

Monday, July 11, 2016

CRUD Operation In ASP.NET MVC Using AJAX And Bootstrap

By Anoop Kumar Sharma ASP.NET MVC 8 comments

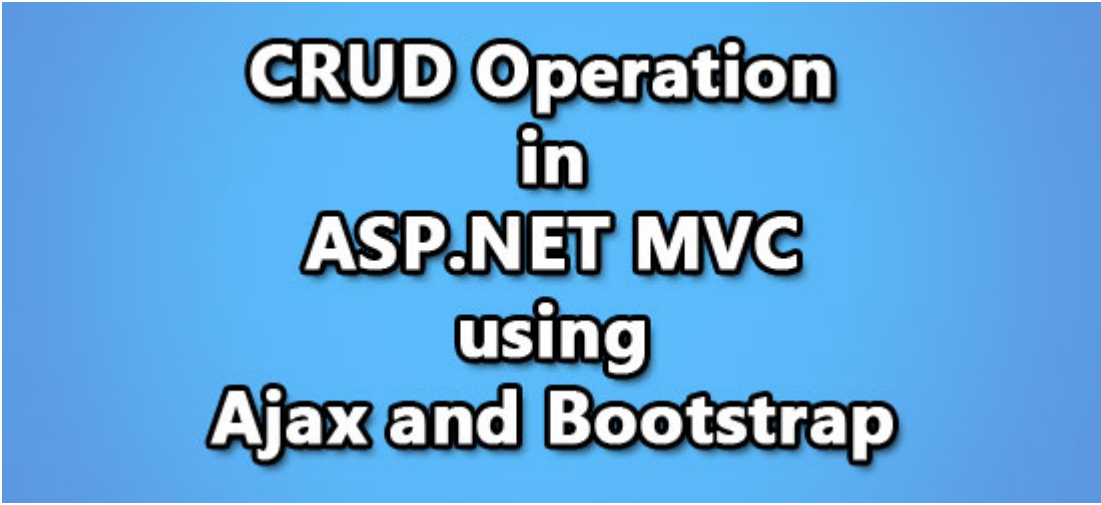
This article shows, how to perform CRUD Operation in ASP.NET MVC, using AJAX and Bootstrap. In previous ASP.NET MVC tutorials of this series, we saw,

[Creating First Application In ASP.NET MVC](#)
[Pass Parameter Or Query String In Action Method In ASP.NET MVC](#)
[Passing Data from Controller To View In ASP.NET MVC](#)
[Strongly Typed View Vs Dynamically Typed View In ASP.NET MVC](#)
[Working With Built-In HTML Helper Classes In ASP.NET MVC](#)
[Inline and Custom HTML Helpers In ASP.NET MVC](#)

What is AJAX and Bootstrap?

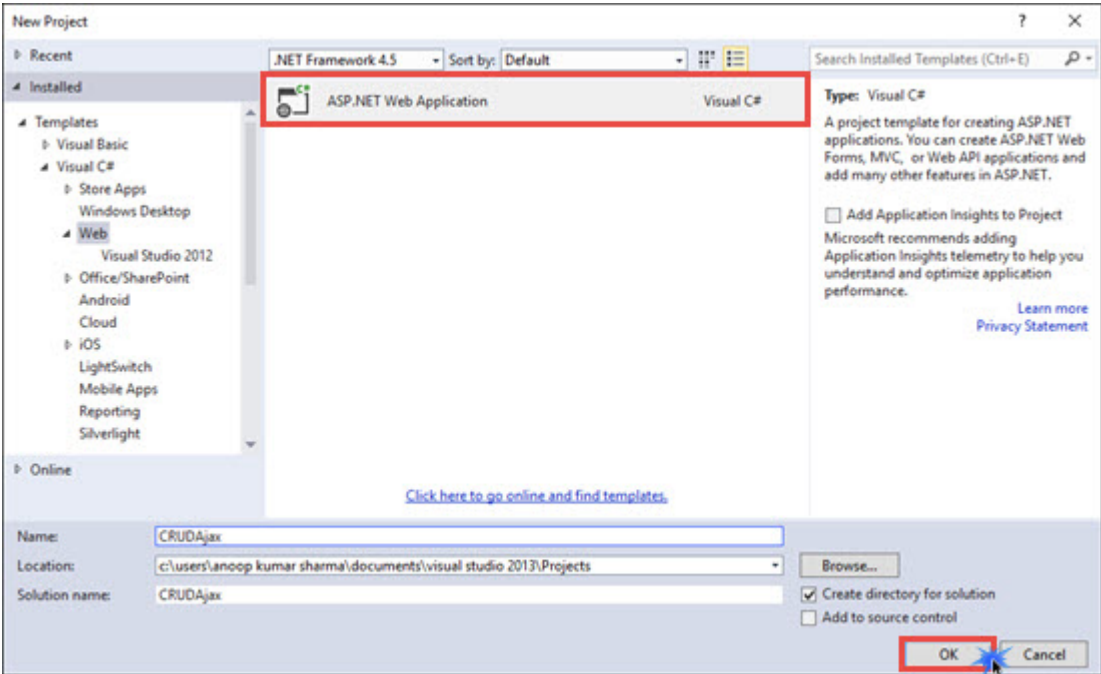
AJAX (Asynchronous JavaScript and XML) in the Web Application is used to update parts of the existing page and to retrieve the data from the Server asynchronously. AJAX improves the performance of the Web Application and makes the Application more interactive.

