



Soheib Kiani
Front-end Developer

تفاوت

e.preventDefault()

e.stopPropagation()

رجاوا اسکریپت





Soheib Kiani
Front-end Developer

کاربرد اصلی این متدها مربوط به **Browser events**

هستنس

این متدها به ترتیب برای متوقف کردن عملکرد
پیش‌فرض رویداد و متوقف کردن انتشار
رویداد به المان‌های دیگر درخت DOM استفاده
می‌شوند.





Soheib Kiani
Front-end Developer

وظایف هر کدام چیه؟

e.preventDefault()

وظیفش اینکه عملکرد پیش‌فرض رویداد را لغو کنه. به عبارت دیگر، با فراخوانی این متده، رویداد اصلی که باید در پاسخ به عملکرد طبیعی المان اجرا می‌شد، کنسل بشه و هیچ اقدامی انجام نشه

e.stopPropagation()

وظیفش اینکه انتشار رویدار به المان‌های دیگر درخت DOM را متوقف کنه. به عبارت دیگر، با فراخوانی این متده، رویدار به المان‌های والد و پدربرزگ و ... ارسال نمی‌شه و فقط در المان فعلی پردازش می‌شه





Soheib Kiani
Front-end Developer

مثال تئوری هم بینیم

e.preventDefault()

فرض کنید که یک لینک دارید و می‌خواهید با کلیک بر روی آن، صفحه به صورت پیش‌فرض جابجا نشود. به جای آن می‌خواهید یک اقدام خاص دیگر (برای مثال، نمایش پنجره‌ای) را انجام دهید. در این حالت، با فراخوانی `e.preventDefault()` در رویداد کلیک، عملکرد پیش‌فرض جابجایی صفحه لغو می‌شود و پنجره‌ای که می‌خواهید نمایش داده شود، نمایش داده می‌شود.

e.stopPropagation()

فرض کنید یک المان دارید که دارای المان‌های فرزندی هستش. شما می‌خواهید رویداد کلیک روی المان فرزندی خاصی را پردازش کنید و اما نمی‌خواهید که رویداد به سایر المان‌های فرزندی یا والد پردازش بشه. با فراخوانی `e.stopPropagation()` در رویداد کلیک، رویداد تنها در المان فرزندی خاص پردازش می‌شه و به سایر المان‌ها ارسال نمی‌شه.





Soheib Kiani
Front-end Developer

اصلیت بخوبی توضیح دارم

مثال کدی هم بینیم

...

html

```
<div id="parent">  
  <button id="child">Click me</button>  
</div>
```

...

JS

```
const parent = document.getElementById('parent');  
const child = document.getElementById('child');  
  
parent.addEventListener('click', function(e) {  
  console.log('Parent clicked');  
});  
  
child.addEventListener('click', function(e) {  
  e.stopPropagation();  
  console.log('Child clicked');  
});
```





Soheib Kiani
Front-end Developer

در این مثال، وقتی روی المان "child" کلیک میکنی، رویداد کلیک پردازش میشه و پیام "Child clicked" در کنسول نمایش داده میشه. اما اگر نکنیم، پیام "Child clicked" در کنسول نمایش داده میشه و سپس پیام "Parent clicked" نیز در کنسول نمایش داده میشه چون رویداد به المان والد ارسال میشه. با فراخوانی `e.stopPropagation()`، ارسال رویداد به المان والد متوقف میشه و تنها رویداد در المان "child" پردازش میشه



JS

```
const parent = document.getElementById('parent');
const child = document.getElementById('child');

parent.addEventListener('click', function(e) {
  console.log('Parent clicked');
});

child.addEventListener('click', function(e) {
  e.stopPropagation();
  console.log('Child clicked');
});
```





Soheib Kiani
Front-end Developer

اگر برات مفید بود

ممنون میشم به اشتراک بذاریش

