

 免费电子书

学习

sharepoint

Free unaffiliated eBook created from  
**Stack Overflow contributors.**

#sharepoint

.....	1
<b>1: sharepoint</b> .....	<b>2</b>
.....	2
.....	2
Examples .....	2
SharePoint 2016 .....	2
.....	2
.....	2
.....	3
.....	3
.....	3
SharePoint FrameworkWeb .....	3
SharePoint ULS .....	4
.....	4
.....	4
<b>SPMonitoredScope</b> .....	<b>4</b>
<b>2: REST</b> .....	<b>5</b>
.....	5
<b>RESTURL</b> .....	<b>5</b>
<b>REST</b> .....	<b>5</b>
XMLHttpRequest .....	5
jQuery AJAX .....	5
Examples .....	5
.....	6
.....	6
.....	6
.....	6
.....	7
.....	8
.....	9

SharePointID.....	10
SharePoint 2010 RESTCRUD.....	11
<b>3: SharePoint 2013.....</b>	<b>14</b>
.....	14
Examples.....	14
CSR/.....	14
CSRSharePoint.....	14
CSRNew / Edit Item Form.....	15
CSR.....	19
<b>4: SharePoint.....</b>	<b>21</b>
.....	21
.....	21
Examples.....	21
SharePoint 2013SharePoint 2013JSOM.....	21
<b>5: .....</b>	<b>22</b>
Examples.....	22
SharePoint 2016.....	22
SharePoint 2013.....	22
<b>6: JavaScriptJSOM.....</b>	<b>24</b>
.....	24
Examples.....	24
.....	24
.....	25
.....	25
.....	25
.....	25
.....	26
.....	26
ID.....	26
CAML Query.....	27
.....	27
<b>CAML.....</b>	<b>27</b>



.....	37
.....	38
.....	38
Include.....	38
.....	39
.....	39
.....	39
.....	40
.....	40
.....	41
SharePoint.....	41
.....	41
.....	42
SharePoint.....	42
.....	42
.....	43
SharePoint.....	43
.....	44
.....	44
.....	44
.....	44
.....	45
.....	45
.....	46
Web.....	46
Web.....	46
Web.....	47
.....	47
<b>10: App.....</b>	<b>49</b>
Examples.....	49
.....	49
.....	50
Visual Studio.....	51
.....	55



---

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [sharepoint](#)

It is an unofficial and free sharepoint ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official sharepoint.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

---

# 1: sharepoint

SharePointMicrosoft SharePoint.

- **SharePoint Foundation** SharePointSharePoint 2016
- **SharePoint Server** SharePoint. SharePoint. SharePoint FoundationBI
- **SharePoint Online** SharePoint. ◦

**Office 365**SharePoint OnlineMicrosoftSharePoint.

SharePoint

- SharePoint 2013SharePoint 2016 [SharePoint](#)
- Office 365SharePoint [SharePoint](#)
- SharePoint OnlineSharePointOffice 365 [SharePoint](#)
- SharePoint 2013SharePoint Online [http //www.buckleyplanet.com/2014/06/sharepoint-online-vs-onprem-feature-comparison.html](http://www.buckleyplanet.com/2014/06/sharepoint-online-vs-onprem-feature-comparison.html)

2003	SharePoint Portal Server	2002-07-09
2003	SharePoint Portal Server 2003	20031123
2007	SharePoint Server 2007	2007-01-27
2010	<a href="#">Microsoft SharePoint Server 2010</a>	2010-07-15
2013	<a href="#">Microsoft SharePoint Server 2013</a>	201319
2016	<a href="#">Microsoft SharePoint Server 2016</a>	201651

## Examples

### SharePoint 2016

---

SharePoint 2016SharePoint16. 201654. SharePoint 2016. SharePoint. ◦

---

SharePoint. SharePoint. RAM.

- 644
- 12 - 24 GB RAMprod
- 80GB
- 100GB



- 64Windows Server 2012 R2
- SQL Server 2014SQL Server 2016
- .NET Framework 4.5.2.NET Framework 4.6
- 

SharePointSharePoint。

- 
- ;
  - SharePointSetup.exe
  - 
  - 
  - ""
  - 
  - ""

SharePoint 2016。 "" - >"SharePoint 2016" - >"SharePoint 2016"。

- 
- ;
- - SQL Server;
  - SharePoint\_Config
  - DOMAIN \ user\*
  - ""
- ;
- 
- WebSharePointNTLMNegotiateKerberos
- 
- 
- Central Admin
- COMMONPROGRAMFILES\ Microsoft Shared \ Web Server Extensions \ 16 \ LOG

Web。 Central Admin。

- "" - >" - >"
- ""
- - 
  - DOMAIN \ user
- ""
- - Web
  - web URL;SharePoint
- ""

。

## SharePoint FrameworkWeb

[dev.office.com/sharepoint](https://dev.office.com/sharepoint) SharePoint Framework.

SharePoint Framework SharePoint Office 365 SharePoint Online. SharePoint Framework Web Web SharePoint. SharePoint.

[SharePoint Web Hello World 1](#) hello world. [dev.office.com/github](https://dev.office.com/github).

SharePoint Framework Hello World

1. [Yeoman SharePoint Generator](#).

@ microsoft / SharePoint

2. [Visual Studio Code](#).

3. [gulp SharePoint Workbench Web](#)

4. SharePoint Online

URL' [https://your-sharepoint-site/\\_layouts/workbench.aspx](https://your-sharepoint-site/_layouts/workbench.aspx) '

## SharePoint ULS

SharePoint ULS. .

Microsoft [ULS Viewer](#). .

ID. ID. ULSID.

## SPMonitoredScope

SPMonitoredScope.

```
using (new SPMonitoredScope("Feature Monitor"))
{
    // My code here
}
```

- [ISPScopedPerformanceMonitor](#).

[sharepoint](https://sharepoint) <https://riptutorial.com/zh-CN/sharepoint/topic/950/sharepoint>

## 2: REST

# REST URL

REST API SharePoint 2010 SharePoint 2013 [SharePoint 2010 REST API](#) /\_vti\_bin/ListData.svc  
ListData Web [SharePoint 2013](#) /\_api/lists/\_api/web URL

URL http://server/site serverserver site

.....	SharePoint 2010	SharePoint 2013
	/_vti_bin/ListData.svc/ListName	/_api/lists('ListGuid')
	/_vti_bin/ListData.svc/ListName(1)	/_api/lists('ListGuid')/items(1)
		/_api/web

SharePoint 2013 ListData.svc

## REST

JavaScript XMLHttpRequest jQuery AJAX REST

## XMLHttpRequest

```
var xhr = new XMLHttpRequest();
xhr.open(verb, url, true);
xhr.setRequestHeader("Content-Type", "application/json");
xhr.send(data);
```

## jQuery AJAX

```
$.ajax({
  method: verb,
  url: url,
  headers: { "Content-Type": "application/json" },
  data: data
});
```

AJAX [JavaScript AJAX](#)

## Examples



- "People/GroupsTitle EMail"

- 
- NumLegsType/NumLegs ◦

## JavaScript

```
// webUrl: The url of the site (ex. https://www.contoso.com/sites/animals)
// listTitle: The name of the list you want to query
// selectFields: the specific fields you want to get back
// expandFields: the name of the fields that need to be pulled from lookup tables
// callback: the name of the callback function on success
function getItem(webUrl, listTitle, selectFields, expandFields, callback){
    var endpointUrl = webUrl + "/_api/web/lists/getbytitle('" + listTitle + ")/items";
    endpointUrl+= '?$select=' + selectFields.join(",");
    endpointUrl+= '&$expand=' + expandFields.join(",");
    return executeRequest(endpointUrl, 'GET', callback);
}

function executeRequest(url, method, callback, headers, payload)
{
    if (typeof headers == 'undefined'){
        headers = {};
    }
    headers["Accept"] = "application/json;odata=verbose";
    if(method == "POST") {
        headers["X-RequestDigest"] = $("#__REQUESTDIGEST").val();
    }

    var ajaxOptions =
    {
        url: url,
        type: method,
        contentType: "application/json;odata=verbose",
        headers: headers,
        success: function (data) { callback(data) }
    };
    if(method == "POST") {
        ajaxOptions.data = JSON.stringify(payload);
    }

    return $.ajax(ajaxOptions);
}

// Setup the ajax request by setting all of the arguments to the getItem function
function getAnimals() {
    var url = "https://www.contoso.com/sites/animals";
    var listTitle = "AnimalListing";

    var selectFields = [
        "Title",
        "Age",
        "Value",
        "Type/Title",
        "Type/NumLegs"
    ];

    var expandFields = [
        "Type/Title",
```

```

        "Type/NumLegs"
    ];

    getItems(url, listTitle, selectFields, expandFields, processAnimals);
}

// Callback function
// data: returns the data given by SharePoint
function processAnimals(data) {
    console.log(data);
    // Process data here
}

// Start the entire process
getAnimals();

```

MultiLookupColumnName ID12.