Bootstrap is the one of the most popular HTML, CSS and JS framework for developing responsive, mobile first projects on the Web.

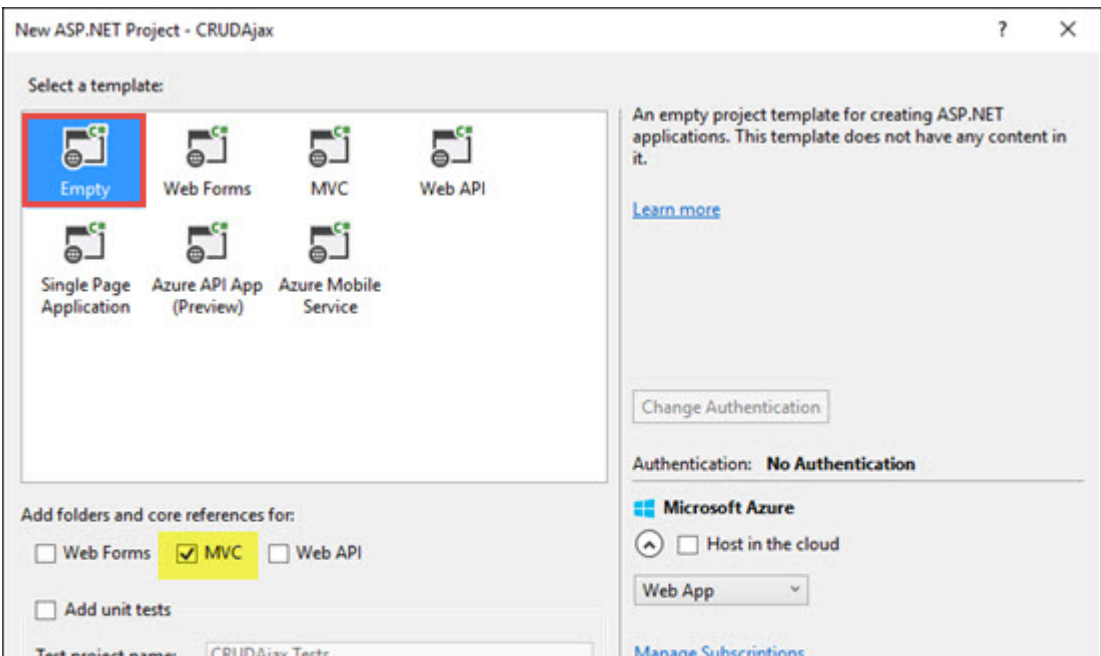


Let's Begin

Create a new ASP.NET Web Application.

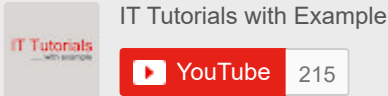


Select Empty ASP.NET MVC template and click OK.



Search this site...

Subscribe us on YouTube





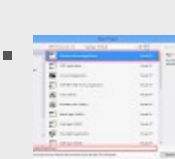


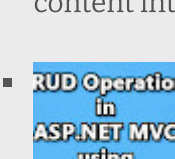
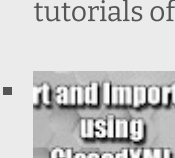
Subscribe Now

SUBSCRIBE TO OUR EMAIL NEWSLETTER & RECEIVE UPDATES RIGHT IN YOUR INBOX.

☐ Enter your email address :)

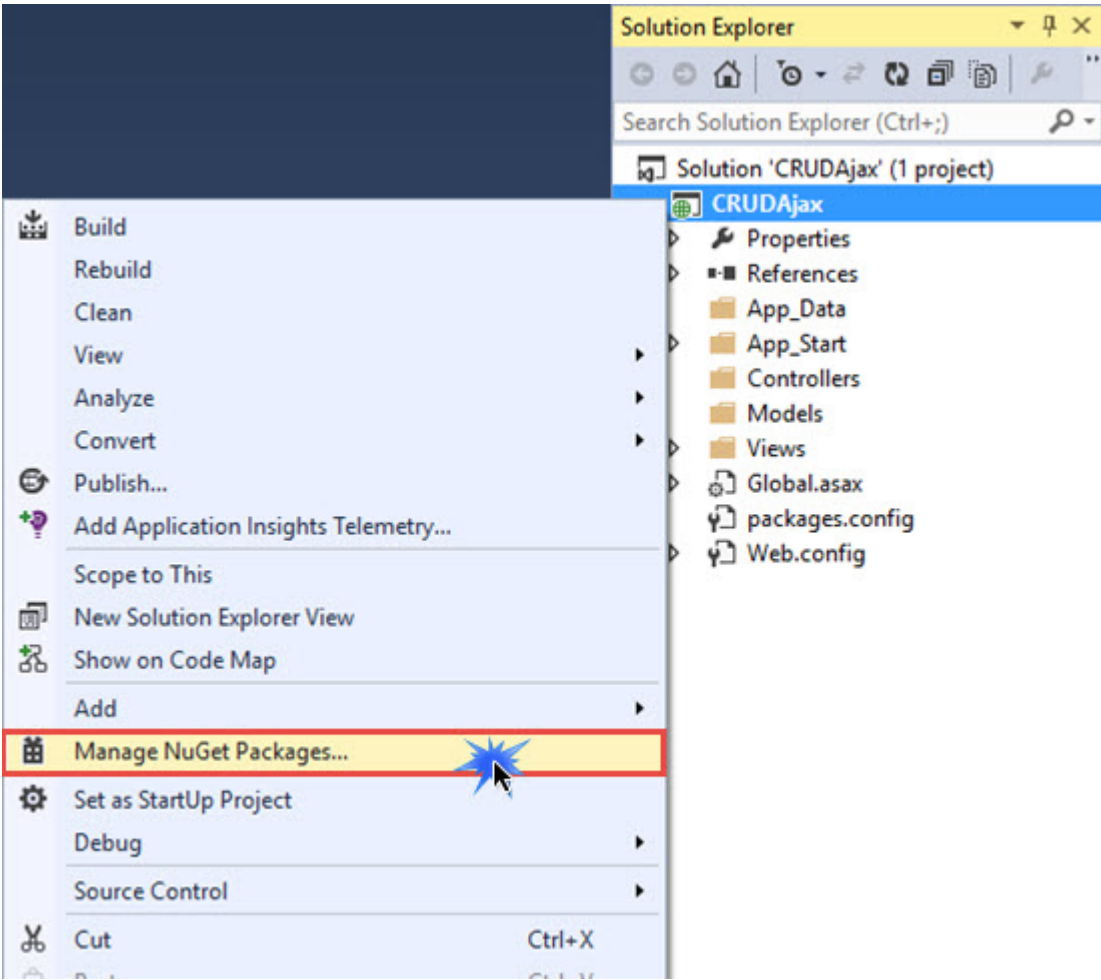
SUBSCRIBE NOW !

