن اکشن‌شیت‌ها و مدال‌های اکشن، چیزی که در بوت‌استرپ ارائه نمی‌شود.
۳. برای بارگیری همه اسکریپت‌ها و سبک‌ها فقط در اول بازدید صفحه. این به شدت عملکرد را افزایش می‌دهد.

برای اکثر کاربران، پیروی از ساختار ساده است. صفحه `div id` را به عنوان `body` در نظر بگیرید. همه چیز داخل آن می‌رود. محتوای صفحه را به عنوان مکانی در نظر بگیرید که فقط محتوایی که در صفحه نمایش است قرار می‌گیرد. این ساده‌ترین تغییر ممکن است که می‌توانیم به آن فکر کنیم و شما را ملزم به یادگیری ساختارهای کد پیچیده نیست.

سیستم AJAX اینجاست تا صفحات را سریع‌تر و روان‌تر بارگذاری کند.

آیا هنوز هم می‌توانم از المان‌های پیش فرض بوت استرپ استفاده کنم؟

بله، کاملاً می‌توانید! فقط مطمئن شوید که اگر از Modals، Toasts، Snackbars استفاده می‌کنید، آنها را خارج از محتوای `class="page-content"` خود قرار دهید و مطمئن شوید که کدهای فراخوانی شما برای آنها در داخل `custom.js` در تابع `init_template()` درج شده است. می‌توانید در بخش چگونه کد جاوا اسکریپت خود را اضافه کنم اطلاعات بیشتری در مورد این موضوع بخوانید.

AJAX چگونه در قالب شما کار می‌کند؟

سیستم AJAX ما برای انتقال صفحه و همچنین برای بهینه سازی انبوه سرعت صفحه است. این بر روی یک مفهوم بسیار ساده کار می‌کند. به عنوان مثال، هنگامی که برای اولین بار به سایت دسترسی پیدا می‌کنید، سیستم AJAX ما به طور خودکار بخش‌های استایل و اسکریپت‌های صفحه شما را به یاد می‌آورد (بدون کش). برای مثال وقتی روی دکمه ای کلیک می‌کنید تا به `awesomeBuyer.html` بروید، کد فقط جایگزین محتوای موجود در داخل `id=page` می‌شود.

هنگامی که صفحه `awesomeBuyer.html` در داخل `id=page` تزریق شد، تابع `init_template()` از `custom.js` از بین می‌رود و دوباره فعال می‌شود. این کار حداکثر کارایی را تضمین می‌کند و در عین حال مطمئن می‌شود که تمام جاوا اسکریپت شما همچنان بدون خطر نشت حافظه یا مشکلات عملکرد در دستگاه‌های با قدرت کمتر اجرا می‌شود.

این به این معنی است که هیچ فراخوانی با فایل‌های JS یا Style برقرار نمی‌شود و همچنان به‌عنوان مواردی که هنگام ورود به سایت بارگیری شده‌اند استفاده می‌شوند. برای دستگاه‌های خاصی که در هر صفحه فراخوانی انجام می‌دهند، مانند اندروید و بیشتر اوقات iOS، این سرعت صفحه را به طرز چشمگیری افزایش می‌دهد.

چگونه سیستم AJAX را غیرفعال کنم؟

اگر انتقال صفحه AJAX را غیرفعال کنید، صفحه شما کمی کندتر احساس می شود، اساساً مانند هر سایت دیگری HTML، CSS و JS که بوت استرپ را اجرا می کند، بارگذاری می شود. با این حال، اگر برای وردپرس، لاراول، دروپال، جوملا یا هر CMS دیگری که در آن باید صفحاتی را از قبل بسازید، برنامه نویسی می کنید، ممکن است لازم باشد.

برای غیرفعال کردن AJAX تنها کاری که باید انجام دهید این است که `custom.js` را باز کنید و تنظیم کنید: `isAJAX = false` - در بالای `custom.js`

چگونه سیستم PWA را غیرفعال کنم؟

تنها کاری که باید انجام دهید این است که `custom.js` را باز کنید و تنظیم کنید: `isPWA = false` - در بالای `custom.js`

پلاگین های CSS و جاوا اسکریپت

نحوه وارد کردن فایل ها به شیوه قدیمی و فراخوانی آنها (توصیه شده برای مبتدیان)

برای اجرای هر اسکریپت و سبکی در تمام صفحات، به خصوص در هنگام استفاده از AJAX، روش صحیحی برای انجام آن وجود دارد.