## jQuery AJAX

### 2010

```

var listName = "YourListName";
var lookupList = "LookupListName";
var idOfItemToUpdate = 1;
var url = "/server/site/_vti_bin/ListData.svc/"+listName+"( "+idOfItemToUpdate+" )";
var data = JSON.stringify({
    MultiLookupColumnName:[
        {__metadata:{uri:"http://yoursiteurl/_vti_bin/ListData.svc/"+lookupList+" (1)"}},
        {__metadata:{uri:"http://yoursiteurl/_vti_bin/ListData.svc/"+lookupList+" (2)"}},
    ]
});
$.ajax({
    method: 'POST',
    url: url,
    contentType: 'application/json',
    headers: {
        "X-HTTP-Method" : "MERGE",
        "If-Match" : "*"
    },
    data: data
});

```

### 2013

```

var listGuid = "id-of-list-to-update"; // use list GUID here
var lookupGuid = "id-of-lookup-list"; // use lookup list GUID here
var idOfItemToUpdate = 1;
var url = "/server/site/_api/lists('"+ listGuid + "')/items(" + idOfItemToUpdate + ")";
var data = JSON.stringify({
    MultiLookupColumnName:[
        {__metadata:{uri:"http://yoursiteurl/_api/lists('" + lookupGuid + "')/items(1)"}},
        {__metadata:{uri:"http://yoursiteurl/_api/lists('" + lookupGuid + "')/items(2)"}},
    ]
});
$.ajax({
    method: 'POST',
    url: url,

```

```

contentType: 'application/json',
headers: {
  "X-HTTP-Method" : "MERGE",
  "If-Match" : "*"
},
data: data
});

```

## XMLHttpRequest

### 2010

```

var listName = "YourListName";
var lookupList = "LookupListName";
var idOfItemToUpdate = 1;
var url = "/server/site/_vti_bin/ListData.svc/YourListName("+idOfItemToUpdate+")";
var data = JSON.stringify({
  MultiLookupColumnName:[
    {__metadata:{uri:"http://yoursiteurl/_vti_bin/ListData.svc/"+lookupList+" (1)"}},
    {__metadata:{uri:"http://yoursiteurl/_vti_bin/ListData.svc/"+lookupList+" (2)"}},
  ]
});
var xhr = new XMLHttpRequest();
xhr.open("POST",url,true);
xhr.setRequestHeader("X-HTTP-Method", "MERGE");
xhr.setRequestHeader("If-Match", "*");
xhr.setRequestHeader("Content-Type","application/json");
xhr.send(data);

```

### 2013

```

var listGuid = "id-of-list-to-update";
var lookupGuid = "id-of-lookup-list";
var idOfItemToUpdate = 1;
var url = "/server/site/_api/lists('"+ listGuid + "')/items("+ idOfItemToUpdate + ")";
var data = JSON.stringify({
  MultiLookupColumnName:[
    {__metadata:{uri:"http://yoursiteurl/_api/lists('" + lookupGuid + "')/items(1)"}},
    {__metadata:{uri:"http://yoursiteurl/_api/lists('" + lookupGuid + "')/items(2)"}},
  ]
});
var xhr = new XMLHttpRequest();
xhr.open("POST",url,true);
xhr.setRequestHeader("X-HTTP-Method", "MERGE");
xhr.setRequestHeader("If-Match", "*");
xhr.setRequestHeader("Content-Type","application/json");
xhr.send(data);

```

## REST

1. \$skip=n\$orderbyn
2. \$top=n\$orderby\$skip°

```

var endpointUrl = "/_api/lists('guid')/items"; // SP2010: "/_vti_bin/ListData.svc/ListName";
$.getJSON(
  endpointUrl + "?$orderby=Id&$top=1000",

```

```

function(data){
    processData(data); // you can do something with the results here
    var count = data.d.results.length;
    getNextBatch(count, processData, onComplete); // fetch next page
}
);

function getNextBatch(totalSoFar, processResults, onCompleteCallback){
    $.getJSON(
        endpointUrl + "?$orderby=Id&$skip="+totalSoFar+"&$top=1000",
        function(data){
            var count = data.d.results.length;
            if(count > 0){
                processResults(data); // do something with results
                getNextBatch(totalSoFar+count, callback); // fetch next page
            }else{
                onCompleteCallback();
            }
        }
    );
}

```

## SharePointID

SharePoint REST APIID。

**listName** - 。

**newItemBody** - 。

**var newItemBody = {\_\_metadata{'type':"SP.Data.MyListNameItem'}};**

```

function CreateListItemWithDetails(listName, newItemBody) {

    var item = newItemBody;
    return $.ajax({
        url: _spPageContextInfo.siteAbsoluteUrl + "/_api/web/lists/getbytitle('" + listName +
        "')/items",
        type: "POST",
        contentType: "application/json;odata=verbose",
        data: JSON.stringify(item),
        headers: {
            "Accept": "application/json;odata=verbose",
            "X-RequestDigest": $("#__REQUESTDIGEST").val(),
            "content-Type": "application/json;odata=verbose"
        }
    });
}

CreateListItemWithDetails(listName, newItemBody)
    .then(function(data) {
        //success callback
        var NewlyCreatedItemId = data.d.ID;
    }, function(data) {
        //failure callback
    });

```



# SharePoint 2010 RESTCRUD

## REST

POST HTTP。 URL POST。 application/json。 JSONJavaScript

```
function createListItem(webUrl, listName, itemProperties, success, failure) {

    $.ajax({
        url: webUrl + "/_vti_bin/listdata.svc/" + listName,
        type: "POST",
        processData: false,
        contentType: "application/json;odata=verbose",
        data: JSON.stringify(itemProperties),
        headers: {
            "Accept": "application/json;odata=verbose"
        },
        success: function (data) {
            success(data.d);
        },
        error: function (data) {
            failure(data.responseJSON.error);
        }
    });
}
```

```
var taskProperties = {
    'TaskName': 'Order Approval',
    'AssignedToId': 12
};

createListItem('https://contoso.sharepoint.com/project/', 'Tasks', taskProperties, function(task) {

    console.log('Task' + task.TaskName + ' has been created');
},
function(error) {
    console.log(JSON.stringify(error));
}
);
```

## REST

GET HTTP。 URL GET。 application/json。 JavaScript

```
function getItemById(webUrl, listName, itemId, success, failure) {
    var url = webUrl + "/_vti_bin/listdata.svc/" + listName + "(" + itemId + ")";
    $.ajax({
        url: url,
        method: "GET",
        headers: { "Accept": "application/json; odata=verbose" },
        success: function (data) {
            success(data.d);
        },
        error: function (data) {
            failure(data.responseJSON.error);
        }
    });
}
```

```
}
```

```
getListItemId('https://contoso.sharepoint.com/project/', 'Tasks', 2, function(taskItem) {  
    console.log(taskItem.TaskName);  
},  
function(error) {  
    console.log(JSON.stringify(error));  
}  
);
```

POST HTTP。 MERGEX-HTTP-Method。 URLPOST If-Match ETag\*。 JavaScript

```
function updateListItem(webUrl, listName, itemId, itemProperties, success, failure)  
{  
    getListItemId(webUrl, listName, itemId, function(item) {  
  
        $.ajax({  
            type: 'POST',  
            url: item.__metadata.uri,  
            contentType: 'application/json',  
            processData: false,  
            headers: {  
                "Accept": "application/json;odata=verbose",  
                "X-HTTP-Method": "MERGE",  
                "If-Match": item.__metadata.etag  
            },  
            data: Sys.Serialization.JavaScriptSerializer.serialize(itemProperties),  
            success: function (data) {  
                success(data);  
            },  
            error: function (data) {  
                failure(data);  
            }  
        });  
  
    },  
function(error) {  
    failure(error);  
});  
}
```

```
var taskProperties = {  
    'TaskName': 'Approval',  
    'AssignedToId': 12  
};  
  
updateListItem('https://contoso.sharepoint.com/project/', 'Tasks', 2, taskProperties, function(item) {  
  
    console.log('Task has been updated');  
},  
function(error) {  
    console.log(JSON.stringify(error));  
}  
);
```

POST

```
function deleteListItem(webUrl, listName, itemId, success, failure) {
    getListItemById(webUrl, listName, itemId, function (item) {
        $.ajax({
            url: item.__metadata.uri,
            type: "POST",
            headers: {
                "Accept": "application/json;odata=verbose",
                "X-Http-Method": "DELETE",
                "If-Match": item.__metadata.etag
            },
            success: function (data) {
                success();
            },
            error: function (data) {
                failure(data.responseJSON.error);
            }
        });
    },
    function (error) {
        failure(error);
    });
}
```

```
deleteListItem('https://contoso.sharepoint.com/project/', 'Tasks', 3, function () {
    console.log('Task has been deleted');
},
function (error) {
    console.log(JSON.stringify(error));
}
);
```

REST <https://riptutorial.com/zh-CN/sharepoint/topic/3045/rest>

# 3: SharePoint 2013

CSRSharePoint 2013。 SharePoint。 。 HTMLJavaScriptXSLT。

## Examples

### CSR/

CSR“ ID ”“ TitleLinkTitle ”。

### 1JS

```
(function () {  
  
    function registerRenderer() {  
        var ctxForm = {};  
        ctxForm.Templates = {};  
  
        ctxForm.Templates = {  
            Fields : {  
                'LinkTitle': { //----- Change Hyperlink of LinkTitle  
                    View : function (ctx) {  
                        var url = String.format('{0}?ID={1}',  
"/sites/Lists/testlist/EditItem.aspx", ctx.CurrentItem.ID);  
                        return String.format('<a href="{0}" onclick="EditItem2(event,  
{0})\');return false;">{1}</a>', url, ctx.CurrentItem.Title);  
                    }  
                },  
                'ID' : { //----- Change Hyperlink from ID field  
                    View : function (ctx) {  
                        var url = String.format('{0}?ID={1}',  
"/IssueTracker/Lists/testlist/DisplayItem.aspx", ctx.CurrentItem.ID);  
                        return String.format('<a href="{0}" onclick="EditItem2(event,  
{0})\');return false;">{1}</a>', url, ctx.CurrentItem.ID);  
                    }  
                },  
            }  
        };  
        SPClientTemplates.TemplateManager.RegisterTemplateOverrides(ctxForm);  
    }  
    ExecuteOrDelayUntilScriptLoaded(registerRenderer, 'clienttemplates.js');  
})();
```