Popular Posts

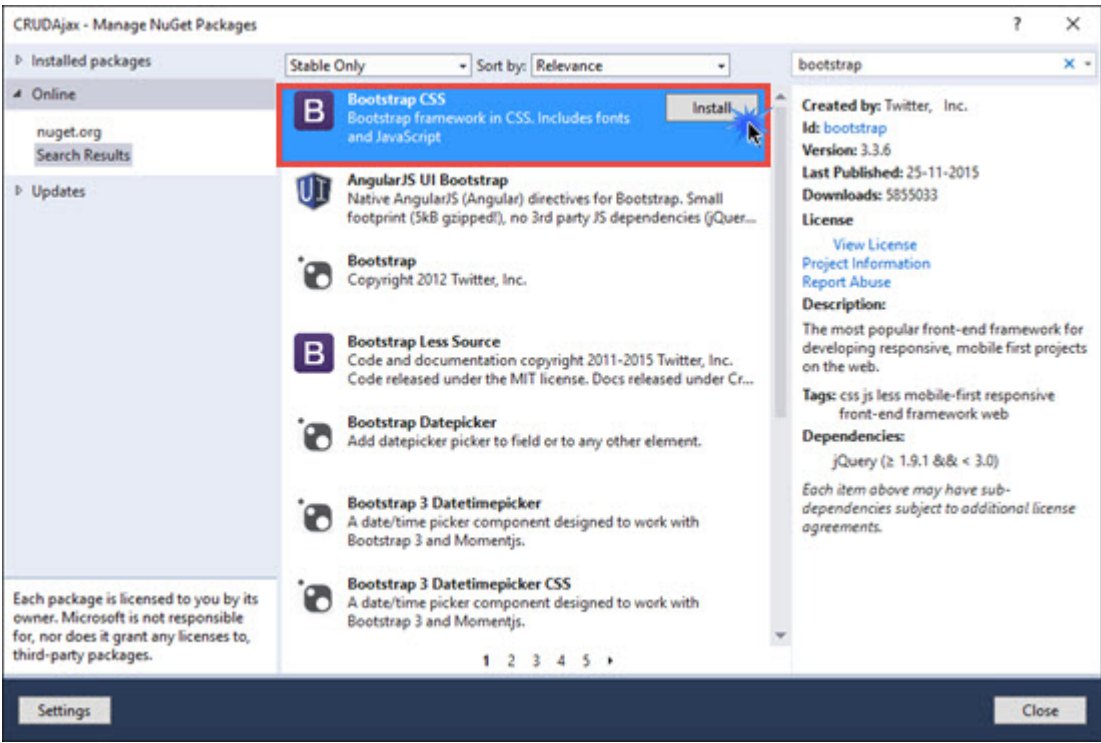
-  Simple Windows Form Login Application in C#
In this Post, we will learn how to create a Simple Windows form Login application. Let's Begin: 1. Create a New Windows Form Applicatio...
-  Create Simple Comment Box in ASP.NET
All of us have seen a comment box on various websites. In this Post, we will learn how to create a Simple Comment Box in ASP.NET. F...
-  Insert, Update and Delete Record in DataGridView C#
In this Article, we will learn How to Insert, Update and Delete Record in DataGridView in C# Windows Form Application. In Previous Post, we...
-  How to Install SQLyog Community Edition
In this article, we will learn How to install SQLyog Community Edition in Windows Operating System. Currently, I am using it to access the M...
-  Server Side Pagination in ASP.NET MVC
In this Article, we will learn How to implement server-side paging in ASP.NET MVC. Pagination is used to divide the data or content into mu...
-  CRUD Operation in ASP.NET MVC Using AJAX And Bootstrap
This article shows, how to perform CRUD Operation in ASP.NET MVC, using AJAX and Bootstrap. In previous ASP.NET MVC tutorials of this serie...
-  Export and Import Excel file using ClosedXML in ASP.NET MVC



