

SharePoint Rest API

By Bijay Kumar Sahoo (@Fewlines4Biju)

In this SharePoint rest api tutorial and examples article, we will discuss how to work with SharePoint rest API, advantages of rest api in SharePoint 2013/2016/Online. Also, we will discuss how SharePoint REST API works, various SharePoint Rest API HTTP commands, various properties of SharePoint Rest API, and also we will see various SharePoint Online rest api examples.

The Rest API SharePoint Online code whatever we will use in this tutorial will work in SharePoint Online as well as SharePoint 2013/2016/2019.

SharePoint Rest API

We worked on various client object model techniques like CSOM, JSOM, etc. Microsoft also introduced REST (Representational State Transfer) service in SharePoint 2013 which is comparable to existing SharePoint client object models.

By using Rest API, we can interact with SharePoint remotely. We can interact to with SharePoint data by using any technology that supports REST web requests like you can perform create, read, update and delete operations from apps for SharePoint, client applications, etc. using REST technologies.

How SharePoint REST API works

In SharePoint 2013/2016/Online you can interact remotely with SharePoint sites using REST API. You will be able to interact by using any technology that supports REST web requests. You need to construct a RESTful HTTP request, using the Open Data Protocol (OData) standard.

To access SharePoint resources using REST, construct a RESTful HTTP request, using the Open Data Protocol (OData) standard, which corresponds to the desired client object model API.

Example:

Here is below the client object model and REST:

Client object model method:

```
List.GetByTitle(listname)
```

REST endpoint:

```
http://server/site/_api/lists/getbytitle('listname')
```

Similarly, if you want to retrieve all the items from a SharePoint list, you will use the below CSOM code.

```
var items = List.GetByTitle("Employees").GetItems();
```

If you want to retrieve the same list items using Rest API in SharePoint 2013, you have to construct the Rest end point URL:

```
https://{site_url}/_api/lists/getbytitle('Employees')/items
```

Example:

```
https://tsinfo.sharepoint.com/sites/enjoysharepoint/_api/lists/getbytitle('Employees')/items
```

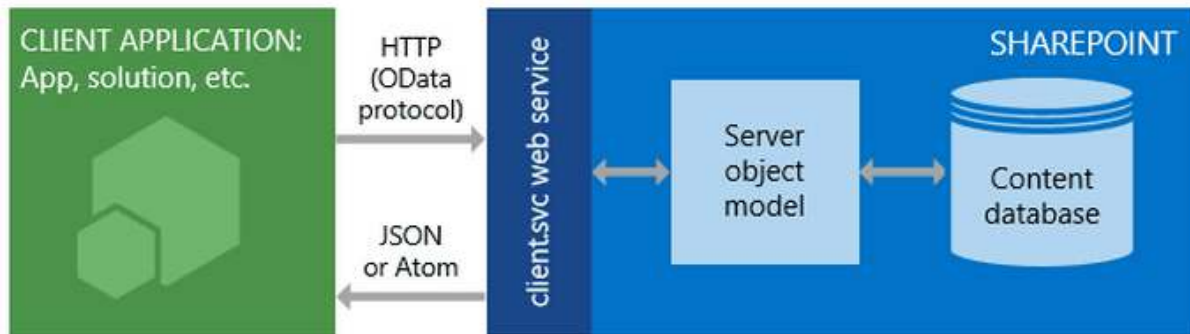
TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

Here your application will send an HTTP request and client.svc web service handles the request which internally calls the Server object model to retrieve data from the Content database.

And then it sends the response in either Atom or JavaScript Object Notation (JSON) format which your client application should be able to parse.



Here your application will send an HTTP request and client.svc web service handles the request which internally calls the Server object model to retrieve data from the Content database.

And then it sends the response in either Atom or JavaScript Object Notation (JSON) format which your client application should be able to parse.

REST is a software architecture that uses uniform resource identifiers (URIs) to specify operations against a remote service.

Open Data Protocol (OData) is the protocol that is used along with REST to access many cloud-based services.

Suppose in the client object model we write `List.GetByTitle(listname)` to retrieve a particular list. The same thing if you want to do in REST then it will be like:

```
https://server/site/_api/lists/getbytitle('listname')
```

This is known as REST endpoint.

SharePoint Rest API HTTP commands

If you want to do some CRUD operations in SharePoint using Rest API like adding items to a list, deleting items from a list, or updating items in the SharePoint list, we have to build the Rest endpoints.