۱. همه اسکریپت های جاوا اسکریپت (پلاگین ها) خود را در داخل همه فایل های html، درست بالای custom.js در پایین صفحه قرار دهید.

۲. کد سفارشی جاوا اسکریپت / کدهای فراخوانی خود را در داخل custom.js در تابع init_template() قرار دهید.

این ضروری است، بنابراین اگر شما مستقیماً یا از صفحه دیگری وارد اسکریپت شوید، صفحه می تواند به اسکریپت دسترسی پیدا کند، و تابع init_template() هنگامی که در صفحه جدیدی فرود می آید فعال می شود.

آیا می توانم از جاوا اسکریپت مانند هر قالب کلاسیک دیگری استفاده کنم؟ بدون AJAX؟

بله حتما! در custom.js در بالای صفحه، isAJAX=true را پیدا کنید. پارامتر و آن را روی false قرار دهید.

ایمپورت فایل ها بر حسب تقاضا و فراخوانی (برای عملکرد بهتر)

از آنجایی که صفحه ما دیگر به طور پیش فرض از jQuery استفاده نمی کند، وارد کردن و فراخوانی افزونه ها را می توان به صورت ماژولار انجام داد. به این معنی که می توانید آنها را از custom.js بر اساس یک ماشه وارد کنید. ماشه شناسایی می شود و اسکریپت به صورت خارجی بارگیری می شود و پس از بارگیری کامل صفحه شما فعال می شود. این سیستم با هر فریم ورک back-end یا سفارشی مانند ReactJS یا VueJS یا AngularJS کاملاً کاربردی است زیرا کاملاً با جاوا اسکریپت خالص ساخته شده است. بیایید گام به گام توضیح دهیم که چگونه کار می کند. اما ابتدا به چند سوال متداول در مورد این سیستم پاسخ می دهیم.

چرا این بهتر از بارگذاری صفحه به صفحه معمولی است؟

برای مثال پلاگین charts.js را در نظر بگیرید. این افزونه حدود ۳۲۰ کیلوبایت است. این یک ضربه بزرگ برای اولین بارگذاری صفحه است که زمانی که می خواهید نرخ پرش خود را پایین نگه دارید و کاربران خود را در صفحه خود نگه دارید بسیار مهم است. به عنوان مثال، با توجه به اینکه اکثر سایت ها نمودار دایره ای را در صفحه اصلی ارائه نمی کنند، بارگیری این افزونه با صفحه به طور همزمان

بی‌فایده خواهد بود. بنابراین، سیستم ماژولار می‌داند که چه زمانی یک نمودار در صفحه وجود دارد، و افزونه را در صورت تقاضا بارگیری می‌کند، بنابراین، با بارگذاری بسیار سریع‌تر صفحه، تجربه کلی کاربر را بسیار بهتر می‌کند.

می‌توانم این کار را با هر صفحه HTML انجام دهم، چرا سیستم شما بهتر است؟

البته، شما می‌توانید این کار را حتی با صفحه ما انجام دهید، کافی است `isAJAX = false` را تنظیم کنید. در بالای `custom.js` قرار دارد و اکنون می‌توانید فایل‌های JS خود را به طور معمول مانند هر قالب HTML اصلی دیگر بارگیری کنید. اما هدف اصلی ساخت صفحه با AJAX این است که صفحه را سریع‌تر کنید، بسیار سریع‌تر از قالب‌های معمولی HTML، زیرا وقتی از صفحه‌ای به صفحه دیگر می‌روید، برای اسکرپت‌ها فراخوانی وجود ندارد. سیستم مدولار دقیقاً این کار را انجام می‌دهد. این فقط اسکرپت‌های بزرگ را در صفحاتی که مورد نیاز است ارائه می‌دهد، نه در همه صفحات.