### 2List ViewWebJS Linkjs~sitecollection / SiteAssets / CSRCodeFile.js

JSlink。 “~sitecollection / YourJSfFilePath”。

### 3AppyDone

### CSRSharePoint。

## CSRSharePoint

```
(function () {

    function RemoveFields(ctx) {
        var fieldName = "Date"; // here Date is field or column name to be hide
        var header = document.querySelectorAll("[displayname=" + fieldName +
        "]")[0].parentNode;
        var index = [].slice.call(header.parentNode.children).indexOf(header) + 1;
        header.style.display = "none";
        for (var i = 0, cells = document.querySelectorAll("td:nth-child(" + index + ")"); i <
        cells.length; i++) {
            cells[i].style.display = "none";
        }
    }

    function registerRenderer() {
        var ctxForm = {};
        ctxForm.Templates = {};
        ctxForm.OnPostRender = RemoveFields;
        SPClientTemplates.TemplateManager.RegisterTemplateOverrides(ctxForm);
    }
    ExecuteOrDelayUntilScriptLoaded(registerRenderer, 'clienttemplates.js');
})();
```

## CSRNew / Edit Item Form

SharePoint。 “/”CSR。

- 
- ID
- 
- “”

## 1JSCSRValidations.jsJS

```
(function () {

    // Create object that have the context information about the field that we want to
    change it's output render
    var fieldContext = {};
    fieldContext.Templates = {};
    fieldContext.Templates.Fields = {
        // Apply the new rendering for Email field on New and Edit Forms
        "Title": {
            "NewForm": titleFieldTemplate,
            "EditForm": titleFieldTemplate
        },
        "Full_x0020_Name": {
            "NewForm": fullNameFieldTemplate,
            "EditForm": fullNameFieldTemplate
        },
        "Email": {
            "NewForm": emailFieldTemplate,
            "EditForm": emailFieldTemplate
        },
    },
```

```

        "Mobile_x0020_Phone": {
            "NewForm": mobilePhoneFieldTemplate,
            "EditForm": mobilePhoneFieldTemplate
        }
    };

    SPClientTemplates.TemplateManager.RegisterTemplateOverrides(fieldContext);

})();

// This function provides the rendering logic
function emailFieldTemplate(ctx) {

    var formCtx = SPClientTemplates.Utility.GetFormContextForCurrentField(ctx);

    // Register a callback just before submit.
    formCtx.registerGetValueCallback(formCtx.fieldName, function () {
        return document.getElementById('inpEmail').value;
    });

    //Create container for various validations
    var validators = new SPClientForms.ClientValidation.ValidatorSet();
    validators.RegisterValidator(new emailValidator());

    // Validation failure handler.
    formCtx.registerValidationErrorCallback(formCtx.fieldName, emailOnError);

    formCtx.registerClientValidator(formCtx.fieldName, validators);

    return "<span dir='none'><input type='text' value='" + formCtx.fieldValue + "'
maxlength='255' id='inpEmail' class='ms-long'> \ <br><span id='spnEmailError' class='ms-
formvalidation ms-csrformvalidation'></span></span>";
}

// Custom validation object to validate email format
emailValidator = function () {
    emailValidator.prototype.Validate = function (value) {
        var isError = false;
        var errorMessage = "";

        //Email format Regex expression
        //var emailRejex = /\S+@\S+\.\S+/;
        var emailRejex = /^((([^\<>() []]HYPERLINK
"\.\.,;:\s@"\.\.,;:\s@"+([\.[^\<>() []]HYPERLINK "\.\.,;:\s@"\.\.,;:\s@"+)*|(\".+\\"))@((\[[0-
9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\)|(([a-zA-Z\0-9]+\.)+[a-zA-Z]{2,}))$)/;

        if (value.trim() == "") {
            isError = true;
            errorMessage = "You must specify a value for this required field.";
        }else if (!emailRejex.test(value) && value.trim()) {
            isError = true;
            errorMessage = "Please enter valid email address";
        }

        //Send error message to error callback function (emailOnError)
        return new SPClientForms.ClientValidation.ValidationResult(isError, errorMessage);

    };
};

// Add error message to spnError element under the input field element

```

```

function emailOnError(error) {
    document.getElementById("spnEmailError").innerHTML = "<span role='alert'" +
error.errorMessage + "</span>";
}

// This function provides the rendering logic
function titleFieldTemplate(ctx) {

    var formCtx = SPClientTemplates.Utility.GetFormContextForCurrentField(ctx);
    // Register a callback just before submit.

    formCtx.registerGetValueCallback(formCtx.fieldName, function () {
        return document.getElementById('inpTitle').value;
    });

    //Create container for various validations
    var validators = new SPClientForms.ClientValidation.ValidatorSet();
    validators.RegisterValidator(new titleValidator());

    // Validation failure handler.
    formCtx.registerValidationErrorCallback(formCtx.fieldName, titleOnError);

    formCtx.registerClientValidator(formCtx.fieldName, validators);

    return "<span dir='none'" + "<input type='text' value='" + formCtx.fieldValue + "'
maxlength='255' id='inpTitle' class='ms-long'" + "\ <br>" + "<span id='spnTitleError' class='ms-
formvalidation ms-csrformvalidation'" + "</span>" + "</span>";
}

// Custom validation object to validate title format
titleValidator = function () {
    titleValidator.prototype.Validate = function (value) {
        var isError = false;
        var errorMessage = "";

        if (value.trim() == "") {
            isError = true;
            errorMessage = "You must specify a value for this required field.";
        }

        //Send error message to error callback function (titleOnError)
        return new SPClientForms.ClientValidation.ValidationResult(isError, errorMessage);
    };
};

// Add error message to spnError element under the input field element
function titleOnError(error) {
    document.getElementById("spnTitleError").innerHTML = "<span role='alert'" +
error.errorMessage + "</span>";
}

// This function provides the rendering logic
function mobilePhoneFieldTemplate(ctx) {

    var formCtx = SPClientTemplates.Utility.GetFormContextForCurrentField(ctx);

    // Register a callback just before submit.
    formCtx.registerGetValueCallback(formCtx.fieldName, function () {
        return document.getElementById('inpMobilePhone').value;
    });
};

```

```

//Create container for various validations
var validators = new SPClientForms.ClientValidation.ValidatorSet();
validators.RegisterValidator(new mobilePhoneValidator());

// Validation failure handler.
formCtx.registerValidationErrorCallback(formCtx.fieldName, mobilePhoneOnError);

formCtx.registerClientValidator(formCtx.fieldName, validators);

return "<span dir='none'><input type='text' value='" + formCtx.fieldValue + "'
maxlength='255' id='inpMobilePhone' class='ms-long'> \ <br><span id='spnMobilePhoneError'
class='ms-formvalidation ms-csrfvalidation'></span></span>";
}

// Custom validation object to validate mobilePhone format
mobilePhoneValidator = function () {
    mobilePhoneValidator.prototype.Validate = function (value) {
        var isError = false;
        var errorMessage = "";

        //MobilePhone format Regex expression
        //var mobilePhoneRejex = /\S+@\S+\.\S+\/;
        var mobilePhoneRejex = /^[0-9]+$/;

        if (value.trim() == "") {
            isError = true;
            errorMessage = "You must specify a value for this required field.";
        }else if (!mobilePhoneRejex.test(value) && value.trim()) {
            isError = true;
            errorMessage = "Please enter valid mobile phone number";
        }

        //Send error message to error callback function (mobilePhoneOnError)
        return new SPClientForms.ClientValidation.ValidationResult(isError, errorMessage);
    };
};

// Add error message to spnError element under the input field element
function mobilePhoneOnError(error) {
    document.getElementById("spnMobilePhoneError").innerHTML = "<span role='alert'>" +
error.errorMessage + "</span>";
}

// This function provides the rendering logic
function fullNameFieldTemplate(ctx) {

    var formCtx = SPClientTemplates.Utility.GetFormContextForCurrentField(ctx);

    // Register a callback just before submit.
    formCtx.registerGetValueCallback(formCtx.fieldName, function () {
        return document.getElementById('inpFullName').value;
    });

    //Create container for various validations
    var validators = new SPClientForms.ClientValidation.ValidatorSet();
    validators.RegisterValidator(new fullNameValidator());

    // Validation failure handler.
    formCtx.registerValidationErrorCallback(formCtx.fieldName, fullNameOnError);
}

```



```

        formCtx.registerClientValidator(formCtx.fieldName, validators);

        return "<span dir='none'><input type='text' value='" + formCtx.fieldValue + "'
maxlength='255' id='inpFullName' class='ms-long'> \ <br><span id='spnFullNameError' class='ms-
formvalidation ms-csrformvalidation'></span></span>";
    }

    // Custom validation object to validate fullName format
    fullNameValidator = function () {
        fullNameValidator.prototype.Validate = function (value) {
            var isError = false;
            var errorMessage = "";

            //FullName format Regex expression
            var fullNameRejex = /^[a-z ,.'-]+$ /i;

            if (value.trim() == "") {
                isError = true;
                errorMessage = "You must specify a value for this required field.";
            } else if (!fullNameRejex.test(value) && value.trim()) {
                isError = true;
                errorMessage = "Please enter valid name";
            }

            //Send error message to error callback function (fullNameOnError)
            return new SPClientForms.ClientValidation.ValidationResult(isError, errorMessage);

        };
    };

    // Add error message to spnError element under the input field element
    function fullNameOnError(error) {
        document.getElementById("spnFullNameError").innerHTML = "<span role='alert'>" +
error.errorMessage + "</span>";
    }