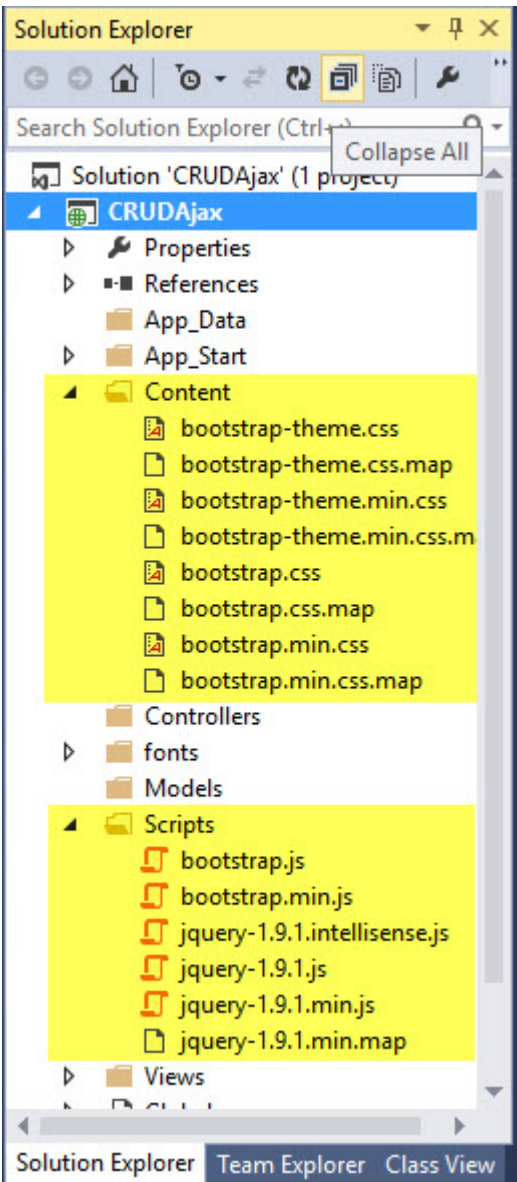
Now, right-click on the project and click Manage NuGet Packages.



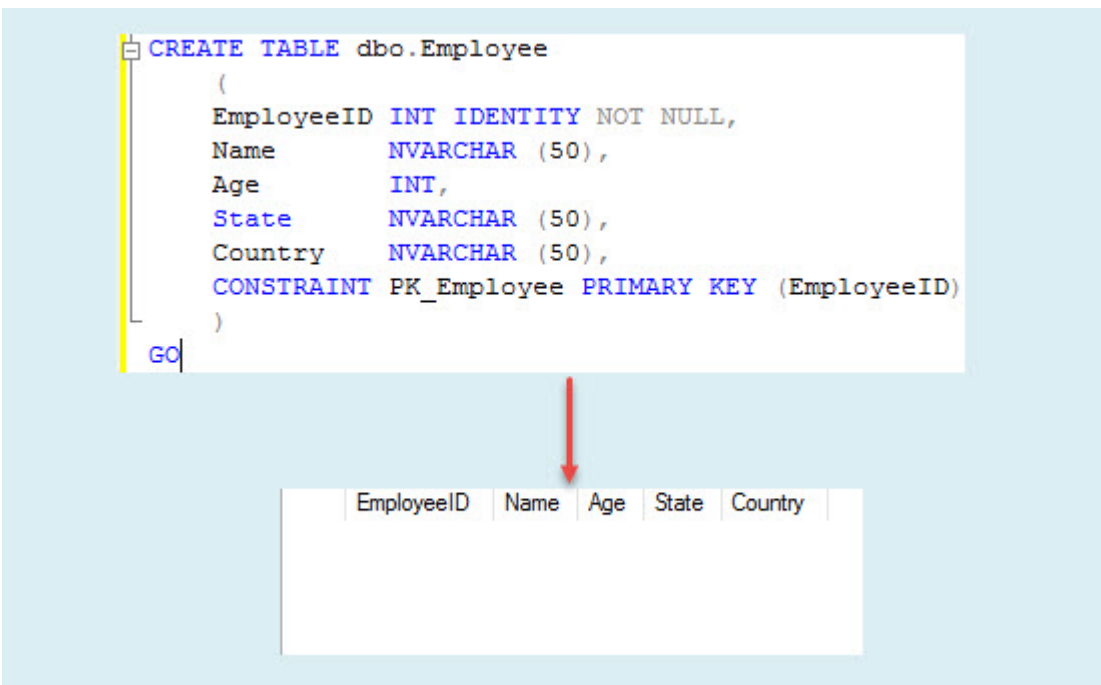
Search for Bootstrap and then click Install button.



After installing the package, you will see the Content and Scripts folder being added in your Solution Explorer.



Now, create a database and add a table (named Employee). The following is the schema for creating a table Employee:



In software applications, sometime user needs to export or Import the data in Excel format in order to perform several operations. In this ...

Duplicate Character from String
Program to remove the duplicate character from the string
In this program, we will learn How to remove the duplicate character from the string in C#. I will recommend you to check the below link...

Enable RDLC Report in Visual Studio 2017
In this Article, you will learn How to enable or install RDLC Report Designer in Visual Studio 2017.

Learn Angular in Hindi (YouTube Series)
Recently, we have begun the “Learn Angular in Hindi” Series on our official YouTube channel. Many people commented/wrote us to start a ser...

