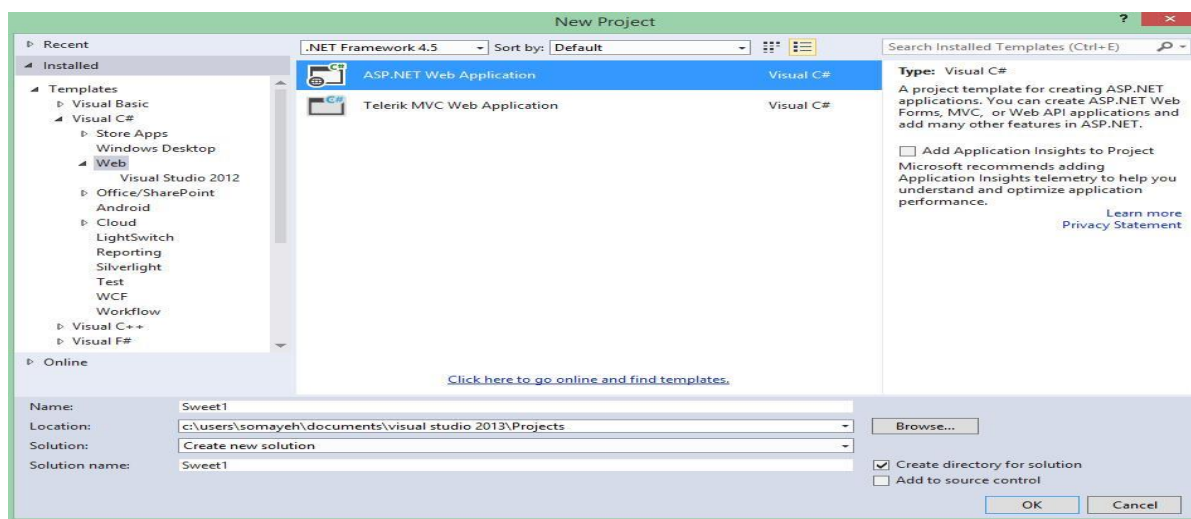
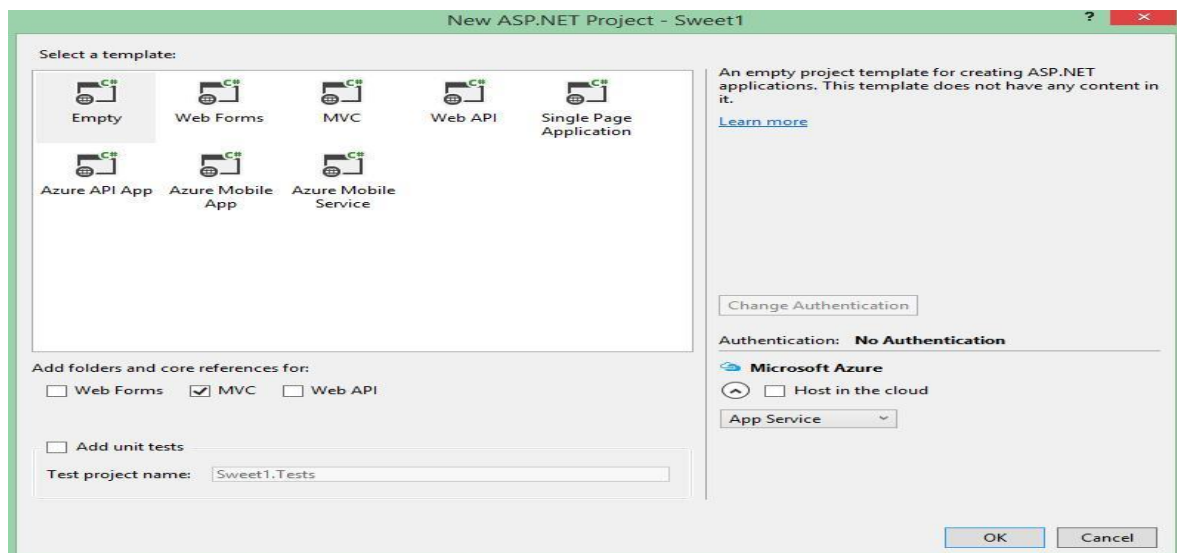


پیغام هایی که توسط جاوااسکریپت داده می شوند به نظر ابتدایی می آیند و زیبایی خاصی ندارند ، در این مقاله نحوه پیاده سازی Sweet Alert در MVC و در جاهای مختلف ، مثلا در هنگام حذف یک رکورد و یا در هنگام Validation های سیستم را آموزش خواهیم داد.