```

## 2. Web。

3Web - > JS - >js~sitecollection / SiteAssets / CSRValidations.js

## 4Web。

## CSR

“IsApprovalNeeded””””。

## CSR。

.....

```

(function () {

    function preTaskFormRenderer(renderCtx) {
        modifyColumns(renderCtx);
    }

```

```
function modifyColumns(renderCtx)
{
    var arrayLength= renderCtx.ListSchema.Field.length;
    for (var i=0; i < arrayLength;i++)
    {
        if(renderCtx.ListSchema.Field[i].DisplayName == 'IsApprovalNeeded')
        {
            var newTitle= "Is Approval Needed?";
            var linkTitleField = renderCtx.ListSchema.Field[i];
            linkTitleField.DisplayName = newTitle;
        }
    }
}

function registerRenderer()
{
    var ctxForm = {};
    ctxForm.Templates = {};
    ctxForm.OnPreRender = preTaskFormRenderer;
    SPClientTemplates.TemplateManager.RegisterTemplateOverrides(ctxForm);
}

ExecuteOrDelayUntilScriptLoaded(registerRenderer, 'clienttemplates.js');

})();
```

SharePoint 2013 <https://riptutorial.com/zh-CN/sharepoint/topic/8317/sharepoint-2013>

# 4: SharePoint

SharePoint

[http //www.letsharepoint.com/what-is-user-information-list-in-sharepoint-2013/](http://www.letsharepoint.com/what-is-user-information-list-in-sharepoint-2013/)

## Examples

SharePoint 2013 SharePoint 2013 JSOM

SharePoint 2013 SharePoint 2013 JSOM

JSOM Javascript User Profile Service UPS。 UPS。

- AccountName FirstName LastName WorkEmail。

- Active Directory。

- SharePoint ""。

JSOM "" JSOM CSOM REST

\*/。

JSOM

SharePoint -

Visual Studio 2013 New Project "App for SharePoint 2013"。 SharePointmu SharePoint Online URL  
SharePoint。 。

3./。

4. "Default.aspx" JavaScript。

SharePoint <https://riptutorial.com/zh-CN/sharepoint/topic/9876/sharepoint>

# 5:

## Examples

### SharePoint 2016

16.0.4366.1000	20164	SharePoint Server 2016
16.0.4336.1000	RTM	SharePoint Server 2016
16.0.4327.1000		SharePoint Server 2016
16.0.4266.1001	16.0.4306.1002 Beta 2	SharePoint Server 2016

### SharePoint 2013

15.0.4623.1001	20146
15.0.4631.1001	20147
15.0.4641.1001	20148
15.0.4649.1001	20149
15.0.4659.1001	201410
15.0.4667.1000	201411
15.0.4675.1000	201412
15.0.4693.1001	20152
15.0.4701.1001	20153
15.0.4711.1000	20154SP1 REQ
15.0.4719.1002	20155
15.0.4727.1001	20156
15.0.4737.1000	20157
15.0.4745.1000	20158
15.0.4753.1003	20159

15.0.4763.1002	201510
15.0.4771.1000	201511
15.0.4779.1000	201512
15.0.4787.1000	MS16-004
15.0.4787.1000	Januar 2016
15.0.4797.1001	20162
15.0.4805.1000	20163
15.0.4815.1000	20164
15.0.4823.1003	20165
15.0.4833.1000	20166

[SharePoint 2013 Build NumberCU's](#)

<https://riptutorial.com/zh-CN/sharepoint/topic/2737/>

---

# 6: JavaScriptJSOM

JavaScriptSharePoint 2010。 Web。

## SharePointJavaScript

SharePoint 2013JavaScriptWeb。

SharePoint 2010Content Editor Web“HTML。

SPSharePoint 2013 。

SharePoint 2010 JavaScript。

## JSOM

### JavaScript

1. ClientContext。
2. ClientContextSharePoint。
3. 。
4. loadClientContext。
5. ClientContextexecuteQueryAsync。
6. 。

JSOMSharePointWeb [REST.NET](#) 。

## Examples

```
function getContentTypes(site_url,name_of_the_library){
    var ctx = new SP.ClientContext(site_url);
    var web = ctx.get_web();
    list = web.get_lists().getByTitle(name_of_the_library);

    // You can include any property of the SP.ContentType object (sp.js), for this example we
    are just getting the name
    ctx.load(list, 'ContentTypes.Include(Name)');
    ctx.executeQueryAsync(onQuerySucceeded, onQueryFailed);
}

function onQuerySucceeded(sender, args) {
    // var list is the one that we used in function "getContentTypes"
    var contentTypesEnumerator = (list.get_contentTypes()).getEnumerator();

    while (contentTypesEnumerator.moveNext()) {
        var contentType = contentTypesEnumerator.get_current();
        alert(contentType.get_name());
    }
}

function onQueryFailed(sender, args) {
```

```
    alert('Request failed. ' + args.get_message() + '\n' + args.get_stackTrace());
}
```

```
SP.SOD.executeOrDelayUntilScriptLoaded( function(){ deleteItem(1); }, "sp.js");
```

```
function deleteItem(id){
    var clientContext = new SP.ClientContext();
    var list = clientContext.get_web().get_lists().getByTitle("List Title");
    var item = list.getItemById(id);
    item.deleteObject();
    clientContext.executeQueryAsync(function(){
        alert("Item #"+id+" deleted successfully!");
    },function(sender,args){alert(args.get_message());});
}
```

```
SP.SOD.executeOrDelayUntilScriptLoaded(createItem,"sp.js");
```

```
function createItem(){
    var clientContext = new SP.ClientContext();
    var list = clientContext.get_web().get_lists().getByTitle("List Title");
    var newItem = list.addItem();
    newItem.set_item("Title","Example Title");
    newItem.update();
    clientContext.load(newItem); // only needed to retrieve info from newly created item
    clientContext.executeQueryAsync(function(){
        var itemId = newItem.get_item("ID");
        alert("Item #"+itemId+" Created Successfully!");
    },function(sender,args){
        alert(args.get_message());
    });
}
```

1. addItem
2. set\_item
3. update
4. executeQueryAsync

load◦ ◦

◦ ListItemCreationInformationunderlyingObjectTypeSP.FileSystemObjectType.folder leafName◦

addItem◦

```
// ...
var itemCreateInfo = new SP.ListItemCreationInformation();
itemCreateInfo.set_underlyingObjectType(SP.FileSystemObjectType.folder);
itemCreateInfo.set_leafName(folderName);
var newItem = list.addItem(itemCreateInfo);
// ...
```

ClientContextexecuteQueryAsync◦

```

SP.SOD.executeOrDelayUntilScriptLoaded(createFolder,"sp.js");

function createFolder(){
    var now = new Date();
    var timeStamp = now.getYear() + "-" + (now.getMonth()+1) + "-" + now.getDate()
        + "T" + now.getHours()+"_"+now.getMinutes()+"
"+now.getSeconds()+"_"+now.getMilliseconds();
    var clientContext = new SP.ClientContext();
    var list = clientContext.get_web().get_lists().getByTitle("Library Title");
    var itemCreateInfo = new SP.ListItemCreationInformation();
    itemCreateInfo.set_underlyingObjectType(SP.FileSystemObjectType.folder);
    itemCreateInfo.set_leafName(timeStamp);
    var newItem = list.addItem(itemCreateInfo);
    newItem.update();
    clientContext.load(newItem);
    var rootFolder = list.get_rootFolder(); // Note: use a list's root folder to determine its
server relative URL
    clientContext.load(rootFolder);
    clientContext.executeQueryAsync(function(){
        var itemId = newItem.get_item("ID");
        var name = newItem.get_item("FileLeafRef");
        SP.UI.ModalDialog.showModalDialog(
            {
                title: "Folder \""+name+"\" (#"+itemId+") Created Successfully!",
                url: rootFolder.get_serverRelativeUrl() + "/" + name
            }
        );
    },function(sender,args){alert(args.get_message());});
}

```

```

SP.SOD.executeOrDelayUntilScriptLoaded(showUserInfo,"sp.js");

function showUserInfo(){
    var clientContext = new SP.ClientContext();
    var user = clientContext.get_web().get_currentUser();
    clientContext.load(user);
    clientContext.executeQueryAsync(function(){
        var details = "ID: "+user.get_id()+"\n"+
            "Title: "+user.get_title()+"\n"+
            "Login: "+user.get_loginName()+"\n"+
            "Email: "+user.get_email();
        alert(details);
    },function(sender,args){alert(args.get_message());});
}

```

## ID

```

SP.SOD.executeOrDelayUntilScriptLoaded(myFunction,"sp.js");

function myFunction(){
    var clientContext = new SP.ClientContext();
    var list = clientContext.get_web().get_lists().getByTitle("List Title");
    var item = list.getItemById(1); // get item with ID == 1
    clientContext.load(item);
    clientContext.executeQueryAsync(
        function(){ // onSuccess

```



```

        var title = item.get_item("Title");
        alert(title);
    },
    function(sender, args) { // onError
        alert(args.get_message());
    }
    );
}

