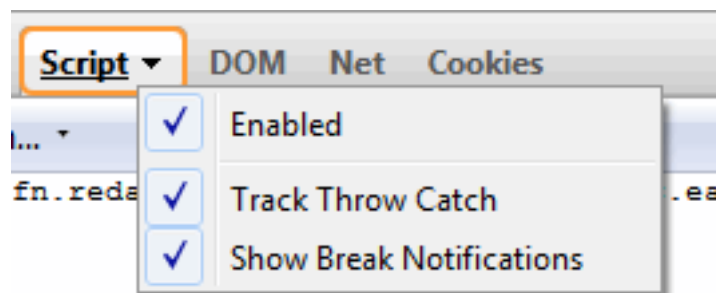


تب Script در FireBug مخصوص دیباگ کردن کدهای JavaScript است. امکاناتی که در این قسمت گنجانده شده بسیار کاربردی بوده و همچنین در بیشتر قسمت‌ها با ابزارهای خطایابی دیگر مشابه است. برای مثال اگر با Visual Studio کار کرده باشید، با نحوه‌ی ایجاد Break Point، قسمت‌های Watch, Stack و ... آشنا خواهید بود.

این پنل هم مشابه پنل‌های دیگر فایر باگ دارای یک بخش با عنوان Options Menu هست که با راست کلیک کردن بروی عنوان یا کلیک بروی مثلث کنار عنوان پنل قابل دسترسی است. تنظیماتی که در اینجا قابل تعیین است عبارتند از:



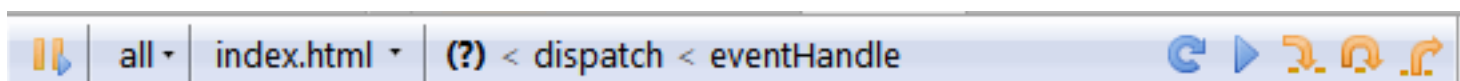
Enabled/Disabled : برای فعال/غیرفعال کردن پنل است. فعال بودن این قسمت ممکن است موجب کاهش سرعت بارگذاری صفحات شود.

Track Throw/Catch : در صورت فعال بودن این گزینه، در صورت رویدادن خطا در بلاک try/catch، دیباگر در آن نقطه از برنامه متوقف شده و کنترل برنامه به شما داده می‌شود. (البته بنده موفق بررسی کامل این قابلیت نشدم. ظاهراً ورژن‌های آخری باگ دارند. مثل غیرفعال شدن کامل همین پنل، "Debugger not activated" اگر کسی موفقیتی در کار با این مورد داشت، سپاسگذار میشوم اطلاع بدهد).

Show Break Notifications : در صورت فعال بودن این گزینه، هنگام توقف کدی در صفحه، اطلاعاتی در مورد علت توقف آن در بالای پنل ارائه خواهد شد.

توجه داشته باشید که ممکن است در ورژن‌های مختلف، تعداد این گزینه‌ها بیشتر یا کمتر باشد.

Panel Toolbar



نواری است که ابزارهای پنل بروی آن قرار دارند و به ترتیب عبارتند از:

Break On Next : این دکمه که مشابه آن در اکثر پنل‌ها وجود دارد، هنگام اجرای یک دستور JavaScript آن را متوقف کرده و شما می‌توانید به بررسی آن بپردازید.

Script Type Menu : با انتخاب یکی از چهار گزینه موجود می‌توانید نتیجه اسکریپت‌های اضافه شده به صفحه که در قسمت Script Location Menu نمایش داده شده اند را فیلتر کنید. (متأسفانه این گزینه هم مشابه گزینه Track Throw/Catch برای بنده،

ورژن 1.11.4 نتیجه ای نداشته است.)

Script Location Menu : در این قسمت اسکرپت هایی که در صفحه وجود دارند نمایش داده می شوند. مشابه [پنل HTML](#) می توانید هنگام بازبودن این لیست، با تایپ کردن نتایج را فیلتر کنید.