To build the Rest API endpoints we need to use the HTTP requests like below:

Want to learn SharePoint? Check out
SharePoint Online Developer Training Course:

<https://www.spguides.com/sharepoint-development-training/>

- If you want to read data from the SharePoint site then you should use HTTP **GET** request.
Example: reading list items
- Similarly, if you want to add or update then you should use an HTTP **POST** request.
Example: Create a list or library
- If you want to update an existing resource you can use the HTTP **PUT** request. Example:
Update a list or library Title or Description
- If you want to Delete then you should do an HTTP **DELETE** request. Example: Delete a
SharePoint List or Library

HTTP operations in SharePoint 2013 REST services

The endpoints in the SharePoint 2013 REST service correspond to the types and members in the SharePoint client object models.

By using HTTP requests, you can use these REST endpoints to perform typical CRUD (Create, Read, Update, and Delete) operations against SharePoint entities, such as lists and sites.

Typically, endpoints that represent Read operations map to HTTP GET commands. Endpoints that represent update operations map to HTTP POST commands, and endpoints that represent update or insert operations map to HTTP PUT commands.

In SharePoint 2013, use POST to create entities such as lists and sites. The SharePoint 2013 REST service supports sending POST commands that include object definitions to endpoints that represent collections.

For example, you could send a POST command that included a new list object definition in ATOM to the following URL, to create a SharePoint list:

`https://<site url>/_api/web/lists`

For POST operations, any properties that are not required are set to their default values. If you attempt to set a read-only property as part of a POST operation, the service returns an exception.

Use PUT and MERGE operations to update existing SharePoint objects. Any service endpoint that represents an object property set operation supports both PUT requests and MERGE requests.

For MERGE requests, setting properties is optional; any properties that you do not explicitly set retain their current property.

For PUT commands, however, any properties you do not explicitly set are set to their default properties. In addition, if you do not specify all required properties in object updates when using HTTP PUT commands, the REST service returns an exception.

Use the HTTP DELETE command against the specific endpoint URL to delete the SharePoint object represented by that endpoint. In the case of recyclable objects, such as lists, files, and list items, this results in a Recycle operation.

To access a specific SharePoint site collection using Rest API, you can use like below:

`https://server/site/_api/site`

To access a specific SharePoint site in Rest API, you can use like below:

`https://server/site/_api/web`

SharePoint 2013 REST endpoints examples

Here are a few rest endpoints for SharePoint 2013/2016/Online. The REST interface exposes all of the SharePoint entities and operations that are available in the other SharePoint client APIs.

One advantage of using REST is that you don't have to add references to any SharePoint libraries or client assemblies. Instead, you make HTTP requests to the appropriate endpoints to retrieve or update SharePoint entities, such as webs, lists, and list items.

Here we will see the different areas, what is corresponding client object model, corresponding REST EndPoint, and the Server Object Model.

List

Client Object Model: ClientContext.Web.Lists

REST EndPoint: https://server/site/_api/web/lists

Server Object Model: SPList

Site

Client Object Model: ClientContext.Site

REST EndPoint: https://server/site/_api/site

Server Object Model: SPSite

Web

Client Object Model: ClientContext.Web

REST EndPoint: https://server/site/_api/web

User Profile

Client Object Model: ClientContext.Web.Lists

REST EndPoint: https://server/site/_api/Sp.UserProfiles.PeopleManager

Search

Client Object Model: ClientContext.Web.Lists

REST EndPoint: https://server/site/_api/search

Publishing

Client Object Model: ClientContext.Web.Lists

REST EndPoint: https://server/site/_api/publishing

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

A few more SharePoint Rest API end point examples:

- https://{site url}/_api/web/title : Get the title of the web or site
- https://{site url}/_api/lists : Retrieve all the lists and libraries presented on the SharePoint site
- [https://{site url}/_api/lists/getbytitle\('listname'\)/items](https://{site url}/_api/lists/getbytitle('listname')/items): Retrieve all the items from a particular list
- [https://{site url}/_api/lists/getbytitle\('listname'\)?select=Title](https://{site url}/_api/lists/getbytitle('listname')?select=Title) : Retrieve Title of the SharePoint list

Advantages of using rest api in SharePoint

The advantages of using REST is that you don't have to add references to any SharePoint 2013 libraries or client assemblies. Instead, you make HTTP requests to the appropriate endpoints to retrieve or update SharePoint entities, such as webs, lists, and list items.