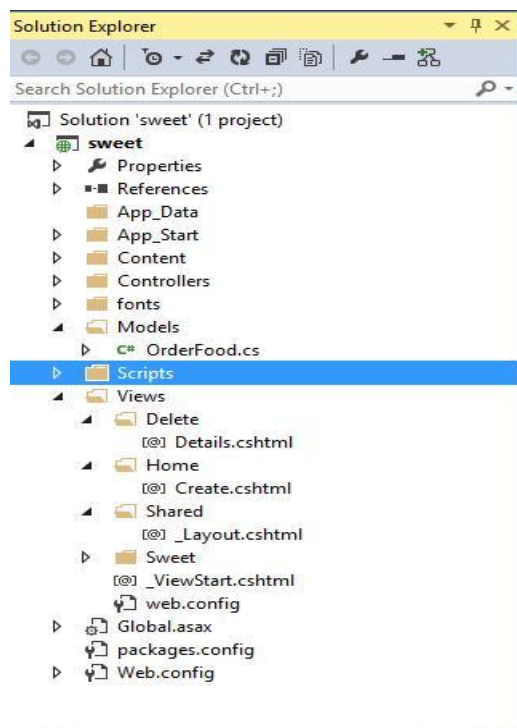


پیغام هایی که توسط جاوااسکریپت داده می شوند به نظر ابتدایی می آیند و زیبایی خاصی ندارند ، در این مقاله نحوه پیاده سازی Sweet Alert در MVC و در جاهای مختلف ، مثلا در هنگام حذف یک رکورد و یا در هنگام Validation های سیستم را آموزش خواهیم داد. ابتدا یک Application از نوع mvc و از مدل Empty ایجاد می کنیم





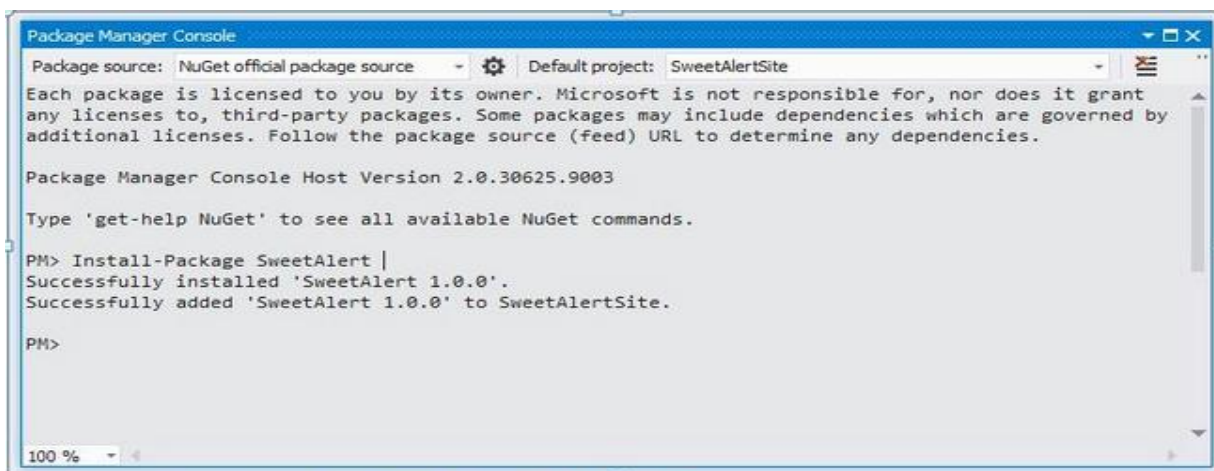
حال شکل پوشه ها در داخل پروژه ما به صورت زیر است



به مسیر `Tools->NugetPackageManager->PackageManagerConsol` بروید و بعد از نوشتن دستور زیر کلید اینتر را فشار دهید

