

نمایش بلادرنگ اعلامی به تمام کاربران در هنگام درج یک رکورد جدید به صورت notification

عنوان:

سیروان عقیفی

نویسنده:

۱۳:۴۵ ۱۳۹۳/۱۰/۱۲

تاریخ:

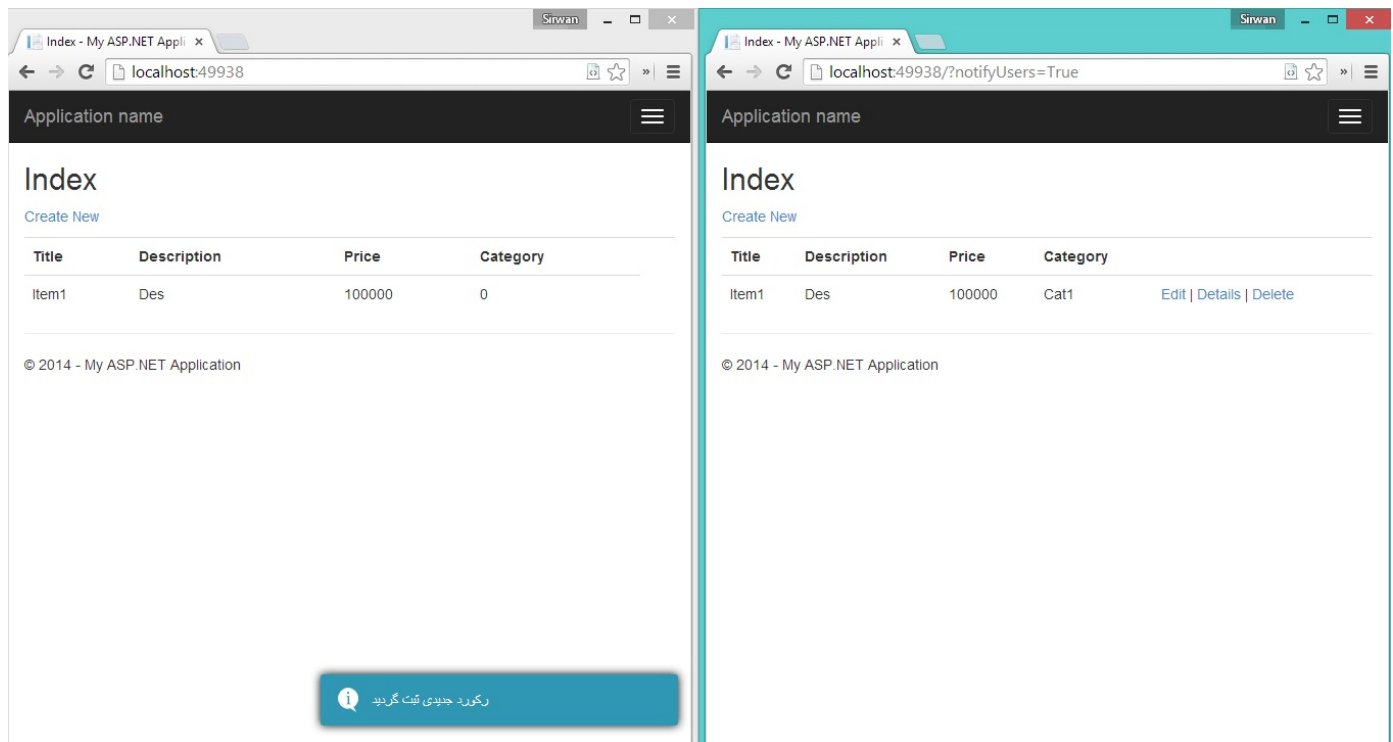
www.dotnettips.info

آدرس:

MVC, SignalR

گروه‌ها:

در ادامه می‌خواهیم مثالی را که در [این مطلب](#) مورد بررسی قرار گرفت، به صورتی تغییر دهیم که با ثبت یک آیتم جدید درون دیتابیس، یک notification، به تمامی کاربران متصل به هاب ارسال شود. همچنین با کلیک بر روی Notification سطر جدید نیز بلافاصله نمایش داده شود:



در این مثال برای نمایش پیام به صورت notification، از کتابخانه [toastr](#) استفاده می‌کنیم که از طریق nuget می‌توانید آن را به پروژه اضافه کنید:

```
PM> Install-Package toastr
```

کار با این کتابخانه خیلی ساده است؛ کافی است فایل‌های js و css آن را به فایل layout اضافه کرده و به این صورت از آن استفاده کنیم:

```
toastr.info("نمایش یک پیام - info");
toastr.success("نمایش یک پیام - success");
toastr.error("نمایش یک پیام - error");
toastr.warning("نمایش یک پیام - warning");
```

دستورات فوق خروجی‌های زیر را نمایش می‌دهد:



برای پیام‌های فوق نیز می‌توانید عنوانی را انتخاب کنید:

```
toastr.success("نمایش یک پیام - success", "عنوان");
```

اگر به فایل js این کتابخانه مراجعه کنید، می‌توانید مقادیر پیش‌فرض آن را برای نمایش یک پیام مشاهده کنید. برای سفارشی‌سازی آن نیز می‌توانید به این صورت عمل کنید:

```
toastr.options = {
  tapToDismiss: true,
  toastClass: 'toast',
  containerId: 'toast-container',
  debug: false,

  showMethod: 'fadeIn', //fadeIn, slideDown, and show are built into jQuery
  showDuration: 300,
  showEasing: 'swing', //swing and linear are built into jQuery
  onShown: undefined,
  hideMethod: 'fadeOut',
  hideDuration: 1000,
  hideEasing: 'swing',
  onHidden: undefined,

  extendedTimeOut: 1000,
  iconClasses: {
    error: 'toast-error',
    info: 'toast-info',
    success: 'toast-success',
    warning: 'toast-warning'
  },
  iconClass: 'toast-info',
  positionClass: 'toast-top-right',
  timeout: 5000, // Set timeout and extendedTimeOut to 0 to make it sticky
  titleClass: 'toast-title',
  messageClass: 'toast-message',
  target: 'body',
  closeHtml: '<button>&times;</button>',
  newestOnTop: true,
  preventDuplicates: false,
  progressBar: false
};
```

اکنون برای نمایش این نوع پیام‌ها در زمان اتصال به هاب (در واقع در زمان ثبت یک رکورد جدید) نیاز به ارسال پارامتر خاصی به

سرور (از سمت کلاینت) نمی‌باشد. تنها باید کدهای سمت سرور یعنی هاب را به گونه‌ای تغییر دهیم تا به محض فراخوانی SendNotification، آخرین رکورد ثبت شده در دیتابیس را به تمامی کلاینت‌های متصل به هاب ارسال کند:

```
public class NotificationHub : Hub
{
    private readonly IProductService _productService;

    public NotificationHub(IProductService productService)
    {
        _productService = productService;
    }

    public void SendNotification()
    {
        Clients.Others.ShowNotification(_productService.GetLastProduct());
    }
}
```

در سمت کلاینت نیز کدها همانند مثال قبل هستند؛ با این تفاوت که در متد سمت کلاینت باید اطلاعات ارسال شده از سمت سرور را با نمایش یک notification به کاربران اطلاع دهیم:

```
var notify = $.connection.notificationHub;
notify.client.showNotification = function (data) {
    toastr.info("رکورد جدیدی ثبت گردید جهت نمایش اینجا کلیک کنید");
};
$.connection.hub.start().done(function () {
    @({
        if (ViewBag.NotifyUsers)
        {
            <text>notify.server.sendNotification();</text>
        }
    });
});
```