In jsom, SP.js files need to be loaded before the code execution, but in Rest API no such .js files are required.

Another advantages of Rest API in SharePoint is that, Rest API is easier to work and even you can use the browser to see the result and to test them.

How to get and send the form digest value

If you are doing a POST request in Rest API in SharePoint, then you need to pass the form digest value in the X-RequestDigest header. Even if you are using the Rest API code in a script editor web part or if you are using it in a SharePoint Add-in, you need to

You can do this by writing the below lines of code:

```
"X-RequestDigest": $('#__REQUESTDIGEST').val()
```

SharePoint Rest API Examples

Below are few examples of SharePoint Rest API.

Have you checked our SharePoint Developer Training Courses?

<https://sharepointsky.teachable.com/>

Retrieve user display name using Rest API in SharePoint Online/2013

In the first SharePoint 2013 rest api example, we will see how to get a user display name using Rest API in SharePoint.

Below is the Rest API code to retrieve the user display name using Rest API in SharePoint Online.

Here I on a button click I am displaying the display name of the logged in user in SharePoint using Rest API.

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

```
<input type="button" id="btnClick" value="Click Here"/>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script>
$(function(){
$("#btnClick").click(function(){
var userid = _spPageContextInfo.userId;
var requestUri = _spPageContextInfo.webAbsoluteUrl + "/_api/web/getuserbyid(" + userid + ")";
var requestHeaders = { "accept": "application/json;odata=verbose" };
$.ajax({
url: requestUri,
contentType: "application/json;odata=verbose",
headers: requestHeaders,
success: onSuccess,
error: onError
});
function onSuccess(data, request) {
var userinfo = data.d;
alert ('Display Name: '+ userinfo.Title);
}
function onError(error) {
alert("error");
}
});
});
</script>
```

Here Title property displays the user display name, similarly, you can use the Email property to retrieve the Email ID of the logged user and you can also use the LoginName property to retrieve the user's login name in SharePoint Online or SharePoint 2013/2016.

Delete SharePoint 2013/Online list Item using REST API

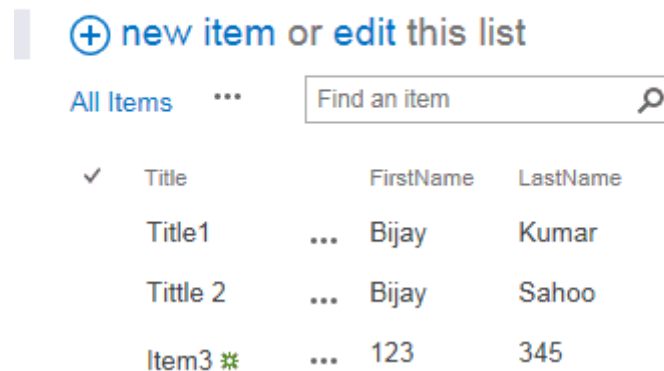
In this SharePoint Rest API example, we will see how we can delete a SharePoint 2013 list item using REST API. If you want to do SharePoint operation without using server-side code then Rest API is very much useful.

Here I have a button in a web part page and when user clicks on that button, the list item will get deleted.

Copy the below code and paste inside a script editor web part. Here we are retrieving the list item by using the getItemById method.

Here you will see there are 3 items in the list like below and by using REST API we will delete the 2nd item from the list.

BijayTestList



| ✓ | Title | ... | FirstName | LastName |
|---|---------|-----|-----------|----------|
| | Title1 | ... | Bijay | Kumar |
| | Title 2 | ... | Bijay | Sahoo |
| | Item3 | ✳ | 123 | 345 |

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script>
$(function(){
$("#btnClick").click(function(){
var requestUri = _spPageContextInfo.webAbsoluteUrl +
"/_api/web/lists/getByTitle('BijayTestList')/getItemById(2)";
$.ajax({
url: requestUri,
type: "DELETE",
headers: {
"accept": "application/json",
"X-RequestDigest": $("#_REQUESTDIGEST").val(),
"IF-MATCH": "*"
},
success: onSuccess,
error: onError
});
});
});
```

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

```
function onSuccess(data) {
alert(data+ ' Item Deleted');
}
function onError
(error) {
alert(JSON.stringify(error));
}
});
});
</script>
```