Install-Package SweetAlert

بعد از موفقیت آمیز بودن اجرای این دستور Solution Explorer به شکل زیر خواهد بود



حال بر روی پوشه مدل خود کلیک راست کرده و یک کلاس به نام OrderFood اضافه کنید

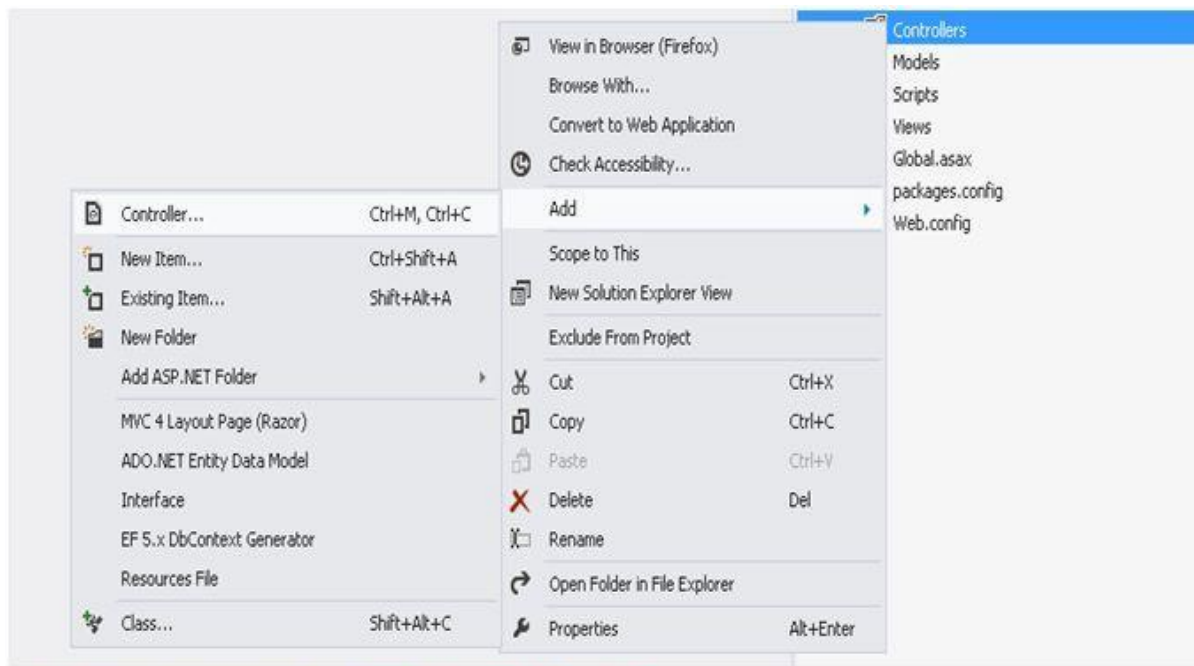
کد این کلاس به صورت زیر است

```

public class OrderFood
{
    [Key]
    public int OrderID
    {
        get;
        set;
    }
    [Required(ErrorMessage = "Please enter OrderCode")]
    public string OrderCode
    {
        get;
        set;
    }
}
  
```

```
[Required(ErrorMessage = "Please enter OrderPrice")]
public string OrderPrice
{
    get;
    set;
}
```