```

## CAML Query

### SP.CamlQuery.set\_viewXml CAML

```

SP.SOD.executeOrDelayUntilScriptLoaded(showListItems, "core.js");

function showListItems() {
    var clientContext = new SP.ClientContext();
    var list = clientContext.get_web().get_lists().getByTitle("List Title");
    var camlQuery = new SP.CamlQuery();
    camlQuery.set_viewXml(
        "<View><Query>" +
        "  <Where>" +
        "    <Eq><FieldRef Name=\"Title\"/><Value Type=\"Text\">Value</Value></Eq>" +
        "  </Where>" +
        "  <OrderBy><FieldRef Name=\"Modified\" Ascending=\"FALSE\"/></OrderBy>" +
        "</Query>" +
        // "<RowLimit>5000</RowLimit>" +
        "</View>");
    var items = list.getItems(camlQuery);
    clientContext.load(items);
    clientContext.executeQueryAsync(function() {
        var itemArray = [];
        var itemEnumerator = items.getEnumerator();
        while(itemEnumerator.moveNext()) {
            var item = itemEnumerator.get_current();
            var id = item.get_item("ID");
            var title = item.get_item("Title");
            itemArray.push(id + ": " + title);
        }
        alert("ID: Title\n"+itemArray.join("\n"));
    }, function(sender, args) { alert(args.get_message()); });
}

```

## CAML

### CAML.RowLimit

get\_listItemCollectionPosition SP.CamlQuery.set\_listItemCollectionPosition

```

SP.SOD.executeOrDelayUntilScriptLoaded(showListItems, "sp.js");

function showListItems() {
    var itemArray = [];

```

```

var clientContext = new SP.ClientContext();
var list = clientContext.get_web().get_lists().getByTitle("List Title");
var viewXml =
    "<View><Query>" +
        "<OrderBy><FieldRef Name=\"Modified\" Ascending=\"FALSE\"/></OrderBy>" +
    "</Query>" +
        "<RowLimit>1</RowLimit>" +
    "</View>";
var camlQuery = new SP.CamlQuery();
camlQuery.set_viewXml(viewXml);
var items = list.getItems(camlQuery);
clientContext.load(items);
clientContext.executeQueryAsync(loadResults, showError);

function loadResults(){
    var resultsFound = false;
    var itemEnumerator = items.getEnumerator();
    while(itemEnumerator.moveNext()){
        var item = itemEnumerator.get_current();
        var id = item.get_item("ID");
        var title = item.get_item("Title");
        itemArray.push(id + ": " + title);
    }
    var pos = items.get_listItemCollectionPosition();// <- get position
    if(pos !== null){ // <-- position is null when no more results are returned
        if(confirm("Results so far: \nID: Title\n"+itemArray.join("\n"))){
            camlQuery = new SP.CamlQuery();
            camlQuery.set_listItemCollectionPosition(pos);// <- set position for next
batch
            camlQuery.set_viewXml(viewXml);
            items = list.getItems(camlQuery);
            clientContext.load(items);
            clientContext.executeQueryAsync(loadResults, showError);
        }
    }else{
        alert("Total Results: \nID: Title\n"+itemArray.join("\n")); // <- display when no
more results
    }
}
function showError(sender, args){
    alert(args.get_message());
}
}

```

JavaScriptJSOM <https://riptutorial.com/zh-CN/sharepoint/topic/1316/javascript-jsom->

# 7: JavaScript

- `var options = SP.UI.$create_DialogOptions;`
- `var modalDialog = SP.UI.ModalDialog.showModalDialogoptions;`

	URL。 <code>urlhtml</code> 。 <code>urlhtml</code> 。
HTML	HTML。
X	x。
ÿ	y。
	◦ <b>autosizefalse</b> 768px
	◦ 576px
allowMaximize	“ ”。
showMaximized	◦
showClose	“ ”。
	◦
dialogReturnValueCallback	◦ <code>SP.UI.DialogResultDialogResultreturnValue</code> 。
ARGS	◦

`SP.UI.ModalDialog` SharePoint 2010 [JavaScript](#) SharePoint2013Office3652016。

- [SP.UI.ModalDialog.showModalDialogMSDN](#)
- [SPDNUI.DialogResultMSDN](#)

## Examples

```
SP.SOD.executeOrDelayUntilScriptLoaded(showDialog, "sp.js");

function showDialog(){
    var options = SP.UI.$create_DialogOptions();
    options.url = "/mySite/lists/myList/NewForm.aspx";
    options.dialogReturnValueCallback = myCallBackFunction;
    SP.UI.ModalDialog.showModalDialog(options);
    function myCallBackFunction(result,data){
        switch(result){
```

```

        case SP.UI.DialogResult.invalid:
            alert("The dialog result was invalid");
            break;
        case SP.UI.DialogResult.cancel:
            alert("You clicked cancel or close");
            break;
        case SP.UI.DialogResult.OK:
            alert("You clicked OK, creating an item in the list.");
            break;
    }
}
}

```

```

SP.SOD.executeOrDelayUntilScriptLoaded(showDialog, "sp.js");

function showDialog(){
    SP.UI.ModalDialog.showModalDialog(
        { url: "/org/it/web/wik/Lists/ExampleCode/DispForm.aspx?ID=6" }
    );
}

```

```

SP.SOD.executeOrDelayUntilScriptLoaded(showDialog, "sp.js");

function showDialog(){
    var dialogOptions = SP.UI.$create_DialogOptions();
    dialogOptions.title = "Your Title Here!";
    var dummyElement = document.createElement("div");
    dummyElement.style.textAlign = "center";
    dummyElement.appendChild(document.createElement("br"));
    dummyElement.appendChild(document.createTextNode("Some beautifully crafted text.));
    dummyElement.appendChild(document.createElement("br"));
    dialogOptions.html = dummyElement;
    SP.UI.ModalDialog.showModalDialog(dialogOptions);
}

```

JavaScript <https://riptutorial.com/zh-CN/sharepoint/topic/6868/javascript>

---

# 8:

---

SharePoint ◦ SPSite UI/RootWeb ◦ Web SPWeb UI/ SPList webpart/ SPListItem ◦

---

- SharePoint Visual Studio™ Microsoft.SharePoint.
- SharePoint Windows Server.
- SharePoint.

## Examples

### Hello World

2013

SharePoint 2013 64-bit/.

**CPUx64.**

```
using System;
using Microsoft.SharePoint;

namespace StackOverflow
{
    class Samples
    {
        static void Main()
        {
            using (SPSite site = new SPSite("http://server/sites/siteCollection"))
            using (SPWeb web = site.OpenWeb())
            {
                Console.WriteLine("Title: {0} Description: {1}", web.Title, web.Description);
            }
        }
    }
}
```

## SharePoint

### SharePoint WebPowerShell

```
$wacoll = get-spwebapplication
foreach($wa in $wacoll){
    if($wa.IsAdministrationWebApplication -eq $false){
        foreach($site in $wa.Sites){
            foreach($web in $site.AllWebs){
                # your code here
                $web.Dispose()
            }
        }
    }
}
```

```

        $site.Dispose()
    }
}
}

```

```

using (SPSite site = new SPSite("http://server/sites/siteCollection"))
using (SPWeb web = site.OpenWeb())
{
    SPList list = web.Lists["Some list"];

    // It is always better and faster to query list items with GetItems method with
    // empty SPQuery object than to use Items property
    SPListItemCollection items = list.GetItems(new SPQuery());
    foreach (SPListItem item in items)
    {
        // Do some operation with item
    }
}

```

```

using (SPSite site = new SPSite("http://server/sites/siteCollection"))
using (SPWeb web = site.OpenWeb())
{
    SPList list = web.Lists["Some list"];
    SPQuery query = new SPQuery()
    {
        RowLimit = 100
    };

    do
    {
        SPListItemCollection items = list.GetItems(query);
        foreach (SPListItem item in items)
        {
            // Do some operation with item
        }

        // Assign current position to SPQuery object
        query.ListItemCollectionPosition = items.ListItemCollectionPosition;
    } while (query.ListItemCollectionPosition != null);
}

```

```

using (SPSite site = new SPSite("http://server/sites/siteCollection"))
using (SPWeb web = site.OpenWeb())
{
    string listUrl = string.Format("{0}{1}", web.ServerRelativeUrl, "Lists/SomeList");
    SPList list = web.GetList(listUrl);
}

```

。 。 。 “ ” 。 。

“ ” 。

```

using (SPSite site = new SPSite("http://server/sites/siteCollection"))
using (SPWeb web = site.OpenWeb())
{
    SPList list = web.Lists["Announcements"];
}

```

```
SPListItem item = list.AddItem();  
item[SPBuiltInFieldId.Title] = "My new item";  
item[SPBuiltInFieldId.Body] = "Hello World!";  
item.Update();  
}
```

<https://riptutorial.com/zh-CN/sharepoint/topic/7543/-->

# 9: CSOM

- [MSDN](#)。
- .NETDLLMicrosoft.SharePoint.Client.dllMicrosoft.SharePoint.Client.Runtime.dll。 ProgramFiles\ Common Files \ Microsoft Shared \ web server extensions \ 16 \ ISAPI SharePoint。
- Microsoft.SharePointOnline.CSOM NuGet“on prem”SP O365。
- clientContext.LoadclientContext.ExecuteQuery。 [LoadExecuteQuery](#)