Contact us

Name

Email *

Message *

Send

Categories

.Net Core Android Angular AngularJS ASP.NET ASP.NET Core ASP.NET MVC Azure Bootstrap c C# C# Console Program Cloud Computing css Google Cloud Platform HTML Interview Javascript MongoDB Python React Native Reports SQL TypeScript Visual Studio Windows Form Application Windows Phone Application WPF Xamarin

Like us on Facebook



Blog Archive

- 2022 (4)
- 2021 (11)
- 2020 (19)
- 2019 (5)
- 2018 (11)
- 2017 (7)
- ▼ 2016 (10)
 - October (1)
 - September (1)
 - August (2)
 - ▼ July (2)
 - HandleError Action Filter In ASP.NET MVC
 - CRUD Operation In ASP.NET MVC Using AJAX And Boots...
 - June (1)
 - May (1)
 - April (1)
 - February (1)
- 2015 (24)
- 2014 (38)
- 2013 (6)

After the table creation, create the stored procedures for Select, Insert, Update and Delete operations.

```
--Select Employees
Create Procedure SelectEmployee
as
Begin
Select * from Employee;
End

--Insert and Update Employee
Create Procedure InsertUpdateEmployee
(
@Id integer,
@Name nvarchar(50),
@Age integer,
@State nvarchar(50),
@Country nvarchar(50),
@Action varchar(10)
)
As
Begin
if @Action='Insert'
Begin
Insert into Employee (Name, Age, [State], Country)
values (@Name, @Age, @State, @Country);
End
if @Action='Update'
Begin
Update Employee set Name=@Name, Age=@Age, [State]=@State, Country=@Country where
EmployeeID=@Id;
End
End

--Delete Employee
Create Procedure DeleteEmployee
(
@Id integer
)
as
Begin
Delete Employee where EmployeeID=@Id;
End
```

Right click on Modal Folder and add Employee.cs class.

Employee.cs Code:

```
public class Employee
{
    public int EmployeeID { get; set; }

    public string Name { get; set; }

    public int Age { get; set; }

    public string State { get; set; }

    public string Country { get; set; }
}
```

Now, add another class in Modal Folder named as EmployeeDB.cs for the database related operations. In this example, I am going to use ADO.NET to access the data from the database.

```
public class EmployeeDB
{
    //declare connection string
    string cs = ConfigurationManager.ConnectionStrings["DBCS"].ConnectionString;

    //Return list of all Employees
    public List<Employee> ListAll()
    {
        List<Employee> lst = new List<Employee>();
        using(SqlConnection con=new SqlConnection(cs))
        {
            con.Open();
            SqlCommand com = new SqlCommand("SelectEmployee", con);
            com.CommandType = CommandType.StoredProcedure;
            SqlDataReader rdr = com.ExecuteReader();
            while(rdr.Read())
            {
                lst.Add(new Employee {
                    EmployeeID=Convert.ToInt32(rdr["EmployeeID"]),
                    Name=rdr["Name"].ToString(),
                    Age = Convert.ToInt32(rdr["Age"]),
                    State = rdr["State"].ToString(),
                    Country = rdr["Country"].ToString(),
                });
            }
            return lst;
        }
    }

    //Method for Adding an Employee
    public int Add(Employee emp)
```

```

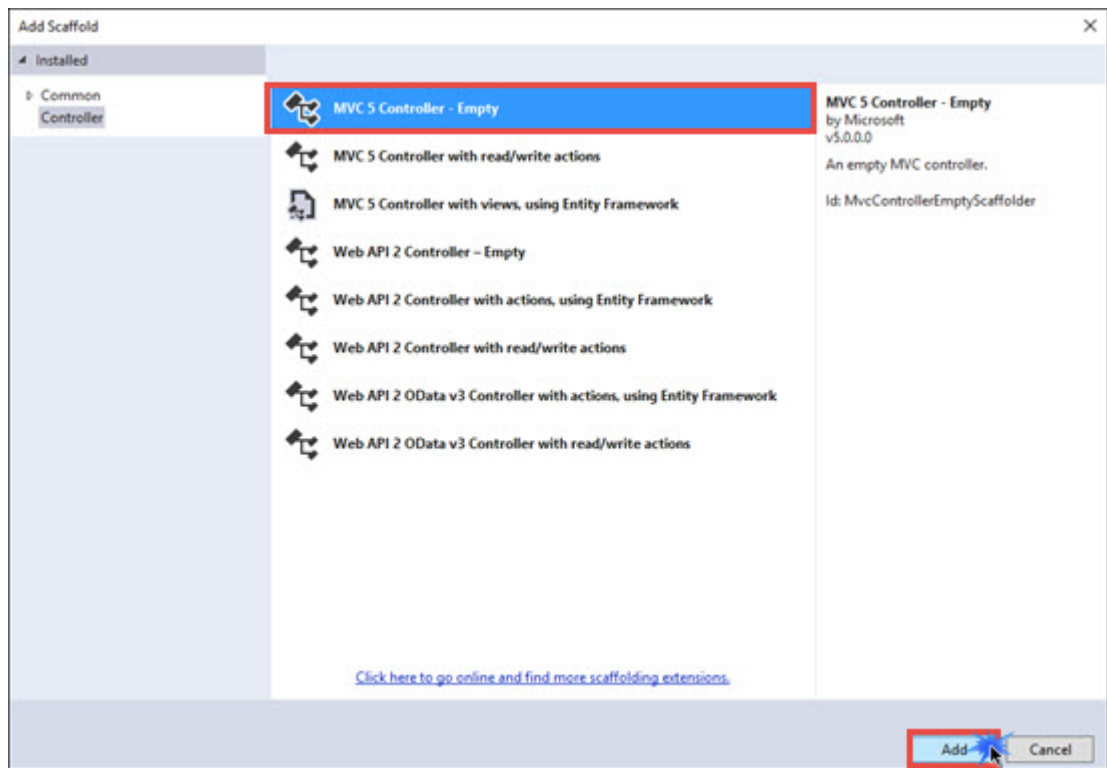
{
    int i;
    using(SqlConnection con=new SqlConnection(cs))
    {
        con.Open();
        SqlCommand com = new SqlCommand("InsertUpdateEmployee", con);
        com.CommandType = CommandType.StoredProcedure;
        com.Parameters.AddWithValue("@Id", emp.EmployeeID);
        com.Parameters.AddWithValue("@Name", emp.Name);
        com.Parameters.AddWithValue("@Age", emp.Age);
        com.Parameters.AddWithValue("@State", emp.State);
        com.Parameters.AddWithValue("@Country", emp.Country);
        com.Parameters.AddWithValue("@Action", "Insert");
        i = com.ExecuteNonQuery();
    }
    return i;
}

//Method for Updating Employee record
public int Update(Employee emp)
{
    int i;
    using (SqlConnection con = new SqlConnection(cs))
    {
        con.Open();
        SqlCommand com = new SqlCommand("InsertUpdateEmployee", con);
        com.CommandType = CommandType.StoredProcedure;
        com.Parameters.AddWithValue("@Id", emp.EmployeeID);
        com.Parameters.AddWithValue("@Name", emp.Name);
        com.Parameters.AddWithValue("@Age", emp.Age);
        com.Parameters.AddWithValue("@State", emp.State);
        com.Parameters.AddWithValue("@Country", emp.Country);
        com.Parameters.AddWithValue("@Action", "Update");
        i = com.ExecuteNonQuery();
    }
    return i;
}

//Method for Deleting an Employee
public int Delete(int ID)
{
    int i;
    using (SqlConnection con = new SqlConnection(cs))
    {
        con.Open();
        SqlCommand com = new SqlCommand("DeleteEmployee", con);
        com.CommandType = CommandType.StoredProcedure;
        com.Parameters.AddWithValue("@Id", ID);
        i = com.ExecuteNonQuery();
    }
    return i;
}
}

