

در ادامه مطلب قبلی [آموزش \(jQuery\) جی کوئری #3](#) به ادامه بحث می‌پردازیم.

با توجه به حالت‌های مختلف و گزینه‌های گوناگونی که انتخاب‌کننده‌ها در اختیار ما گذاشته اند، اگر هنوز دنبال قدرت بیشتری از انتخاب‌کننده‌ها هستید در ادامه به چند مورد از آنها اشاره خواهیم کرد.

3-1- انتخاب عناصر بر اساس موقعیت

گاهی اوقات انتخاب عناصر با توجه به مکان آنها و یا موقعیت مکانی آنها نسبت به سایر اجزا صورت می‌پذیرد؛ برای مثال اولین لینک صفحه و یا اولین لینک هر پاراگراف و یا گزینه‌ی آخر از لیست، jQuery شیوه‌ای خاص را برای چنین انتخاب‌هایی ارائه کرده است. برای مثال دستور زیر اولین لینک موجود در صفحه را انتخاب می‌کند:

```
a:first
```

دستور زیر چکاری انجام می‌دهد؟

```
p:odd
```

دستور بالا تمامی پاراگراف‌های فرد را انتخاب می‌کند. روش‌های دیگری هم ممکن است بخواهیم استفاده کنیم؛ مثلاً دستور زیر تمامی پاراگراف‌های زوج را انتخاب می‌کند:

```
p:even
```

یا با استفاده از دستور زیر میتوان آخرین فرزند یک والد را انتخاب کرد؛ در زیر آخرین فرزند یک انتخاب می‌شود. علاوه بر انتخاب‌کننده‌هایی که ذکر شد؛ تعداد قابل توجه دیگری نیز وجود دارند که در جدول 2-2 ذکر شده اند.

جدول 2-2: انتخاب گرهای پیشرفته موقعیت عناصر که توسط jQuery پشتیبانی می‌شوند

توضیح	فیلتر
اولین عنصر که با شرط ما مطابقت می‌کند را انتخاب می‌کند، li a:first اولین لینکی را که فرزند لیست به حساب می‌آیند؛ را بر می‌گرداند	:first
آخرین عنصری که با شرط ما مطابقت کند را انتخاب می‌کند. li a:last آخرین لینک از فرزندان لیست را برمی‌گرداند.	:last
اولین فرزند عنصر که با شرط ما مطابقت می‌کند را انتخاب می‌کند. li a:first-child اولین عنصر لینک از هر لیست را برمی‌گرداند.	:first-child
آخرین فرزند عنصر که با شرط ما مطابقت می‌کند را انتخاب می‌کند. li a:last-child اولین عنصر لینک از هر لیست را	:last-child

توضیح	فیلتر
برمی گرداند.	
تمام عناصری که پدر آنها تنها همان فرزند را داد، برمی گرداند.	:only-child
n امین فرزند عنصری که با شرط ما مطابقت داشته باشد را انتخاب می کند. <code>li:nth-child(2) //comment</code> دومین عنصر از هر لیست را برمی گرداند.	:nth-child(n)
فرزندان زوج یا فرد عنصر را انتخاب می کند. <code>li:nth-child(even) //comment</code> تمام عناصر زوج لیست ها را برمی گرداند.	:nth-child(even یا odd)
n امین فرزند عنصری که از طریق فرمول ارایه شده به دست می آید را انتخاب می کند. اگر ۷ صفر باشد، نیازی به نوشتن آن نیست. <code>li:nth-child(3n) //comment</code> تمام عناصر ضریب 3 لیست ها را بر می گرداند، در حالی که <code>li:nth-child(5n+1) //comment</code> عناصری از لیست را برمی گرداند که بعد از عنصرهای ضریب 5 لیست ها قرار گرفته باشند.	:nth-child(Xn+Y)
تمام عناصر زوج یا فرد که با شرط ما مطابقت کنند را انتخاب می کند. <code>li:even</code> تمامی عناصر زوج لیست ها را بر می گرداند.	:even یا :odd
n امین عنصر انتخاب شده را برمی گرداند.	:eq(n)
عناصر بعد از n امین عنصر را بر می گرداند. (در واقع عناصری که بزرگتر از عنصر n ام هستند را بر می گرداند)	:gt(n)
عناصر قبل از n امین عنصر را بر می گرداند. (در واقع عناصری که کوچکتر از عنصر n ام هستند را بر می گرداند)	:lt(n)

پ.ن: در جدول بالا در توضیحات بعضی از انتخاب گر ها `//comment` نوشته شده است، اینها جز دستور نبوده و فقط برای نمایش صحیح پرانتز در صفحه اینترنتی نوشته شده است، در عمل نیازی به اینها نیست.

نکته ای که در مورد انتخاب گره های جدول بالا وجود دارد این است که در فیلتر `:nth-child` برای سازگاری با CSS، مقدار شمارشگر از 1 آغاز می شود، اما در سایر فیلترها از قاعده ای که اکثر زبانهای برنامه نویسی استفاده شده است و شمارشگر آنها از صفر آغاز می شود. برای درک این موضوع مثال زیر را در نظر بگیرید:

```
<table id="languages">
  <thead>
    <tr>
      <th>Language</th>
      <th>Type</th>
      <th>Invented</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Java</td>
      <td>Static</td>
      <td>1995</td>
    </tr>
    <tr>
      <td>Ruby</td>
      <td>Dynamic</td>
      <td>1993</td>
    </tr>
    <tr>
      <td>Smalltalk</td>
```

```

        <td>Dynamic</td>
        <td>1972</td>
    </tr>
    <tr>
        <td>C++</td>
        <td>Static</td>
        <td>1983</td>
    </tr>
</tbody>
</table>

```

حال می‌خواهیم از این جدول، محتویات تمام خانه‌هایی که نام یک زبان برنامه نویسی در آنهاست را انتخاب نماییم. از آنجا که نام این زبانها در اولین ستون از هر سطر قرار دارد. می‌توانیم دستوری مانند زیر بنویسیم:

```
table#languages tbody td:first-child
```

و یا با استفاده از دستور زیر این کار را انجام دهیم:

```
table#languages tbody td:nth-child(1)
```

اما دستور اول مختصرتر و خواناتر است. پس از آن برای دسترسی به نوع هریک از زبانهای برنامه نویسی، دستور انتخاب کننده دوم را به صورت

```
:nth-child(2)
```

تغییر می‌دهیم، و همچنین با تغییر پارامتر 2 به 3 سالی که هر یک از زبانها ابداع شده اند، انتخاب می‌شوند. بدیهی است در این حالت دو دستور

```

:nth-child(3)
یا
:last-child

```

با یکدیگر برابرند. اما هردوی آنها ستون آخر از هر سطر را انتخاب می‌کنند، در شرایطی که بخواهیم آخرین خانه جدول انتخاب شود (خانه ای با مقدار 1983)، از **td:last** استفاده می‌کنیم. توجه کنید در حالی که دستور **td:eq(2) // comment** خانه ای با مقدار 1995 را انتخاب می‌کند، دستور **td:nth-child(2) // comment** تمام خانه‌های بیان کننده نوع زبانها را انتخاب می‌کند. بنابراین به خاطر داشته باشید که مقدار ابتدایی شمارشگر فیلتر **eq** از صفر است و این مقدار برای فیلتر **nth-child** یک تعیین شده است.

در پست بعدی انتخاب گره‌های CSS و فیلترهای سفارشی jQuery را بررسی خواهیم کرد.