## Examples

### Hello world

SharePointSPSiteSSOMCSOMWebSPWebSSOMWebCSOM。 UI。 WebUI。 /。 sitesMyServer Web。

```
using System;
using Microsoft.SharePoint.Client;

namespace Microsoft.SDK.SharePointServices.Samples
{
    class RetrieveWebsite
    {
        static void Main()
        {
            // This is the URL of the target web we are interested in.
            string siteUrl = "http://MyServer/sites/MySiteCollection";
            // The client context is allows us to queue up requests for the server
            // Note that the context can only ask questions about the site it is created for
            using (ClientContext clientContext = new ClientContext(siteUrl))
            {
                // To make it easier to read the code, pull the target web
                // context off of the client context and store in a variable
                Web oWebsite = clientContext.Web;
                // Tell the client context we want to request information about the
                // Web from the server
                clientContext.Load(oWebsite);
                // After we are done creating the batch of information we need from the sever,
                // request the data from SharePoint
                clientContext.ExecuteQuery();
                // Print the results of the query
                Console.WriteLine("Title: {0} Description: {1}", oWebsite.Title,
oWebsite.Description);
            }
        }
    }
}
```

```
ClientContext clientContext = new ClientContext(siteUrl);
```



```
Web oWebsite = clientContext.Web;
clientContext.Load(oWebsite);
clientContext.ExecuteQuery();
Console.WriteLine("Title: {0} Description: {1}", oWebsite.Title, oWebsite.Description);
```

o

```
ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;
clientContext.Load(
    oWebsite,
    website => website.Title,
    website => website.Created);
clientContext.ExecuteQuery();
Console.WriteLine("Title: {0} Created: {1}", oWebsite.Title, oWebsite.Created);
```

o

```
ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = context.Web;
oWebsite.Title = "Updated Web Site";
oWebsite.Description = "This is an updated Web site.";
oWebsite.Update();
clientContext.ExecuteQuery();
```

o

```
string siteUrl = "http://MyServer/sites/MySiteCollection";
string blogDescription = "A new blog Web site.";
int blogLanguage = 1033;
string blogTitle = "Blog Web Site";
string blogUrl = "blogwebsite";
bool blogPermissions = false;
string webTemplate = "BLOG#0";

ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;

WebCreationInformation webCreateInfo = new WebCreationInformation();
webCreateInfo.Description = blogDescription;
webCreateInfo.Language = blogLanguage;
webCreateInfo.Title = blogTitle;
webCreateInfo.Url = blogUrl;
webCreateInfo.UseSamePermissionsAsParentSite = blogPermissions;
webCreateInfo.WebTemplate = webTemplate;

Web oNewWebsite = oWebsite.Webs.Add(webCreateInfo);

clientContext.Load(
    oNewWebsite,
    website => website.ServerRelativeUrl,
    website => website.Created);

clientContext.ExecuteQuery();
```

```
Console.WriteLine("Server-relative Url: {0} Created: {1}", oNewWebsite.ServerRelativeUrl,
oNewWebsite.Created);
```

o

```
ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;
ListCollection collList = oWebsite.Lists;

clientContext.Load(collList);

clientContext.ExecuteQuery();

foreach (List oList in collList)
{
    Console.WriteLine("Title: {0} Created: {1}", oList.Title, oList.Created.ToString());
}
```

o

```
ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;
ListCollection collList = oWebsite.Lists;

clientContext.Load(
    collList,
    lists => lists.Include(
        list => list.Title,
        list => list.Id));

clientContext.ExecuteQuery();

foreach (List oList in collList)
{
    Console.WriteLine("Title: {0} ID: {1}", oList.Title, oList.Id.ToString("D"));
}
```

o

```
ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;
ListCollection collList = oWebsite.Lists;

IEnumerable<List> resultCollection = clientContext.LoadQuery(
    collList.Include(
        list=>list.Title,
        list=>list.Id));

clientContext.ExecuteQuery();

foreach (List oList in resultCollection)
{
    Console.WriteLine("Title: {0} ID: {1}", oList.Title, oList.Id.ToString("D"));
}
```

```

ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;
ListCollection collList = oWebsite.Lists;

IEnumerable<SP.List> listInfo = clientContext.LoadQuery(
    collList.Include(
        list => list.Title,
        list => list.Fields.Include(
            field => field.Title,
            field => field.InternalName)));

clientContext.ExecuteQuery();

foreach (SP.List oList in listInfo)
{
    FieldCollection collField = oList.Fields;

    foreach (SP.Field oField in collField)
    {
        Regex regEx = new Regex("name", RegexOptions.IgnoreCase);

        if (regEx.IsMatch(oField.InternalName))
        {
            Console.WriteLine("List: {0} \n\t Field Title: {1} \n\t Field Internal Name: {2}",

                oList.Title, oField.Title, oField.InternalName);
        }
    }
}

```

```

ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;

ListCreationInformation listCreationInfo = new ListCreationInformation();
listCreationInfo.Title = "My Announcements List";
listCreationInfo.TemplateType = (int)ListTemplateType.Announcements;

List oList = oWebsite.Lists.Add(listCreationInfo);

clientContext.ExecuteQuery();

```

```

ClientContext clientContext = new ClientContext(siteUrl);

SP.List oList = clientContext.Web.Lists.GetByTitle("Announcements");

SP.Field oField = oList.Fields.AddFieldAsXml("<Field DisplayName='MyField' Type='Number' />",

    true, AddFieldOptions.DefaultValue);

SP.FieldNumber fieldNumber = clientContext.CastTo<FieldNumber>(oField);

```

```

fieldNumber.MaximumValue = 100;
fieldNumber.MinimumValue = 35;

fieldNumber.Update();

clientContext.ExecuteQuery();

```

o

```

ClientContext clientContext = new ClientContext(siteUrl);
Web oWebsite = clientContext.Web;

List oList = oWebsite.Lists.GetByTitle("My Announcements List");

oList.DeleteObject();

clientContext.ExecuteQuery();

```

o

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("Announcements");

CamlQuery camlQuery = new CamlQuery();
camlQuery.ViewXml = "<View><Query><Where><Geq><FieldRef Name='ID' />" +
    "<Value Type='Number'>10</Value></Geq></Where></Query><RowLimit>100</RowLimit></View>";
ListItemCollection collListItem = oList.GetItems(camlQuery);

clientContext.Load(collListItem);

clientContext.ExecuteQuery();

foreach (ListItem oListItem in collListItem)
{
    Console.WriteLine("ID: {0} \nTitle: {1} \nBody: {2}", oListItem.Id, oListItem["Title"],
oListItem["Body"]);
}

```

## o Include

o o o

```

ClientContext clientContext = new ClientContext(siteUrl);
List oList = clientContext.Web.Lists.GetByTitle("Announcements");

CamlQuery camlQuery = new CamlQuery();
camlQuery.ViewXml = "<View><RowLimit>100</RowLimit></View>";

ListItemCollection collListItem = oList.GetItems(camlQuery);

// The first line of this request indicates the list item collection to load from the server
// The second line uses a lambda to request that from the server
// also include additional properties in the response
// The third though fifth lines are the properties being requested from the server
clientContext.Load(collListItem,

```

```

        items => items.Include(
            item => item.Id,
            item => item.DisplayName,
            item => item.HasUniqueRoleAssignments));

clientContext.ExecuteQuery();

foreach (ListItem oListItem in collListItem)
{
    Console.WriteLine("ID: {0} \nDisplay name: {1} \nUnique role assignments: {2}",
        oListItem.Id, oListItem.DisplayName, oListItem.HasUniqueRoleAssignments);
}

```

o

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("Announcements");

CamlQuery camlQuery = new CamlQuery();
ListItemCollection collListItem = oList.GetItems(camlQuery);

clientContext.Load(
    collListItem,
    items => items.Take(5).Include(
        item => item["Title"],
        item => item["Body"]));

clientContext.ExecuteQuery();

foreach (ListItem oListItem in collListItem)
{
    Console.WriteLine("Title: {0} \nBody: {1}\n", oListItem["Title"], oListItem["Body"]);
}

```

o

```

ClientContext clientContext = new ClientContext(siteUrl);
ListCollection collList = clientContext.Web.Lists;

clientContext.Load(
    collList,
    lists => lists.Where(
        list => list.Hidden == false).Include(
            list => list.Title,
            list => list.Items.Take(10)));

clientContext.ExecuteQuery();

foreach (SP.List oList in clientContext.Web.Lists)
{
    string listTitle = oList.Title;
    int itemCount = oList.Items.Count;

    Console.WriteLine("List {0} returned with {1} items", listTitle, itemCount);
}

```

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("Announcements");

ListItemCollectionPosition itemPosition = null;

while (true)
{
    CamlQuery camlQuery = new CamlQuery();

    camlQuery.ListItemCollectionPosition = itemPosition;

    camlQuery.ViewXml = "<View><ViewFields><FieldRef Name='ID' />" +
        "<FieldRef Name='Title' /><FieldRef Name='Body' />" +
        "</ViewFields><RowLimit>5</RowLimit></View>";

    ListItemCollection collListItem = oList.GetItems(camlQuery);

    clientContext.Load(collListItem);

    clientContext.ExecuteQuery();

    itemPosition = collListItem.ListItemCollectionPosition;

    foreach (ListItem oListItem in collListItem)
    {
        Console.WriteLine("Title: {0}: \nBody: {1}", oListItem["Title"], oListItem["Body"]);
    }

    if (itemPosition == null)
    {
        break;
    }

    Console.WriteLine("\n" + itemPosition.PagingInfo + "\n");
}

```

```

ClientContext clientContext = new ClientContext(siteUrl);
List oList = clientContext.Web.Lists.GetByTitle("Announcements");

ListItemCreationInformation itemCreateInfo = new ListItemCreationInformation();
ListItem oListItem = oList.AddItem(itemCreateInfo);
oListItem["Title"] = "My New Item!";
oListItem["Body"] = "Hello World!";

oListItem.Update();

clientContext.ExecuteQuery();

```

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("Announcements");
ListItem oListItem = oList.Items.GetById(3);

oListItem["Title"] = "My Updated Title.";

oListItem.Update();

clientContext.ExecuteQuery();