یک کنترلر به نام Home ایجاد کنید این کنترلر از نوع Empty است



بعد از افزودن یک کنترلر از نوع Empty فقط یک action به نام ایندکس در داخل آن قرار دارد Action . های زیر را به پروژه اضافه کنید

```
public ActionResult Index()
{
    return View();
}
```

```
[HttpGet]
```

```

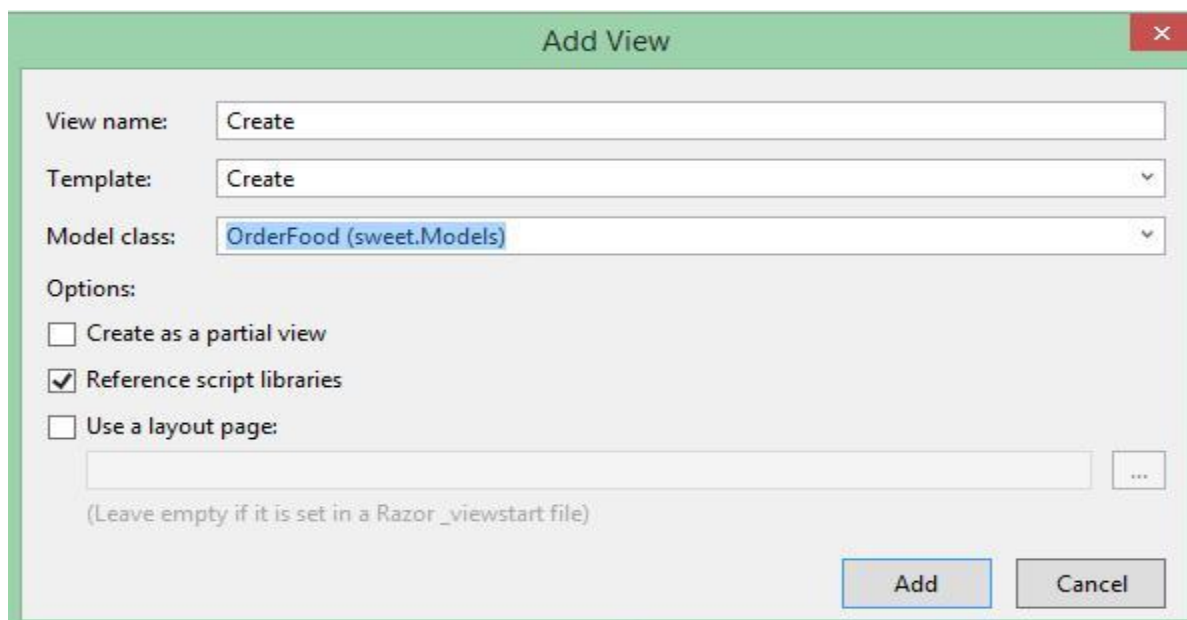
public ActionResult Create()
{
    return View(new OrderFood());
}

[HttpPost]
public ActionResult Create(OrderFood OrderFood)
{
    if (ModelState.IsValid)
    {
        TempData["Message"] = "Your Order " + OrderFood.OrderCode + " has been Saved Successfully ";
        return RedirectToAction("Create");
    }
    else
    {
        }

    }
    return View(OrderFood);
}

```

برای کنترلر Home و اکشن Create یک view اضافه کنید نوع این view از نوع create باشد و مدل آن را هم orderFood قرار دهید.



با این کار یک پوشه همنام کنترلر در داخل View ایجاد خواهد شد و همانم با اکشن ها (Action) هم فایل هایی با پسوند .cshtml قرار خواهند گرفت.

حال باید اسکریپت های لازم را در داخل این صفحه view قرار دهیم

```
<script src="~/Scripts/jquery-1.10.2.min.js"></script>
<script src="~/Scripts/jquery.validate.min.js"></script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
<script src="~/Scripts/jquery-1.11.3.min.js"></script>

<script src="~/Scripts/jquery.validate.min.js"></script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
<script src="https://lipis.github.io/bootstrap-sweetalert/lib/sweetalert.js"></script>
<link href="~/Content/sweetalert/sweetalert.css" rel="stylesheet" />
```

از آنجایی که در داخل مدل خود تعریف کرده ایم که چه فیلد هایی نباید خالی باشند اگر در آنها چیزی وارد نکنیم و سپس دکمه submit را بزنیم پیغام خطای validation که توسط جاوااسکریپت داده می شود را دریافت می کنیم. برای این که این پیغام را به شکل زیباتر و توسط Sweet Alert دریافت کنیم اسکریپت های زیر را به صفحه مورد نظر خود اضافه می کنیم.

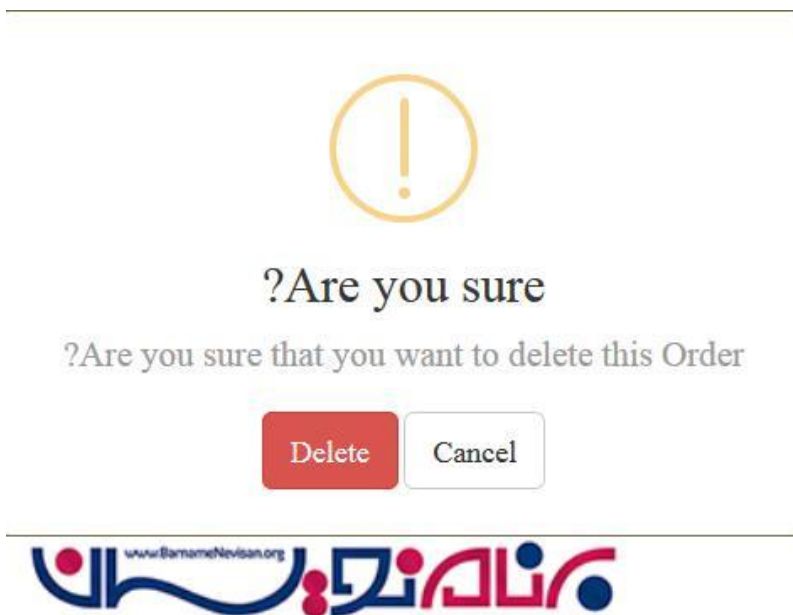
```
<script type="text/javascript">
function validateData() {

    if ($("#OrderCode").val() == "") {
        swal("لطفاً مقدار OrderCode را وارد کنید!");
        return false;
    } else if ($("#OrderPrice").val() == "") {
        swal("لطفاً مقدار OrderPrice را وارد کنید!");
        return false;
    } else {
        return true;
    }
}

@if(TempData["Message"] != null) { <text >
$(window).load(function()
{
$(document).ready(function()
{
swal({
title: "Message",
text: "@TempData["Message "]",
```

```
type: "success"  
});  
});  
}); </text>  
}  
</script>
```