```

Right click on Controllers folder, add an Empty Controller and name it as HomeController.



Now, open HomeController and add the following action methods:

```

public class HomeController : Controller
{
    EmployeeDB empDB = new EmployeeDB();
    // GET: Home
    public ActionResult Index()
    {
        return View();
    }
    public JsonResult List()
    {
        return Json(empDB.ListAll(), JsonRequestBehavior.AllowGet);
    }
    public JsonResult Add(Employee emp)
    {
        return Json(empDB.Add(emp), JsonRequestBehavior.AllowGet);
    }
    public JsonResult GetbyID(int ID)
    {
        var Employee = empDB.ListAll().Find(x => x.EmployeeID.Equals(ID));
    }
}

```



```
        return Json(Employee, JsonRequestBehavior.AllowGet);
    }
    public JsonResult Update(Employee emp)
    {
        return Json(empDB.Update(emp), JsonRequestBehavior.AllowGet);
    }
    public JsonResult Delete(int ID)
    {
        return Json(empDB.Delete(ID), JsonRequestBehavior.AllowGet);
    }
}
```

Right click on the Index action method of HomeController and click on Add View. As we are going to use Bootstrap and AJAX, we have to add their relative Scripts and CSS references in the head section of the view. I have also added employee.js, which will contain all AJAX code, that are required for CRUD operation.

```
<script src="../../Scripts/jquery-1.9.1.js"></script>
<script src="../../Scripts/bootstrap.js"></script>
<link href="../../Content/bootstrap.css" rel="stylesheet" />
<script src="../../Scripts/employee.js"></script>
```

Add the code, given below, in Index.cshtml view:

```
<div class="container">
    <h2>Employees Record</h2>
    <button type="button" class="btn btn-primary" data-toggle="modal" data-
target="#myModal" onclick="clearTextBox();">Add New Employee</button><br /><br />
    <table class="table table-bordered table-hover">
        <thead>
            <tr>
                <th>
                    ID
                </th>
                <th>
                    Name
                </th>
                <th>
                    Age
                </th>
                <th>
                    State
                </th>
                <th>
                    Country
                </th>
                <th>
                    Action
                </th>
            </tr>
        </thead>
        <tbody class="tbody">

        </tbody>
    </table>
</div>
<div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-
labelledby="myModallabel" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                @<button type="button" class="close" data-dismiss="modal"><span aria-
hidden="true">&times;</span></button>*@
                <button type="button" class="close" data-dismiss="modal">&times;</button>
                <h4 class="modal-title" id="myModallabel">Add Employee</h4>
            </div>
            <div class="modal-body">
                <form>
                    <div class="form-group">
                        <label for="EmployeeId">ID</label>
                        <input type="text" class="form-control" id="EmployeeID"
placeholder="Id" disabled="disabled"/>
                    </div>
                    <div class="form-group">
                        <label for="Name">Name</label>
                        <input type="text" class="form-control" id="Name"
placeholder="Name"/>
                    </div>
                    <div class="form-group">
                        <label for="Age">Age</label>
                        <input type="text" class="form-control" id="Age" placeholder="Age"
/>
                    </div>
                    <div class="form-group">
                        <label for="State">State</label>
                        <input type="text" class="form-control" id="State"
placeholder="State"/>
                    </div>
                    <div class="form-group">
                        <label for="Country">Country</label>
                        <input type="text" class="form-control" id="Country"
placeholder="Country"/>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
```

```

        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-primary" id="btnAdd" onclick="return
Add();">Add</button>
            <button type="button" class="btn btn-primary" id="btnUpdate"
style="display:none;" onclick="Update();">Update</button>
            <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
        </div>
    </div>
</div>
</div>
</div>