```

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("Announcements");
ListItem oListItem = oList.GetItemById(2);

oListItem.DeleteObject();

clientContext.ExecuteQuery();

```

## SharePoint

```

ClientContext clientContext = new ClientContext("http://MyServer/sites/MySiteCollection");
GroupCollection collGroup = clientContext.Web.SiteGroups;
Group oGroup = collGroup.GetById(7);
UserCollection collUser = oGroup.Users;

clientContext.Load(collUser);

clientContext.ExecuteQuery();

foreach (User oUser in collUser)
{
    Console.WriteLine("User: {0} ID: {1} Email: {2} Login Name: {3}",
        oUser.Title, oUser.Id, oUser.Email, oUser.LoginName);
}

```

```

ClientContext clientContext = new ClientContext("http://MyServer/sites/MySiteCollection");
GroupCollection collGroup = clientContext.Web.SiteGroups;
Group oGroup = collGroup.GetById(7);
UserCollection collUser = oGroup.Users;

clientContext.Load(collUser,
    users => users.Include(
        user => user.Title,
        user => user.LoginName,
        user => user.Email));

clientContext.ExecuteQuery();

```

```

foreach (User oUser in collUser)
{
    Console.WriteLine("User: {0} Login name: {1} Email: {2}",
        oUser.Title, oUser.LoginName, oUser.Email);
}

```

o

```

ClientContext clientContext = new ClientContext("http://MyServer/sites/MySiteCollection");
GroupCollection collGroup = clientContext.Web.SiteGroups;

clientContext.Load(collGroup);

clientContext.Load(collGroup,
    groups => groups.Include(
        group => group.Users));

clientContext.ExecuteQuery();

foreach (Group oGroup in collGroup)
{
    UserCollection collUser = oGroup.Users;

    foreach (User oUser in collUser)
    {
        Console.WriteLine("Group ID: {0} Group Title: {1} User: {2} Login Name: {3}",
            oGroup.Id, oGroup.Title, oUser.Title, oUser.LoginName);
    }
}

```

## o **SharePoint**

```

ClientContext clientContext = new ClientContext("http://MyServer/sites/MySiteCollection ");
GroupCollection collGroup = clientContext.Web.SiteGroups;
Group oGroup = collGroup.GetById(6);

UserCreationInformation userCreationInfo = new UserCreationInformation();
userCreationInfo.Email = "alias@somewhere.com";
userCreationInfo.LoginName = @"DOMAIN\alias";
userCreationInfo.Title = "John";

User oUser = oGroup.Users.Add(userCreationInfo);

clientContext.ExecuteQuery();

```

o

```

ClientContext oClientContext = new ClientContext("http://MyServer/sites/MySiteCollection");

Web oWebsite = clientContext.Web;

BasePermissions permissions = new BasePermissions();
permissions.Set(PermissionKind.CreateAlerts);
permissions.Set(PermissionKind.ManageAlerts);

```



```

RoleDefinitionCreationInformation roleCreationInfo = new RoleDefinitionCreationInformation();

roleCreationInfo.BasePermissions = permissions;
roleCreationInfo.Description = "A new role with create and manage alerts permission";
roleCreationInfo.Name = "Create and Manage Alerts";
roleCreationInfo.Order = 4;

RoleDefinition oRoleDefinition = oWebsite.RoleDefinitions.Add(roleCreationInfo);

clientContext.ExecuteQuery();

Console.WriteLine("{0} role created.", oRoleDefinition.Name);

```

o

```

ClientContext oClientContext = new
ClientContext("http://MyServer/sites/MySiteCollection/MyWebSite");
Web oWebsite = clientContext.Web;

Principal oUser = oWebsite.SiteUsers.GetByLoginName(@"DOMAIN\alias");

RoleDefinition oRoleDefinition = oWebsite.RoleDefinitions.GetByName("Create and Manage
Alerts");
RoleDefinitionBindingCollection collRoleDefinitionBinding = new
RoleDefinitionBindingCollection(clientContext);
collRoleDefinitionBinding.Add(oRoleDefinition);

RoleAssignment oRoleAssignment = oWebsite.RoleAssignments.Add(oUser,
collRoleDefinitionBinding);

clientContext.Load(oUser,
    user => user.Title);

clientContext.Load(oRoleDefinition,
    role => role.Name);

clientContext.ExecuteQuery();

Console.WriteLine("{0} added with {1} role.", oUser.Title, oRoleDefinition.Name);

```

## o **SharePoint**

```

ClientContext oClientContext = new
ClientContext("http://MyServer/sites/MySiteCollection/MyWebSite");
Web oWebsite = clientContext.Web;

GroupCreationInformation groupCreationInfo = new GroupCreationInformation();
groupCreationInfo.Title = "My New Group";
groupCreationInfo.Description = "Description of new group.";
Group oGroup = oWebsite.SiteGroups.Add(groupCreationInfo);

RoleDefinitionBindingCollection collRoleDefinitionBinding = new
RoleDefinitionBindingCollection(clientContext);

RoleDefinition oRoleDefinition = oWebsite.RoleDefinitions.GetByType(RoleType.Contributor);

collRoleDefinitionBinding.Add(oRoleDefinition);

```

```

oWebsite.RoleAssignments.Add(oGroup, collRoleDefinitionBinding);

clientContext.Load(oGroup,
    group => group.Title);

clientContext.Load(oRoleDefinition,
    role => role.Name);

clientContext.ExecuteQuery();

Console.WriteLine("{0} created and assigned {1} role.", oGroup.Title, oRoleDefinition.Name);
}

```

o

```

string siteUrl = "http://MyServer/sites/MySiteCollection";
ClientContext oContext = new ClientContext(siteUrl);
SP.List oList = oContext.Web.Lists.GetByTitle("Announcements");

oList.BreakRoleInheritance(true, false);

oContext.ExecuteQuery();

```

o

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("MyList");

int itemId = 3;
ListItem oListItem = oList.Items.GetById(itemId);

oListItem.BreakRoleInheritance(false);

User oUser = clientContext.Web.SiteUsers.GetByLoginName(@"DOMAIN\alias");

RoleDefinitionBindingCollection collRoleDefinitionBinding = new
RoleDefinitionBindingCollection(clientContext);

collRoleDefinitionBinding.Add(clientContext.Web.RoleDefinitions.GetByType(RoleType.Reader));

oListItem.RoleAssignments.Add(oUser, collRoleDefinitionBinding);

clientContext.ExecuteQuery();

```

o

```

ClientContext clientContext = new ClientContext(siteUrl);
SP.List oList = clientContext.Web.Lists.GetByTitle("MyList");

int itemId = 2;
ListItem oListItem = oList.Items.GetById(itemId);

oListItem.BreakRoleInheritance(true);

```

```

User oUser = clientContext.Web.SiteUsers.GetByLoginName(@"DOMAIN\alias");
oListItem.RoleAssignments.GetByPrincipal(oUser).DeleteObject();

RoleDefinitionBindingCollection collRollDefinitionBinding = new
RoleDefinitionBindingCollection(clientContext);

collRollDefinitionBinding.Add(clientContext.Web.RoleDefinitions.GetByType(RoleType.Reader));

oListItem.RoleAssignments.Add(oUser, collRollDefinitionBinding);

clientContext.ExecuteQuery();

```

o

```

string urlWebsite = "http://MyServer/sites/MySiteCollection";
ClientContext clientContext = new ClientContext(urlWebsite);
Web oWebsite = clientContext.Web;

List oList = oWebsite.Lists.GetByTitle("My List");
UserCustomActionCollection collUserCustomAction = oList.UserCustomActions;

UserCustomAction oUserCustomAction = collUserCustomAction.Add();
oUserCustomAction.Location = "EditControlBlock";
oUserCustomAction.Sequence = 100;
oUserCustomAction.Title = "My First User Custom Action";
oUserCustomAction.Url = urlWebsite + @"/_layouts/MyPage.aspx";
oUserCustomAction.Update();

clientContext.Load(oList,
    list => list.UserCustomActions);

clientContext.ExecuteQuery();

```

o

```

string urlWebsite = "http://MyServer/sites/SiteCollection";
ClientContext clientContext = new ClientContext(urlWebsite);
Web oWebsite = clientContext.Web;

List oList = oWebsite.Lists.GetByTitle("My List");
UserCustomActionCollection collUserCustomAction = oList.UserCustomActions;

clientContext.Load(collUserCustomAction,
    userCustomActions => userCustomActions.Include(
        userCustomAction => userCustomAction.Title));

clientContext.ExecuteQuery();

foreach (UserCustomAction oUserCustomAction in collUserCustomAction)
{
    if (oUserCustomAction.Title == "My First User Custom Action")
    {
        oUserCustomAction.ImageUrl = "http://MyServer/_layouts/images/MyIcon.png";
        oUserCustomAction.Update();

        clientContext.ExecuteQuery();
    }
}