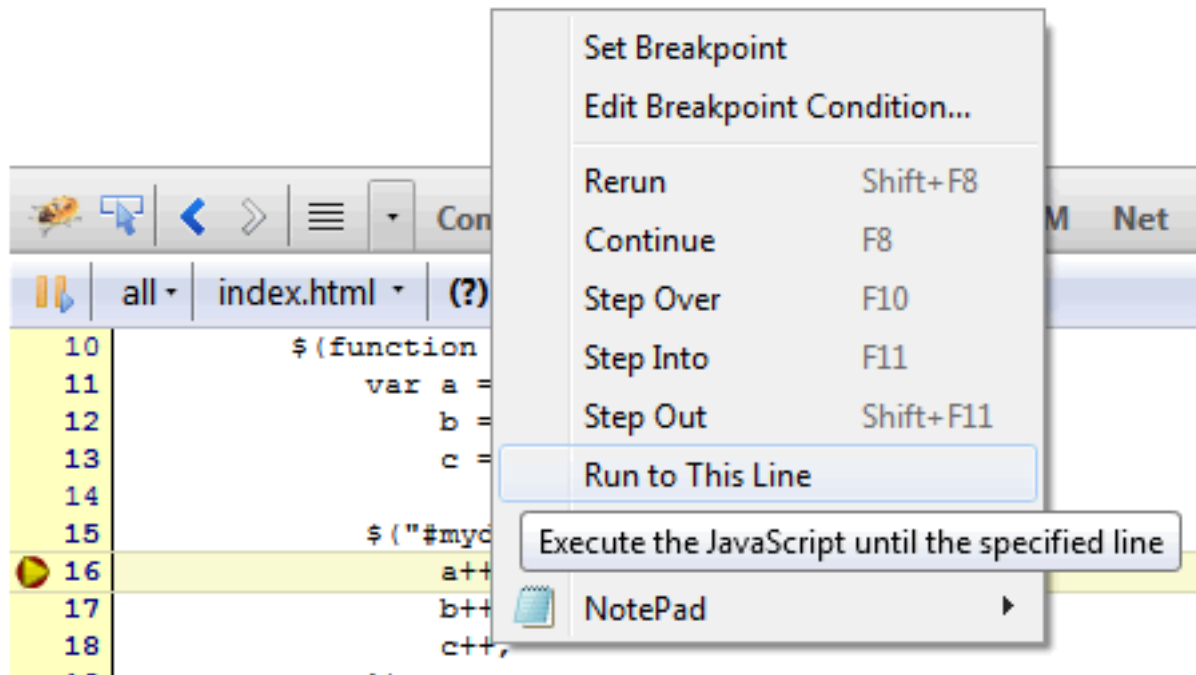
همچنین با راست کلیک کردن بروی این قسمت می توانید آیتم یا فایل انتخاب شده را در ادیتور مورد نظر باز کنید، در پنل DOM بررسی کنید، فایل را در یک تب جدید باز کنید، آدرس فایل را در حافظه موقت کپی کنید.

Execution Control Buttons : این دکمه ها زمانی که دیباگر به یک Break Point برسد یا به دلیل فعال بودن دکمه های Break On ... متوقف شود، فعال می شوند و با کمک آن ها می توانید عملیات خطایابی را ادامه دهید.
در جدول زیر این دکمه ها و توضیحات تکمیلی هریک را مشاهده می کنید:

نوع	دکمه	Shortcut	توضیحات
اجرای مجدد		Shift + F18	Stack فعلی را مجدد اجرا می کند. (Stack: لیستی از توابع به ترتیبی که با فراخوانی آنها تابع جاری فراخوانی شده است. در مقاله بعدی توضیحات بیشتری ارائه خواهد شد.)
ادامه		F8	اجرای برنامه را (تا Break Point بعدی) ادامه می دهد.
وارد شدن		F11	در صورت رسیدن به یک تابع، با زدن این دکمه دیباگر وارد بندهای آن تابع می شود.
رد شدن		F10	اجرای برنامه را در سطح (Scope) فعلی ادامه می دهد. مشابه قبلی وارد تابع نمی شود.
خارج شدن		Shift + F11	کنترل به تابعی که تابع جاری را فراخوانی کرده است باز میگردد. روشی سودمند زمانی که هنگام خطایابی وارد کدهای jQuery یا مثل آن می شوید (:

بعد از Script Location Menu قسمتی وجود دارد که وضعیت فعلی Stack را نشان می دهد و با کلیک بروی هرکدام، به کد آن قسمت هدایت می شوید. (البته اگر فایرباگ شما در این قسمت باگ نداشته باشد!)