```

In the code, given above, we have added a button for adding New Employee. On clicking, It will open the modal dialog box of the bootstrap, which contains several fields of the employees for saving. We have also added a table, which will be populated with the use of AJAX.

Employee.js Code

```

//Load Data in Table when documents is ready
$(document).ready(function () {
    loadData();
});

//Load Data function
function loadData() {
    $.ajax({
        url: "/Home/List",
        type: "GET",
        contentType: "application/json;charset=utf-8",
        dataType: "json",
        success: function (result) {
            var html = '';
            $.each(result, function (key, item) {
                html += '<tr>';
                html += '<td>' + item.EmployeeID + '</td>';
                html += '<td>' + item.Name + '</td>';
                html += '<td>' + item.Age + '</td>';
                html += '<td>' + item.State + '</td>';
                html += '<td>' + item.Country + '</td>';
                html += '<td><a href="#" onclick="return getbyID(' + item.EmployeeID +
')">Edit</a> | <a href="#" onclick="Delele(' + item.EmployeeID + ')">Delete</a></td>';
                html += '</tr>';
            });
            $('.tbody').html(html);
        },
        error: function (errorMessage) {
            alert(errorMessage.responseText);
        }
    });
}

//Add Data Function
function Add() {
    var res = validate();
    if (res == false) {
        return false;
    }
    var empObj = {
        EmployeeID: $('#EmployeeID').val(),
        Name: $('#Name').val(),
        Age: $('#Age').val(),
        State: $('#State').val(),
        Country: $('#Country').val()
    };
    $.ajax({
        url: "/Home/Add",
        data: JSON.stringify(empObj),
        type: "POST",
        contentType: "application/json;charset=utf-8",
        dataType: "json",
        success: function (result) {
            loadData();
            $('#myModal').modal('hide');
        },
        error: function (errorMessage) {
            alert(errorMessage.responseText);
        }
    });
}

//Function for getting the Data Based upon Employee ID
function getbyID(EmpID) {
    $('#Name').css('border-color', 'lightgrey');
    $('#Age').css('border-color', 'lightgrey');
    $('#State').css('border-color', 'lightgrey');
    $('#Country').css('border-color', 'lightgrey');
    $.ajax({
        url: "/Home/getbyID/" + EmpID,
        typr: "GET",
        contentType: "application/json;charset=UTF-8",
        dataType: "json",
        success: function (result) {
            $('#EmployeeID').val(result.EmployeeID);
        }
    });
}

```

```

        $('#Name').val(result.Name);
        $('#Age').val(result.Age);
        $('#State').val(result.State);
        $('#Country').val(result.Country);

        $('#myModal').modal('show');
        $('#btnUpdate').show();
        $('#btnAdd').hide();
    },
    error: function (errorMessage) {
        alert(errorMessage.responseText);
    }
});
return false;
}

//function for updating employee's record
function Update() {
    var res = validate();
    if (res == false) {
        return false;
    }
    var empObj = {
        EmployeeID: $('#EmployeeID').val(),
        Name: $('#Name').val(),
        Age: $('#Age').val(),
        State: $('#State').val(),
        Country: $('#Country').val(),
    };
    $.ajax({
        url: "/Home/Update",
        data: JSON.stringify(empObj),
        type: "POST",
        contentType: "application/json;charset=utf-8",
        dataType: "json",
        success: function (result) {
            loadData();
            $('#myModal').modal('hide');
            $('#EmployeeID').val("");
            $('#Name').val("");
            $('#Age').val("");
            $('#State').val("");
            $('#Country').val("");
        },
        error: function (errorMessage) {
            alert(errorMessage.responseText);
        }
    });
}

//function for deleting employee's record
function Delele(ID) {
    var ans = confirm("Are you sure you want to delete this Record?");
    if (ans) {
        $.ajax({
            url: "/Home/Delete/" + ID,
            type: "POST",
            contentType: "application/json;charset=UTF-8",
            dataType: "json",
            success: function (result) {
                loadData();
            },
            error: function (errorMessage) {
                alert(errorMessage.responseText);
            }
        });
    }
}

//Function for clearing the textboxes
function clearTextBox() {
    $('#EmployeeID').val("");
    $('#Name').val("");
    $('#Age').val("");
    $('#State').val("");
    $('#Country').val("");
    $('#btnUpdate').hide();
    $('#btnAdd').show();
    $('#Name').css('border-color', 'lightgrey');
    $('#Age').css('border-color', 'lightgrey');
    $('#State').css('border-color', 'lightgrey');
    $('#Country').css('border-color', 'lightgrey');
}

//Valdidation using jquery
function validate() {
    var isValid = true;
    if ($('#Name').val().trim() == "") {
        $('#Name').css('border-color', 'Red');
        isValid = false;
    }
    else {
        $('#Name').css('border-color', 'lightgrey');
    }
}
```

```

    if ($('#Age').val().trim() == "") {
        $('#Age').css('border-color', 'Red');
        isValid = false;
    }
    else {
        $('#Age').css('border-color', 'lightgrey');
    }
    if ($('#State').val().trim() == "") {
        $('#State').css('border-color', 'Red');
        isValid = false;
    }
    else {
        $('#State').css('border-color', 'lightgrey');
    }
    if ($('#Country').val().trim() == "") {
        $('#Country').css('border-color', 'Red');
        isValid = false;
    }
    else {
        $('#Country').css('border-color', 'lightgrey');
    }
    return isValid;
}