```
<input type="button" id="btnClick" value="Click Here"/>
```

Now you can see the list will have only two items in the SharePoint Online list.

BijayTestList

@EnjoySharePoint.com

[+ new item](#) or [edit this list](#)

All Items ...

| ✓ | Title | | FirstName | LastName |
|---|---------|-----|-----------|----------|
| | Title1 | ... | Bijay | Kumar |
| | Item3 ✖ | ... | 123 | 345 |

This is how we can delete list item using Rest API in SharePoint Online or SharePoint 2013/2016.

Delete SharePoint 2013/Online list using REST API

In this SharePoint Rest API example, we will discuss how we can delete a SharePoint 2013/Online list using the REST API. In SharePoint 2013/2016/Online you can use the REST API to do lots of operations.

Here I have a button in a web part page and when a user clicks on that button, the SharePoint list will get deleted.

Copy the below code and paste inside a script editor web part. Here we are retrieving the list by using the getByTitle method.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
```

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253


```
<script>
$(function(){
$("#btnClick").click(function(){
var requestUri = _spPageContextInfo.webAbsoluteUrl +
"/_api/web/lists/getByTitle('Columns_List')";
$.ajax({
url: requestUri,
type: "DELETE",
headers: {
"accept":"application/json",
"X-RequestDigest":$("#_REQUESTDIGEST").val(),
"IF-MATCH":"*"
},
success: onSuccess,
error: onError
});

function onSuccess(data) {
alert('List Deleted');
}
function onError
(error) {
alert(error);
}
});
});
</script>
```

```
<input type="button" id="btnClick" value="Click Here"/>
```

This way we can delete a list using Rest API in SharePoint Online or SharePoint 2013/2016.

get alternate languages from language settings in SharePoint using Rest API

This SharePoint Rest API tutorial explains, how to retrieve alternative languages using Rest API in SharePoint Online. You can see the alternative language from Site settings -> then Language Settings which is under "Site Administration" section.

Site Settings ▸ Language Settings ⓘ

Default Language

The default language of the site is specified when the site is first created.

Default Language:

English

Alternate language(s)

Specify the alternate language(s) that this site will support. Users navigating to this site will be able to change the display language of the site to any one of these languages.

Alternate language(s):

- Arabic
- Bulgarian
- German
- Catalan
- Chinese (Traditional)

Below is the full code which I have put in a script editor web part inside a SharePoint web part page. You can use the code according to your requirement.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script>
$(function(){
$("#btnClick").click(function(){
var requestUri = _spPageContextInfo.webAbsoluteUrl + "/_api/web/supportedUILanguageIds";
$.ajax({
url: requestUri,
type: "GET",
headers: {
"accept": "application/json;odata=verbose",
"content-type": "application/json;odata=verbose",
```

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

```
"X-RequestDigest":$("#_REQUESTDIGEST").val()
```

```
},
```

```
success: onSuccess,
```

```
error: onError
```

```
});
```

```
function onSuccess(data) {
```

```
var value = data.d.SupportedUILanguageIds.results;
```

```
alert(value);
```

```
for (var i=0; i<value.length; i++) {
```

```
alert(value[i]);
```

```
}
```

```
}
```

```
function onError(error) {
```

```
alert('Error');
```

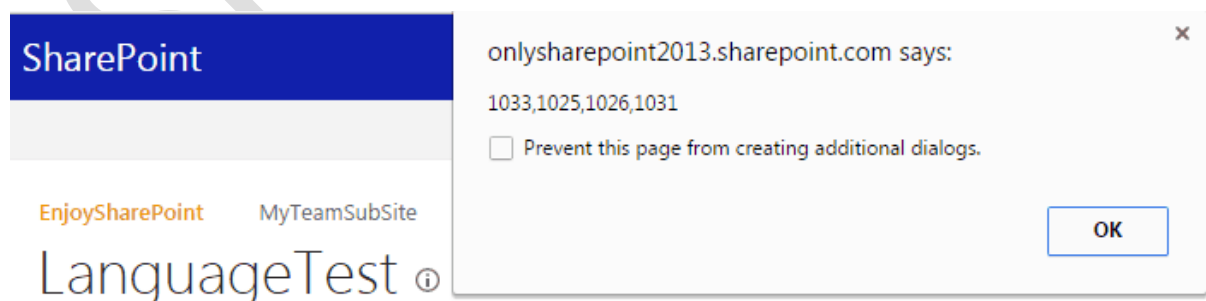
```
}
```

```
});
```

```
});
```

```
</script>
```

```
<input type="button" id="btnClick" value="Get AlternateLanguages"/>
```



Get AlternateLanguages

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

This is how we can get alternate languages from language settings in SharePoint Online or SharePoint 2013/2016 using Rest API.

Retrieve all lists and libraries using Rest API in SharePoint

This Rest API example explains, how to retrieve all lists and libraries using Rest API in SharePoint Online Office 365 or in SharePoint 2013/2016.

Here we will take a button and on click of the button, we will display all lists. Both the HTML code and the rest API code we will put inside a script editor web part which we will add inside a web part page.

Below is the HTML Code:

