



آموزش پروژه محور Kendo CRUD

در این آموزش، قصد داریم تا نحوه انجام عملیات (Create, Retrieve, Update, Delete) CRUD را نمایش دهیم و علاوه بر آن نحوه مدیریت خطاهای سمت سرور را بررسی می کنیم و آن ها را در سرور نمایش می دهیم. برای شروع، ترجیح می دهیم اطلاعات بیشتری از پایگاه داده واکنشی کنیم. فیلدهای City و BirthDate و Title را به شیء Employee اضافه کنید. همچنین یک سازنده ایجاد کنید که یک شیء Employee دیتا را به عنوان ورودی می گیرد و سپس آن را به خصوصیت های مدل نگاشت می کند.

حال شیء مدل Employee به صورت زیر در می آید:

```
public class Employee {  
  
    public string FirstName { get; set; }  
    public string LastName { get; set; }  
    public string Title { get; set; }  
    public DateTime BirthDate { get; set; }  
    public string City { get; set; }  
  
    public Employee(hello_kendo_ui.Data.Employee employee) {  
        this.Id = employee.EmployeeID;  
        this.FirstName = employee.FirstName;  
        this.LastName = employee.LastName;  
        this.Title = employee.Title;  
        this.BirthDate = employee.BirthDate;  
        this.City = employee.City;  
    }  
}
```

متد Get کلاسی EmployeesController

Code

```
// WebAPI will respond to an HTTP GET with this method  
public Models.Response Get() {  
  
    // the the take and skip parameters off of the incoming request  
    int take = _request["take"] == null ? 10 : int.Parse(_request["take"]);  
    int skip = _request["skip"] == null ? 0 : int.Parse(_request["skip"]);  
  
    // get all of the records from the employees table in the  
    // northwind database. return them in a collection of user  
    // defined model objects for easy serialization. skip and then  
    // take the appropriate number of records for paging.  
    var employees = (from e in _context.Employees
```

```

        select new
Models.Employee(e) .Skip(skip) .Take(take) .ToArray();

// returns the generic response object which will contain the
// employees array and the total count
return new Models.Response(employees, _context.Employees.Count());
}

```

تغییر در پیگر بندی گرید برای نمایش صفحه بندی

در نهایت باید تنظیماتی را به پیگر بندی Grid اضافه کنیم تا صفحه بندی سمت سرور تکمیل گردد

توضیح داده شده تمام خصوصیت های مربوط به تنظیم پیگر بندی سمت سرور و اتصال به پایگاه داده در [آموزش اتصال به پایگاه داده](#) است

پیگر بندی Grid برای صفحه بندی سمت سرور

```

<script >

$(function () {
    $("#employeesGrid").kendoGrid({
        columns: [
            { field: "FirstName", title: "First Name" },
            { field:"LastName", title: "Last Name" }
        ],
        dataSource: new kendo.data.DataSource({
            transport: {
                read: "api/employees",
            },
            pageSize:3,
            serverPaging: true,
            schema: {
                data: "Data",
                total: "Count"
            }
        }),
        pageable: true
    });
});
</script>

```

پیگر بندی Grid برای نمایش ستون های

```

<script>
$(function () {
    $("#employeesGrid").kendoGrid({
        columns: [
            { field: "FirstName", title: "First Name" },
            { field:"LastName", title: "Last Name" },
            "Address",
            "City",
            { field: "BirthDate", title: "Birthday", format: "{0:MM/dd/yyyy}"
        }
    }

```

```
],
dataSource: new kendo.data.DataSource({
  transport: {
    read: "api/employees",
  },
  pageSize:3,
  serverPaging: true,
  schema: {
    data: "Data",
    total: "Count"
  },
}),
pageable: true,
editable:true
});
</script>
```

گروه آموزشی طلوع


tolooco.ir
cwxlfi.com