```

Build and run the Application.

Adding a Record (Preview)

Employees Record

Add New Employee

ID	Name	Age	State	Country	Action
----	------	-----	-------	---------	--------

Editing a Record (Preview)

Employees Record

Add New Employee

ID	Name	Age	State	Country	Action
1	Anoop Sharma	22	Delhi	India	Edit Delete

Delete a Record(Preview)

Employees Record

Add New Employee

ID	Name	Age	State	Country	Action
1	Anoop Kumar Sharma	24	Delhi	India	Edit Delete
2	Akshay Kumar	40	Delhi	India	Edit Delete


I hope you like it. Thanks!



Download Source Code


[Download Source Code via Google Drive]

or

 **Clone it from GitHub**


[Clone or Download it from GitHub]

PROTECTED BY **COPYSCAPE** DO NOT COPY

Share This:  Facebook  Twitter  Google+  Stumble  Digg




Related Posts:




Passing Data from Controller to View in ASP.NET MVC
In ASP.Net MVC, We can pass data from the controller to corresponding view using ViewData and ViewBag. We can also pass data from the View to Control... [Read More](#)




HandleError Action Filter In ASP.NET MVC
In this article, you will learn how to handle errors using HandleError Action Filter in ASP.Net MVC. In previous ASP.NET MVC tutorials of thi... [Read More](#)



Working With Built-In HTML Helper Classes In ASP.NET MVC
In this Article, we will learn how to use Built-In HTML Helper Classes in ASP.NET MVC. In previous ASP.NET MVC Tutorials, we saw, Creating First Ap... [Read More](#)



Deploy ASP.Net MVC Application to Windows Azure
In this article, we will learn how to publish or Deploy ASP.Net MVC application to Windows Azure. In previous ASP.NET MVC tutorials of this series, w... [Read More](#)




Inline and Custom Html Helpers in ASP.Net MVC
With the help of HTML Helper class, we can create HTML Controls programmatically. HTML Helpers are used in View to render HTML content. HTML Helpers ... [Read More](#)

Newer Post

Home

Older Post

8 comments:

 **Manav Pandya** 19 September 2016 at 03:18

Nice one brother ,,
can you share all examples folder of demo codes to me if possible
please sir ...


Reply

▼ Replies

 **Anoop Kumar Sharma** 19 September 2016 at 09:13

Thanks Manav. You can download the source code from the link given in the article.

Reply

 **Unknown** 25 October 2016 at 18:09

Nice tutorial.,
i want to add table content sorting and filter with datatable, can you help me sir?


Reply

▼ Replies

 **Anoop Kumar Sharma** 26 October 2016 at 11:07


Thanks asprys dev8, I will post an article on this topic within a week. Stay connected..!!

Reply

 **Unknown** 24 May 2017 at 03:49

i try but i have a erorr in con , connection to database

Reply

 **Vinod** 16 July 2017 at 09:49


Than you for your help .I have one problem when i try to enter a value to Sql server. I got one error on "EmployeeDB" The value "Int i" is -1 instead of 1. Hope you will reply soon .Thank You

Reply

 **Unknown** 16 October 2017 at 22:06

Hi , how to do pagnation on this application

Reply

 **ut** 18 February 2021 at 00:21


the delete(Id) ..ajax
code is not calling controller which is jsonresult delete(int id)

Reply

To leave a comment, click the button below to sign in with Google.

SIGN IN WITH GOOGLE

Featured Tutorials

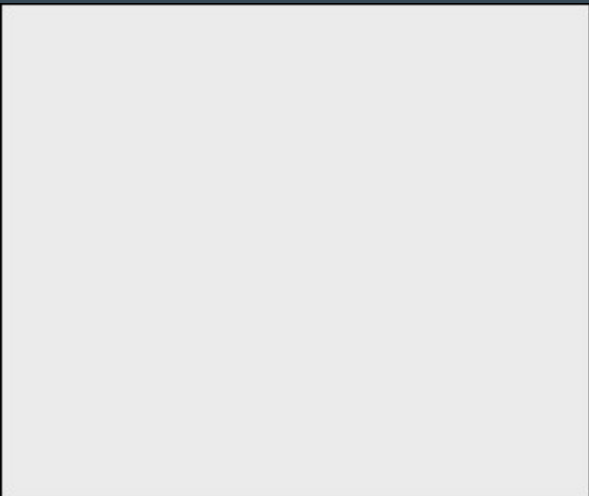
 Windows Form Application

 Windows Phone Application

 ASP.NET Web Application

ASP.NET Developer Jobs

Featured



Pages

- [Home](#)