```
<div>
<input type="button" id="btnSubmit" value="Get All Lists" />
</div>
<div id="divResults"></div>
<div id="divAllLists"></div>
```

Rest API Code:

Here we have first given the jQuery reference. If you are new to jQuery you can check out this article to know Load the Data in JQuery DataTable from SharePoint 2013 List using REST API.

Here also we have used `_spPageContextInfo.webAbsoluteUrl` to get the absolute URL of the site which we have used in the Rest endpoint. You can check out `_spPageContextInfo` javascript or jquery variable in SharePoint 2013 for more information how we can use `_spPageContextInfo`.

Then in `onQuerySucceeded` method, we are retrieving all the lists in for each loop. And we are binding the data into the `<div id="divAllLists"></div>`

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script>
$(function () {
$("#btnSubmit").on("click", function () {
getAllLists();
});
});
```

```
function getAllLists() {
```

TSInfo Technologies (OPC) Pvt Ltd

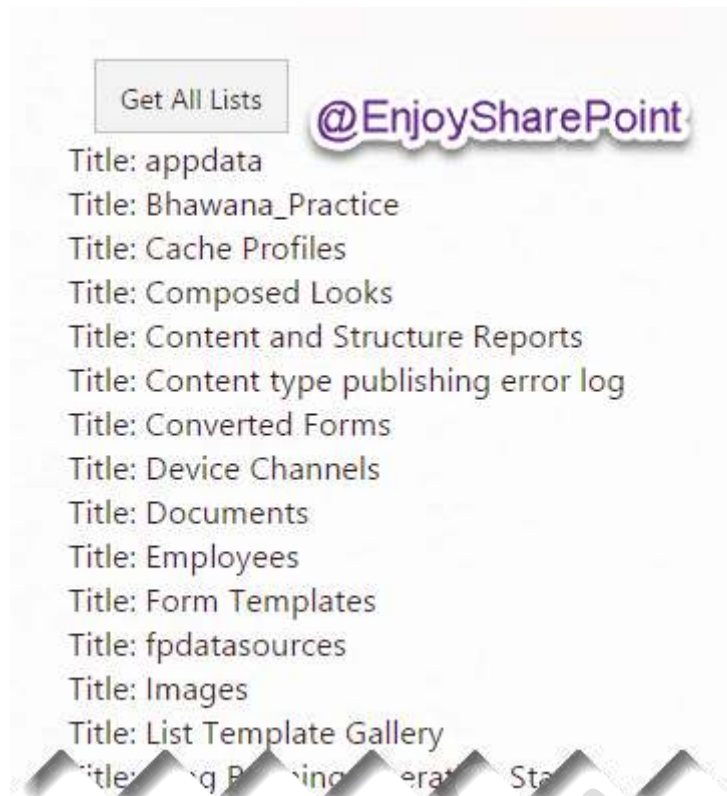
Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253

```
var fullUrl = _spPageContextInfo.webAbsoluteUrl + "/_api/web/lists";
$.ajax({
url: fullUrl,
type: "GET",
headers: {
"accept": "application/json;odata=verbose",
"content-type": "application/json;odata=verbose",
},
success: onQuerySucceeded,
error: onQueryFailed
});
}
```

```
function onQuerySucceeded(data) {
var listItemInfo = "";
$.each(data.d.results, function (key, value) {
listItemInfo += 'Title: ' + value.Title + '<br />';
});
$("#divAllLists").html(listItemInfo);
}
function onQueryFailed() {
alert('Error!');
}
</script>
```

Once you Save the code and click on the button, it will display all the lists and libraries from the SharePoint site.