تنها قابلیت جدیدی که در این منو وجود دارد، Run To This Line است. هنگامی که پنل در حالت دیباگ است، با راست کلیک کردن بروی خط مورد نظر و انتخاب گزینه‌ی Run to This Line می‌توانید خطوط میانی را رد کرده و به خط مورد نظر بروید. البته خطوطی که رد می‌شوند، اجرا می‌شوند. این کار معادل این است که در خط مورد نظر یک Break Point اضافه کنید و دکمه‌ی F8 را بزنید.



Breakpoints

توسط Break Point می‌توانید خطوطی از برنامه را برای خطایابی مشخص کنید. زمانی که دیباگر به نقطه‌ی مورد نظر برسد متوقف شده و می‌توانید به بررسی برنامه بپردازید. می‌توانید با بردن موس بروی متغیرها مقدار آن‌ها را بررسی کنید یا با کمک قسمت‌های Watch و Stack در پنل جانبی اطلاعات بیشتری در مورد اجرای برنامه در نقطه‌ی جاری بدست آورید. (در مورد پنل‌های جانبی در قسمت بعدی توضیح خواهیم داد.) همچنین با کمک دکمه‌هایی که در جدول فوق توضیح داده شده اند روند اجرای برنامه را خط به خط ادامه دهید.

برای ایجاد یک Break Point، بروی شماره‌ی خط مورد نظر کلیک کنید. نقطه‌ای قرمز رنگ نمایش داده خواهد شد. البته دقت کنید که همه‌ی خطوط برنامه اجرا نمی‌شوند و نمی‌توانید در آن خطوط از این امکان بهره ببرید. شماره‌ی خطوط فعال با رنگ سبز مشخص شده اند.

امکان مفید دیگری که همراه با این قابلیت ارائه شده است (که در محیط‌های دیگر هم وجود دارد)، عبارت شرطی است. به این صورت که ضمن قرار دادن Break Point در یک نقطه از برنامه، می‌توانید یک شرط هم برای توقف برنامه تعیین کنید. فرض کنید یک حلقه دارید که 300 بار تکرار می‌شود و مثلاً در اجرای 250ام آن مشکلی وجود دارد. در این حالت می‌توانید از این قابلیت استفاده کنید و شرط توقف را `i == 250` قرار بدهید.

برای تعیین یک شرط برای یک Break Point، بروی خط مورد نظر راست کلیک کنید و شرط را در قسمت مشخص شده وارد کنید.



امکان مفید دیگری که وجود دارد، Break Point خودکار است. اگر از مقالات قبلی دکمه‌ی Break On All Errors در پنل Console و Break On Mutate در پنل HTML را بخاطر داشته باشید می‌دانید که در هر یک هنگام اجرای یک رخداد مورد نظر، دیباگر در خطی که موجب آن رخداد شده است متوقف شده و می‌توانید کنترل برنامه را بدست بگیرید. در این حالت نیازی به ایجاد Break Point نیست و FireBug بصورت خودکار این کار را انجام می‌دهد.

Search

این بخش در همه پنل‌ها وجود دارد با این تفاوت که در پنل Script و CSS دو گزینه‌ی *Use Regular* و *Multiple Files* وجود دارد که به ترتیب امکان جستجو در فایل‌های js و css اضافه شده به صفحه هستند. قابلیت دیگری هم که فقط در پنل Script وجود دارد، پرش به یک خط در برنامه است به این صورت که با وارد کردن # و یک عدد به عنوان شماره‌ی خط، همان خط نمایش داده می‌شود.



در [قسمت بعدی](#) پنل جانبی که شامل سه بخش Watch، Stack و Breakpoints است را بررسی خواهیم کرد.