بعد از اجرا گرفتن این صفحه و عدم وارد کردن فیلدهای ضروری شکل زیر را خواهید دید



کدهای کامل صفحه View به صورت زیر است

```
@model sweet.Models.OrderFood  
  
@{  
    ViewBag.Title = "Create";  
}  
<a>  
</h2> برنامه نویسان :مرجع تخصصی برنامه نویسان </h2>  
</a>  
  
@using (Html.BeginForm())  
{
```

```

@Html.AntiForgeryToken()

<div class="form-horizontal">
  <h4>OrderFood</h4>
  <hr />
  @Html.ValidationSummary(true, "", new { @class = "text-danger" })
  <div class="form-group">
    @Html.LabelFor(model => model.OrderCode, htmlAttributes: new { @class = "control-label col-
md-2" })
    <div class="col-md-10">
      @Html.EditorFor(model => model.OrderCode, new { htmlAttributes = new { @class = "form-
control" } })
      @Html.ValidationMessageFor(model => model.OrderCode, "", new { @class = "text-danger" })
    </div>
  </div>

  <div class="form-group">
    @Html.LabelFor(model => model.OrderPrice, htmlAttributes: new { @class = "control-label col-md-
2" })
    <div class="col-md-10">
      @Html.EditorFor(model => model.OrderPrice, new { htmlAttributes = new { @class = "form-
control" } })
      @Html.ValidationMessageFor(model => model.OrderPrice, "", new { @class = "text-danger" })
    </div>
  </div>

  <div class="form-group">
    <div class="col-md-offset-2 col-md-10">
      <input type="submit" value="Create" class="btn btn-default" />
    </div>
  </div>
</div>
}

<div>
  @Html.ActionLink("Back to List", "Index")
</div>

<script src="~/Scripts/jquery-1.10.2.min.js"></script>
<script src="~/Scripts/jquery.validate.min.js"></script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
<script src="~/Scripts/jquery-1.11.3.min.js"></script>

```



```

<script src="~/Scripts/jquery.validate.min.js"></script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
<script src="https://lipis.github.io/bootstrap-sweetalert/lib/sweetalert.js"></script>
<link href="~/Content/sweetalert/sweetalert.css" rel="stylesheet" />
<script type="text/javascript">
    function validateData() {

        if ($("#OrderCode").val() == "") {
            swal("
                لطفا مقدار OrderCode را وارد کنید
            ");
            return false;
        } else if ($("#OrderPrice").val() == "") {
            swal("
                لطفا مقدار OrderPrice را وارد کنید
            ");
            return false;
        } else {
            return true;
        }
    }
}
@if(TempData["Message"] != null) { <text >
    $(window).load(function()
    {
        $(document).ready(function()
        {
            swal({
                title: "Message",
                text: "@TempData["Message "]",
                type: "success"
            });
        });
    }); </text>
}
</script>

```

```

<input type="submit" onclick=" return validateData();" value="Create"/>

```

حال برای ایجاد پیغام های اعتبارسنجی یک کنترلر به نام Delete ایجاد می کنیم.