اما من به پلاگین خود در اولین بارگذاری نیاز دارم. آن وقت چی؟

از سیستم ایمپورت قدیمی استفاده کنید. اگر اسکرپت شما در صفحه اول مورد نیاز است. می‌توانید از سیستم مدولار نیز استفاده کنید، عملکرد کلی یکسان خواهد بود. برای بهترین عملکرد، اگر به اسکرپت در صفحه اول نیاز دارید، بهتر است از روش اول استفاده کنید.

استفاده از لودر مدولار

به `custom.js` بروید و از جستجو برای یافتن کلمه `"uniqueID"` استفاده کنید. کدی شبیه به کد زیر خواهید دید.

```
1. {  
2.   id: 'uniqueID', // MUST EXIST  
3.   plug: 'pluginName/plugin.js', // MUST EXIST  
4.   call: 'pluginName/pluginName-call.js', // optional
```

```
5. style: 'pluginName/pluginName-style.css', // optional
6. trigger: '.pluginLoadTrigger' // MUST EXIST
7. },
```

بیا باید قدم به قدم توضیح دهیم که هر خط چه کاری انجام می دهد.
لطفاً توجه داشته باشید که همه افزونه ها در پوشه plugins/ ذخیره می شوند

خط ۱ - یک آرایه پلاگین جدید را باز می کند که در آن تمام اقدامات مربوط به افزونه شما قرار می گیرد.

خط ۲ - شناسه منحصر بفردی است که سیستم ما برای تعیین اینکه آیا افزونه قبلاً بارگذاری شده است یا خیر و البته زمانی که صفحه دیگر از آن استفاده نمی کند، آن را از بین ببرد یا بارگیری کند. این روش برای اطمینان از اینکه عملکرد در حداکثر آن است و عدم وجود نشستی حافظه استفاده می شود.

خط ۳ - محل اصلی افزونه است. این خود افزونه است، نه تابع فراخوانی.

خط ۴ - تابع فراخوانی افزونه است. پس از بارگیری موفقیت آمیز افزونه، این فایل راه اندازی می شود و افزونه شما را راه اندازی می کند.

خط ۵ - فایل CSS مربوط به افزونه است. نیازی به نگه داشتن CSS اضافی برای یک افزونه نیست، بنابراین ما آن را با خود افزونه نیز بارگذاری می کنیم.

خط ۶ - عنصر ماشه ای است که جاوا اسکریپت می خواند و مشخص می کند که خطوط فوق باید شروع شوند. بنابراین، برای مثال، در این مورد، pluginLoadTrigger است، اما شما همچنین می توانید از . - yourClass کلاس ها یا yourID# - شناسه ها یا برچسب های داده [data-your-trigger] استفاده کنید.

برای وارد کردن یک افزونه جدید برای خود، کافی است به custom.js بروید و از جستجو برای یافتن کلمه "uniqueID" استفاده کنید.
کافی است یک آرایه را کپی کنید.

این چگونه کار می کند؟

لودر مدولار با جستجوی trigger کار می کند. اگر ماشه قرار داشته باشد، افزونه، فراخوانی و استیایل بارگیری می شود.

آیا می توانم فقط یک بخش را وارد کنم؟

بله حتما. برای مثال پلاگین شمارش معکوس ما را در نظر بگیرید. این فقط افزونه را بارگیری می کند، بدون هیچ سبک یا فراخوانی. واردکننده به آیتم های آرایه شناسه، پلاگین، راه انداز نیاز دارد. همه چیز دیگر اختیاری است.

```
1. {  
2.   id: 'count',  
3.   plug: 'countdown/countdown.js',  
4.   trigger: '.countdown'  
5. },
```

عملکرد و سرعت صفحه

صفحه من به اندازه کافی سریع نیست! میتونی کمک کنی؟

ما با کمال میل صفحه شما را تجزیه و تحلیل خواهیم کرد و نکات و تغییرات دقیقی را که برای استفاده حداکثری از صفحه خود باید انجام دهید، به شما ارائه خواهیم داد. در اینجا لیست کوتاهی از مواردی است که باید در صفحه خود پیاده سازی کنید.

