Traditional web applications rely on synchronous method calls to process data. The entire web page is posted to the server each time data is submitted through the page, causing performance delays.

This problem can be overcome by using AJAX. AJAX allows web applications to call methods asynchronously. Instead of posting the entire page to the web server, only the required data is posted. This improves the overall performance of the web applications.

As we all know, AJAX means Asynchronous JavaScript and XML. It is a client-side script that communicates to and from a server/database without the need for a postback or a complete page refresh. The Ajax speeds up response time.

**Implementation of Ajax can be done in two way in ASP.Net Application**

* using Update Panel and,
* using jQuery

JSON[ Java Script Object Notation]

**What is the use Json ?**

**JSON** is very commonly used concept in MVC and other technology. It is a data interchange medium and a very light weighted code between client and server. It is language independent so any technology and language can use the concept of JSON. It is mainly use when AJAX call and functionality implement in project so it provide the partial rendering without refreshing the entire heavy page of web page it can transfer the data to server as well it can receive the data from server.

**JsonResult is actually a special ActionResult , which suggest the View Engine that the object of JSON type will be returned rather than normal HTML**

1. JSON objects are surrounded by curly braces {}.
2. JSON objects are written in key/value pairs.
3. Keys and values are separated by a colon.

Uses of JSON

* It is used while writing JavaScript based applications that includes browser extensions and websites.
* JSON format is used for serializing and transmitting structured data over network connection.
* It is primarily used to transmit data between a server and web applications.
* Web services and APIs use JSON format to provide public data.
* It can be used with modern programming languages.

Characteristics of JSON

* JSON is easy to read and write.
* It is a lightweight text-based interchange format.
* JSON is language independent.

$(document).ready(function () {

jsonobj = { "name": "Raman", "Age": "24", "Marks": "90" };

var res = "";

res += "Name " + jsonobj.name + "<br>";

res += "Age " + jsonobj.Age + "<br>";

res += "Marks " + jsonobj.Marks;

$("#div1").html(res); });

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**JSON OBJECT AS ARRAY**

jsonobj = [{ "Name": "Mahira", "Age": "24", "City": "Mumbai" }, { "Name": "Raj", "Age": "34", "City": "Lucknow" }];

for(i=0;i<=1;i++)

{

res += jsonobj[i].Name + "<br>";

res += jsonobj[i].Age + "</br>";

res += jsonobj[i].City + "</br>";

res += "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_</br>";

}

$("#div1").html(res);

jsonobj = [{ "name": "Raman", "Age": "24", "Marks": "90" }, { "name": "Ram", "Age": "30", "Marks": "95" }];

var res = "";

for (i = 0; i < jsonobj.length; i++)

{

res+=(jsonobj[i].name + " " + jsonobj[i].Age + " " + jsonobj[i].Marks+"<br>");

}

$("#div1").html(res);

**View Coading**

$.getJSON("/Admin/GetAttendence", {Rn:rollno,Atten:attendance}, function (msg) {

if (msg == "success")

alert("saved records");

})

**Controller Method**

public ActionResult GetAttendence(string Rn, string Atten)

{

string msg = "success";

return Json(msg, JsonRequestBehavior.AllowGet);

//return View();

}

**Insertion and searching Using Json Ajax implementation with page refresh**

**While creating the function related to Json make sure post function should not be created**

$(document).ready(function () {

$("#btn").click(function () {

var name, marks;

nam = $("#name").val();

marks = $("#marks").val();

$.getJSON("/Home/insertion", { n: nam, m: marks }, function (msg) {

if (msg == "success") {

alert('Record Inserted');

}

else

alert('Record not inserted');

});

**Controller part coading**

public ActionResult insertion(string n,string m)

{

int marks = int.Parse(m);

string msg;

string q = "insert into tbl\_student values('" + n + "','" + marks + "')";

bool j = dm.insertupdatedelete(q);

if (j == true)

msg = "success";

else

msg="failure";

return (Json(msg, JsonRequestBehavior.AllowGet));

}

**Search Record**

$("#searchbtn").click(function () {

var ssid = $("#sid").val();

$.getJSON("/Home/search", { searchid: ssid }, function (msg) {

if (msg == "find") {

$("#span1").html("Record Found Successfully");

}

else {

$("#span1").html("Record Not Found Successfully");

}

});

});

**Controller part coading for search**

public ActionResult search(string searchid)

{

int id = int.Parse(searchid);

string msg;

string q="select \* from tbl\_student where studentid='"+id+"'";

DataTable dt = dm.readbulkdata(q);

if (dt.Rows.Count > 0)

msg = "find";

else

msg = "not found";

return (Json(msg, JsonRequestBehavior.AllowGet));

}