در داخل این کنترلر دو Action به نام های Details و delete وجود دارد. ساختار کدهای کنترلر به صورت زیر است

```

[HttpGet]
public ActionResult Details()
{
    List<OrderFood> listor = new List<OrderFood>();
}

```

```

OrderFood OF = new OrderFood();
OF.OrderID = 1;
OF.OrderCode = "0001";
OF.OrderPrice = "1000";

OrderFood OF1 = new OrderFood();
OF1.OrderID = 2;
OF1.OrderCode = "0002";
OF1.OrderPrice = "2000";

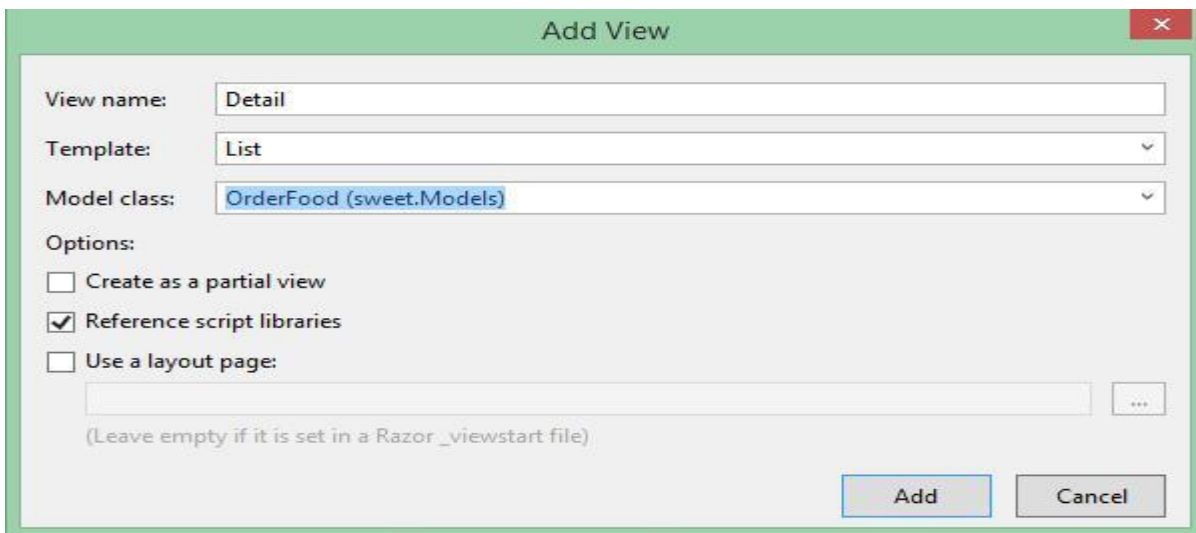
listor.Add(OF);
listor.Add(OF1);

return View(listor);
}

public ActionResult delete(string OrderID)
{
return Json("delete", JsonRequestBehavior.AllowGet);
}

```

بر روی Action به نام Details کلیک راست کرده و یک view با حالت List و با مدل OrderFood ایجاد می کنیم.




بعد از اجرا شکل زیر را خواهیم دید.



حال اسکریپت های لازم را اضافه می کنیم.

```
<meta name="viewport" content="width=device-width" />
<title>Details</title>
<script src="/Scripts/jquery-1.11.3.min.js"></script>
<script src="~/Scripts/jquery.validate.min.js"> </script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"> </script>
<link href="/Content/bootstrap.min.css" rel="stylesheet" />
<script src="/Scripts/bootstrap.js"> </script>
<script src="https://lipis.github.io/bootstrap-sweetalert/lib/sweetalert.js"> </script>

<link href="~/Content/sweetalert/sweetalert.css" rel="stylesheet" />
```

حال باید validation های مورد نظرم را به این صفحه اضافه کنیم تا sweet alert بر طبق این قوانین به ارائه پیام مناسب به کاربر پردازد