تا اینجا همانند مثال قبلی عمل کردیم. یعنی به جای نمایش یک alert بوت‌استرپ، از کتابخانه toastr استفاده کردیم. در مثال قبلی کاربر برای دیدن تغییرات می‌بایستی یکبار صفحه را ریفرش کند، اکنون می‌خواهیم کاربر بعد از کلیک بر روی پیام، بلافاصله سطر جدید را نیز مشاهده کند:

```
var positionClasses = {
    topRight: 'toast-top-right',
    bottomRight: 'toast-bottom-right',
    bottomLeft: 'toast-bottom-left',
    topLeft: 'toast-top-left',
    topCenter: 'toast-top-center',
    bottomCenter: 'toast-bottom-center'
};
var notify = $.connection.notificationHub;
notify.client.showNotification = function (data) {
    toastr.options = {
        showDuration: 300,
        positionClass: positionClasses.bottomRight,
        onclick: function () {
            $('#table tr:last').after("<tr>" +
                "<td>" + data.Title + "</td>" +
                "<td>" + data.Description + "</td>" +
                "<td>" + data.Price + "</td>" +
                "<td>" + data.Category + "</td>" +
                "<td> </td>" +
                "</tr>");
        }
    };
    toastr.info("رکورد جدیدی ثبت گردید جهت نمایش اینجا کلیک کنید");
};
$.connection.hub.start().done(function () {
    @({
        if (ViewBag.NotifyUsers)
        {
            <text>notify.server.sendNotification();</text>
        }
    });
});
```

```
});
```

همانطور که مشاهده می‌کنید از onClick برای toastr استفاده کرده‌ایم. با این callback گفته‌ایم که اگر بر روی پیام کلیک شد، اطلاعات را به صورت یک سطر جدید به جدول اضافه کن:

```
onclick: function () {  
    $('#table tr:last').after("<tr>" +  
        "<td>" + data.Title + "</td>" +  
        "<td>" + data.Description + "</td>" +  
        "<td>" + data.Price + "</td>" +  
        "<td>" + data.Category + "</td>" +  
        "<td> </td>" +  
        "</tr>");  
}
```

مقادیر به صورت یک شیء جاوااسکریپتی برگردانده خواهند شد:

```
data {Id: 12, Title: "Item1", Description: "Des", Price: 100000, Category: 0}
```

که توسط data می‌توانیم به هر کدام از فیلدها، جهت نمایش در خروجی، دسترسی داشته باشیم. دریافت سورس مثال جاری :

[ShowAlertSignalR](#)

نظرات خوانندگان

نویسنده: محمد رعیت پیشه
تاریخ: ۱۷:۱۱ ۱۳۹۳/۱۰/۱۲

نظرتون درباره استفاده از SqlDependency برای اینکار چیه؟ فکر می‌کنید توی یک پروژه بزرگ استفاده از کدامیک بهتر است؟

نویسنده: محسن خان
تاریخ: ۲۱:۴۵ ۱۳۹۳/۱۰/۱۲

SqlDependency محدود هست به SQL Server ضمن اینکه تنظیم آن هم فقط در سمت برنامه نیست. یعنی شخص استفاده کننده باید دسترسی مدیریتی به سرور SQL داشته باشه تا بتونه اون رو تنظیم کنه. ولی زمانیکه با یک ORM کار می‌کنید، تا زمانیکه از API همون ORM استفاده می‌کنید، با عوض کردن پروایدر اون، می‌تونید یک روز از SQL Server و یک روز از اوراکل یا سایر بانک‌های اطلاعاتی استفاده کنید. اینجا است که عدم نیاز به دسترسی مدیریتی و همچنین عمومی‌تر شدن راه حل یک مزیت مهم خواهد شد.

نویسنده: محمد رعیت پیشه
تاریخ: ۲۱:۵۳ ۱۳۹۳/۱۰/۱۲

اگر بخواهیم در یک ویندوز سرویس متوجه تغییرات بشیم چطور؟ یعنی هیچ واسط دیگه ای وجود نداشته باشه

نویسنده: محسن خان
تاریخ: ۲۲:۳۸ ۱۳۹۳/۱۰/۱۲

SignalR محدود به وب نیست: [نگاهی به گزینه‌های مختلف مهبای جهت میزبانی SignalR](#) (در مورد سرور) و کلاینت دات نتی هم می‌تونه داشته باشه: [نگاهی به SignalR Clients](#). حتی کلاینت جاوایی هم می‌تونه داشته باشه: [استفاده از SignalR در اندروید](#)