- از [GTmetrix](#) برای تجزیه و تحلیل صفحه خود استفاده کنید تا نتایج دقیقی از کارهایی که برای افزایش امتیاز سرعت باید انجام شود به دست آورید.
- از یک CDN خوب استفاده کنید - CloudFlare (۱۰۰٪ رایگان).
- فایل های Style.css یا SCSS را باز کنید و تمام اجزای استفاده نشده را از محصول خود حذف کنید. آنها به راحتی روی هم چیده می شوند.
- از تصاویر بزرگتر از ۵۰۰ پیکسل استفاده نکنید. هیچ فایده ای ندارد و صفحه را کند می کند.
- تصاویر خود را با گذراندن آنها از طریق [TinyPNG](#) بهینه کنید تا وزن آنها کاهش یابد.
- پس از اتمام ویرایش CSS، آن را با استفاده از [CSSMminifier](#) کوچک کنید.
- پس از اتمام ویرایش JS (فقط custom.js و pwa.js) آن را با استفاده از [JavaScript Minifier](#) کوچک کنید.

لطفا توجه داشته باشید! - PageInsights و Lighthouse Audit اکثر اوقات نتایج نادرستی ارائه می دهند! اینها به شدت به تعداد افراد آزمایش در لحظه ای که شما آزمایشات خود را انجام می دهید بستگی دارد، و Lighthouse به قدرت دستگاه شما بستگی دارد. ما امتیاز ۸۹٪ را در Macbook Air، ۹۳٪ را در Macbook Pro و ۹۹٪ را در iMac به دست آوردیم. برای به دست آوردن یک نتیجه کامل، همراه با نکات عالی، لطفاً حتماً از [GTmetrix](#) استفاده کنید. آنها نتایج بی طرفانه را بر اساس تکنیک های کد واقعی مورد استفاده برای بهترین عملکرد ارائه می دهند. Lighthouse و PageInsights همیشه از محصولات Google حمایت می کنند و به شما امکان می دهند به نتایجی دست یابید (مگر اینکه از AMP استفاده می کنید یا صفحه شما بسیار بسیار کوچک و بدون ویژگی باشد)

چرا از فایل های CSS و JS کوچکتر استفاده نمی کنید؟ و چرا؟

حتی روی کاغذ، دستگاه موبایل شما قدرت یک لپ تاپ را دارد ولی همه آنها از طراحی ریسپانسیو استفاده می کنند و اگر صفحات به درستی بهینه نشده باشند، عملکرد بارگذاری بسیار ضعیفی دارند. داشتن چندین فایل CSS به شما این امکان را می دهد که فایل های JS/CSS خود را به درستی در حال توسعه سازماندهی و گروه بندی کنید. با این حال، این به این معنی است که چندین درخواست HTTP وجود دارد. درخواست های HTTP از نظر زمان بارگذاری گران تر هستند زیرا باید با سرور اتصال برقرار کرده و فایل را فراخوانی کنند.

به یاد داشته باشید، این یک قالب موبایل است، و اگرچه من یا شما می توانیم آیفون XXXXSSS و Galaxy Note ۱۰۰۰ داشته باشیم، دستگاه هایی با پردازنده کمتر قدرتمند وجود دارند. ما می خواهیم فراخوانی ها را به حداقل برسانیم تا سریع ترین زمان بارگیری ممکن را تضمین کنیم. ادغام تمام CSS در یک یا حداکثر ۲ فایل و همه پلاگین های JS در ۱ فایل افزونه و یک فایل فراخوانی بهترین راه برای اطمینان از حداکثر کارایی است، به خصوص که صفحه شما از AJAX استفاده می کند که زمان بارگذاری سریع و قابل اطمینان را تضمین می کند.

گیر کردی؟ جای نگرانی نیست!

اگر مشکلی دارید که در اینجا ذکر نشده است، لطفاً از طریق [تیکت در راست چین](#) یا [وبسایت شخصی](#) بنده در ارتباط باشید. بنده همیشه در دسترس هستیم، در کمتر از ۳ ساعت به همه سوالات شما پاسخ می دهم.

لطفاً با امتیاز دهی به قالب از بنده حمایت کنید