

性能监控可以很好的帮助开发者了解在各种真实环境下，页面的性能情况是如何的。对于性能监控来说，我们可以直接使用浏览器自带的[Performance API](#)来实现这个功能。

对于性能监控来说，其实我们只需要调用

`performance.getEntriesByType('navigation')`这行代码就行了。对，你没看错，一行代码我们就可以获得页面中各种详细的性能相关信息。

```
> performance.getEntriesByType('navigation')
< ▼ [PerformanceNavigationTiming] ⓘ
  ▼ 0: PerformanceNavigationTiming
    connectEnd: 14.159999991534278
    connectStart: 14.159999991534278
    decodedBodySize: 17583
    domComplete: 3008.5649999964517
    domContentLoadedEventEnd: 1379.350000002887
    domContentLoadedEventStart: 1035.8849999902304
    domInteractive: 1035.8499999856576
    domainLookupEnd: 14.159999991534278
    domainLookupStart: 14.159999991534278
    duration: 3010.3399999788962
    encodedBodySize: 11548
    entryType: "navigation"
    fetchStart: 14.159999991534278
    initiatorType: "navigation"
    loadEventEnd: 3010.3399999788962
    loadEventStart: 3008.5850000032224
    name: "https://juejin.im/"
    nextHopProtocol: ""
    redirectCount: 0
    redirectEnd: 0
    redirectStart: 0
    requestStart: 25.439999997615814
    responseEnd: 102.87499998230487
    responseStart: 54.64499999652617
    secureConnectionStart: 0
    ▶ serverTiming: []
      startTime: 0
      transferSize: 0
      type: "navigate"
      unloadEventEnd: 0
      unloadEventStart: 0
      workerStart: 0
```

我们可以发现这行代码返回了一个数组，内部包含了相当多的信息，从数据开始在网络中传输到页面加载完成都提供了相应的数据。