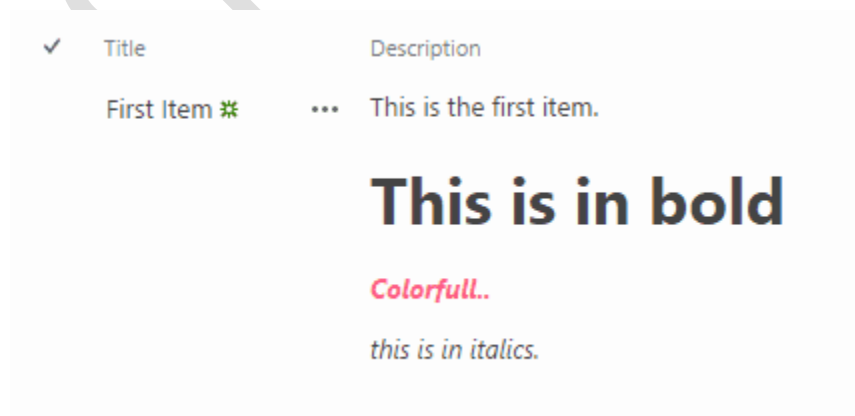


Retrieve Rich Textbox content as plain text using REST API SharePoint 2013

Now we will see how to retrieve Rich textbox content as plain text without HTML tags using REST API in SharePoint 2013.

In SharePoint 2013/2016 or SharePoint Online, I have a list which has a list item that contains a rich textbox field. I want to retrieve that content as plain text using Rest API.

By default my rich textbox content looks like below:



Insert the below code inside and Script editor web part. Then click on the button, you will able to see the result in an alert box.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
```

```
<script>
```

```
$(function(){
```

```
$("#btnClick").click(function(){
```

```
var requestUri = _spPageContextInfo.webAbsoluteUrl +  
"/_api/web/lists/getbytitle('MyTest')/getitembyid(1)";
```

```
var requestHeaders = { "accept": "application/json;odata=verbose" };
```

```
$.ajax({
```

```
url: requestUri,
```

```
contentType: "application/json;odata=verbose",
```

```
headers: requestHeaders,
```

```
success: onSuccess,
```

```
error: onError
```

```
});
```

```
function onSuccess(data, request) {
```

```
var listiteminfo = data.d;
```

```
alert('Content: '+ listiteminfo.Description);
```

```
var pageContentText = $(listiteminfo.Description).text();
```

```
alert (pageContentText);
```

```
}
```

```
function onError(error) {
```

```
alert("error");
```

```
}
```

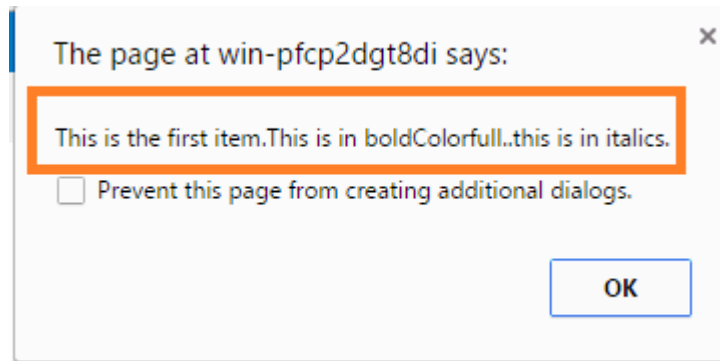
```
});
```

```
});
```

```
</script>
```

```
<input type="button" id="btnClick" value="Click Here"/>
```

After that the result will appear like below:



This is how we can retrieve Rich Textbox content as plain text using REST API SharePoint 2013/2016 or SharePoint Online.

More URL to Refer

<https://www.spguides.com/create-update-delete-sharepoint-list-using-rest-api/>

<https://www.spguides.com/how-to-make-synchronous-rest-api-call-in-sharepoint-online-using-jquery/>

<https://www.spguides.com/rest-api-filter-list-items-created-logged-user-sharepoint-online-sharepoint-2013/>

<https://www.enjoysharepoint.com/create-custom-sharepoint-survey-poll-using-rest-api-and-angularjs/>

<https://www.enjoysharepoint.com/create-list-using-rest-api-in-sharepoint/>

<https://www.enjoysharepoint.com/sharepoint-rest-api-crud-operations/>

<https://www.enjoysharepoint.com/sharepoint-2016-crud-operations-using-rest-api-and-jquery/>

<https://www.enjoysharepoint.com/crud-operations-in-sharepoint-2013-list-item-using-rest-api/>

TSInfo Technologies (OPC) Pvt Ltd

Flat G-04, SJR Residency, Devarabeesanahalli, Bellandur, Bangalore, 560103

Email: info@tsinfotechnologies.com, Phone: +91-9916854253