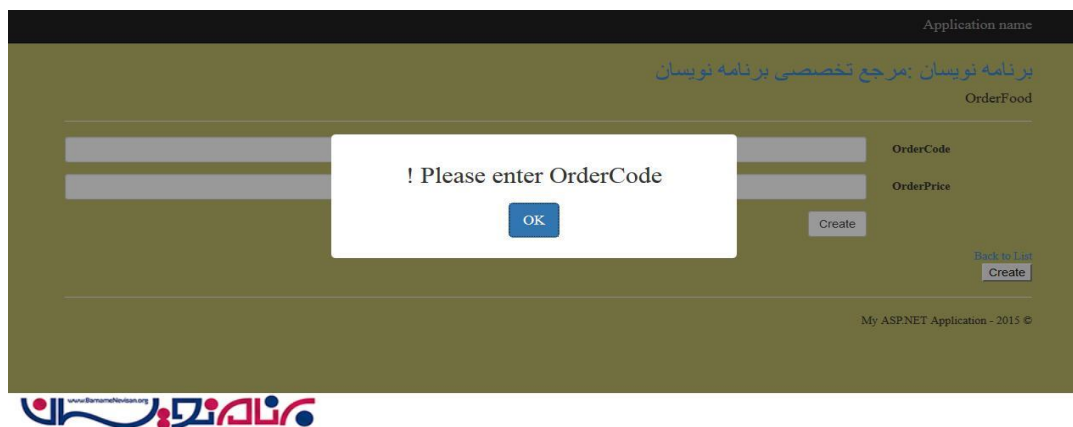
```
<script type="text/javascript">
function deleteOrder(OrderID) {
    debugger;
    swal({
        title: "Are you sure?",
        text: "Are you sure that you want to delete this Order?",
        type: "warning",
        showCancelButton: true,
        closeOnConfirm: false,
```

```

confirmButtonClass: "btn-danger",
confirmButtonText: "Delete",
confirmButtonColor: "#ec6c62"
},
function () {
$.ajax({
url: "/DeleteConfirmation/delete/",
data:
{
"OrderID": OrderID
},
type: "DELETE"
})
.done(function (data) {
sweetAlert
({
title: "Deleted!",
text: "Your file was successfully deleted!",
type: "success"
}),
function () {
window.location.href = '/DeleteConfirmation/Details';
});
})
.error(function (data) {
swal("Oops", "We couldn't connect to the server!", "error");
});
});
} </script>

```

یک نمونه از پیغام های زیبای این تکنولوژی در زیر آورده شده است



کد کامل صفحه Details به صورت زیر است

```
@model IEnumerable<sweet.Models.OrderFood>

@{
    Layout = null;
}

<!DOCTYPE html>

<html>
<head>
    <meta name="viewport" content="width=device-width" />
    <title>Details</title>
    <script src="/Scripts/jquery-1.11.3.min.js"></script>
    <script src="~/Scripts/jquery.validate.min.js"></script>
    <script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
    <link href="/Content/bootstrap.min.css" rel="stylesheet" />
    <script src="/Scripts/bootstrap.js"></script>
    <script src="https://lipis.github.io/bootstrap-sweetalert/lib/sweetalert.js"></script>

    <link href="~/Content/sweetalert/sweetalert.css" rel="stylesheet" />
    <script type="text/javascript">
        function deleteOrder(OrderID) {
            debugger;
            swal({
                title: "Are you sure?",
                text: "Are you sure that you want to delete this Order?",
                type: "warning",
                showCancelButton: true,
                closeOnConfirm: false,
                confirmButtonClass: "btn-danger",
                confirmButtonText: "Delete",
                confirmButtonColor: "#ec6c62"
            },
            function () {
                $.ajax({
                    url: "/DeleteConfirmation/delete/",
                    data:
                    {
                        "OrderID": OrderID
                    },
                    type: "DELETE"
```

```

    })
    .done(function (data) {
        sweetAlert
        ({
            title: "Deleted!",
            text: "Your file was successfully deleted!",
            type: "success"
        },
        function () {
            window.location.href = '/DeleteConfirmation/Details!';
        });
    });
    .error(function (data) {
        swal("Oops", "We couldn't connect to the server!", "error");
    });
});
} </script>
</head>

```

```

<body style="background-color: darkkhaki;">
    <p style = "text-align:left" >
    Delete Order </p> <table style="margin-left:20px;">
        <tr>
            <th style="text-align:center">
                @Html.DisplayNameFor(model => model.OrderCode) |
            </th>
            <th style="text-align:center">
                @Html.DisplayNameFor(model => model.OrderPrice) |
            </th>
            <th> Action </th>
        </tr>
        @foreach (var item in Model)
        {
            <tr>
                <td style="text-align:center">
                    @Html.DisplayFor(modelItem => item.OrderCode) </td> <td style = "text-align:center" >
                    @Html.DisplayFor(modelItem => item.OrderPrice) </td> <td style = "text-align:center" >
                    <input id="Delete"
                        onclick="deleteOrder('@item.OrderID')"
                        type="button"
                        value="Delete" />
                </td>
            </tr>
        }
    }

```



```
</table>  
</body>  
</html>
```