```

```
}
```

o

```
string urlWebsite = "http://MyServer/sites/MySiteCollection";
ClientContext clientContext = new ClientContext(urlWebsite);

Web oWebsite = clientContext.Web;
UserCustomActionCollection collUserCustomAction = oWebsite.UserCustomActions;

UserCustomAction oUserCustomAction = collUserCustomAction.Add();

oUserCustomAction.Location = "Microsoft.SharePoint.StandardMenu";
oUserCustomAction.Group = "SiteActions";
oUserCustomAction.Sequence = 101;
oUserCustomAction.Title = "Website User Custom Action";
oUserCustomAction.Description = "This description appears on the Site Actions menu.";
oUserCustomAction.Url = urlWebsite + @"/_layouts/MyPage.aspx";

oUserCustomAction.Update();

clientContext.Load(oWebsite,
    webSite => webSite.UserCustomActions);

clientContext.ExecuteQuery();
```

o **Web**

```
ClientContext oClientContext = new ClientContext("http://MyServer/sites/MySiteCollection");
File oFile = oClientContext.Web.GetFileByServerRelativeUrl("Default.aspx");
LimitedWebPartManager limitedWebPartManager =
oFile.GetLimitedWebPartManager(PersonalizationScope.Shared);

oClientContext.Load(limitedWebPartManager.WebParts,
    wps => wps.Include(
    wp => wp.WebPart.Title));

oClientContext.ExecuteQuery();

if (limitedWebPartManager.WebParts.Count == 0)
{
    throw new Exception("No Web Parts on this page.");
}

WebPartDefinition oWebPartDefinition = limitedWebPartManager.WebParts[1];
WebPart oWebPart = oWebPartDefinition.WebPart;
oWebPart.Title = "My New Web Part Title";

oWebPartDefinition.SaveWebPartChanges();

oClientContext.ExecuteQuery();
```

o **Web**

```
ClientContext oClientContext = new ClientContext("http://MyServer/sites/MySiteCollection");
```

```

File oFile = oClientContext.Web.GetFileByServerRelativeUrl("Default.aspx");
LimitedWebPartManager limitedWebPartManager =
oFile.GetLimitedWebPartManager(PersonalizationScope.Shared);

string xmlWebPart = "<?xml version=\"1.0\" encoding=\"utf-8\"?>" +
    "<WebPart xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" " +
    " xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" " +
    " xmlns=\"http://schemas.microsoft.com/WebPart/v2\">" +
    "<Title>My Web Part</Title><FrameType>Default</FrameType>" +
    "<Description>Use for formatted text, tables, and images.</Description>" +
    "<IsIncluded>true</IsIncluded><ZoneID></ZoneID><PartOrder>0</PartOrder>" +
    "<FrameState>Normal</FrameState><Height /><Width /><AllowRemove>true</AllowRemove>" +
    "<AllowZoneChange>true</AllowZoneChange><AllowMinimize>true</AllowMinimize>" +
    "<AllowConnect>true</AllowConnect><AllowEdit>true</AllowEdit>" +
    "<AllowHide>true</AllowHide><IsVisible>true</IsVisible><DetailLink /><HelpLink />" +
    "<HelpMode>Modeless</HelpMode><Dir>Default</Dir><PartImageSmall />" +
    "<MissingAssembly>Cannot import this Web Part.</MissingAssembly>" +
    "<PartImageLarge>/_layouts/images/mscontl.gif</PartImageLarge><IsIncludedFilter />" +
    "<Assembly>Microsoft.SharePoint, Version=13.0.0.0, Culture=neutral, " +
    "PublicKeyToken=94de0004b6e3fcc5</Assembly>" +
    "<TypeName>Microsoft.SharePoint.WebPartPages.ContentEditorWebPart</TypeName>" +
    "<ContentLink xmlns=\"http://schemas.microsoft.com/WebPart/v2/ContentEditor\" />" +
    "<Content xmlns=\"http://schemas.microsoft.com/WebPart/v2/ContentEditor\">" +
    "<![CDATA[This is a first paragraph!<DIV>&nbsp;</DIV>And this is a second
paragraph.]]></Content>" +
    "<PartStorage xmlns=\"http://schemas.microsoft.com/WebPart/v2/ContentEditor\"
/></WebPart>";

WebPartDefinition oWebPartDefinition = limitedWebPartManager.ImportWebPart(xmlWebPart);

limitedWebPartManager.AddWebPart(oWebPartDefinition.WebPart, "Left", 1);

oClientContext.ExecuteQuery();

```

## ◦ Web

```

ClientContext oClientContext = new ClientContext("http://MyServer/sites/MySiteCollection");
File oFile =
oClientContext.Web.GetFileByServerRelativeUrl("/sites/MySiteCollection/SitePages/Home.aspx ");
LimitedWebPartManager limitedWebPartManager =
oFile.GetLimitedWebPartManager(PersonalizationScope.Shared);

oClientContext.Load(limitedWebPartManager.WebParts);

oClientContext.ExecuteQuery();

if (limitedWebPartManager.WebParts.Count == 0)
{
    throw new Exception("No Web Parts to delete.");
}

WebPartDefinition webPartDefinition = limitedWebPartManager.WebParts[0];

webPartDefinition.DeleteWebPart();

oClientContext.ExecuteQuery();

```

◦

◦ ◦

[CredentialCache](#)ClientContextCredentials◦

## NTLMSharePoint 2013◦

```
using System.Net;
using Microsoft.SharePoint.Client;

using (ClientContext ctx = new ClientContext("https://onpremises.local/sites/demo/"))
{
    // need the web object
    ctx.Load(ctx.Web);
    ctx.ExecuteQuery();

    // here the default network credentials relate to the identity of the account
    // running the App Pool of your web application.
    CredentialCache credCache = new CredentialCache();
    cc.Add(new Uri(ctx.Web.Url), "NTLM", CredentialCache.DefaultNetworkCredentials);

    ctx.Credentials = credCache;
    ctx.AuthenticationMode = ClientAuthentication.Default;
    ctx.ExecuteQuery();

    // do stuff as elevated app pool account
}
```

## SharePoint◦

[CSOM https://riptutorial.com/zh-CN/sharepoint/topic/2679/-csom-](https://riptutorial.com/zh-CN/sharepoint/topic/2679/-csom-)

---

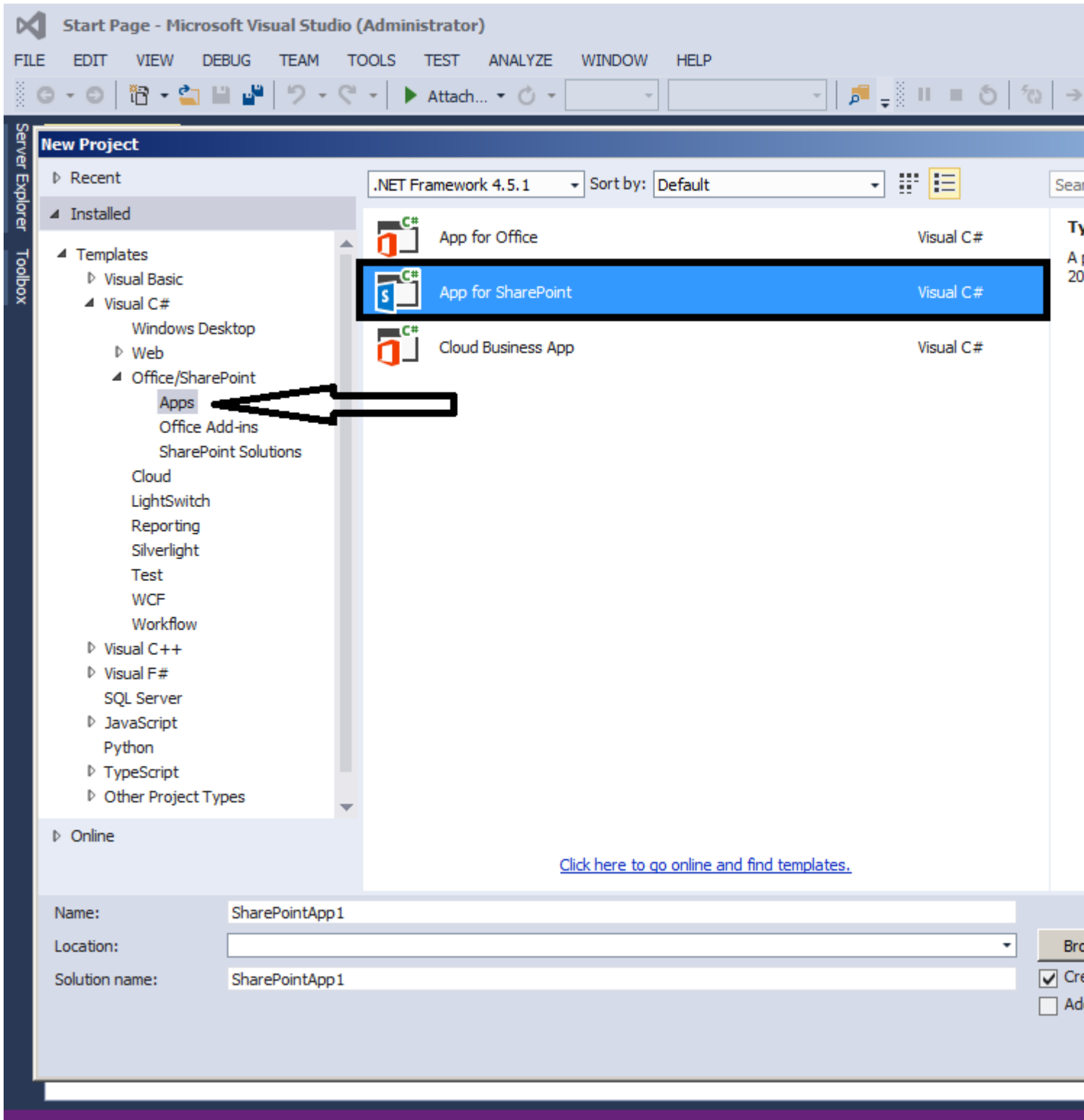
# 10: App

## Examples

App Development Visual Studio 2013. > <https://www.visualstudio.com/products/free-developer-offers-vs>

“”

Office / SharePointApp.



AppVSMicrosoft Office Developer Tools <https://www.visualstudio.com/en-us/features/office-tools-vs.aspx>

visual studioSharePoint. >Office 365

<https://profile.microsoft.com/RegSysProfileCenter/wizardnp.aspx?wizid=14b845d0-938c-45af-b061-f798fbb4d170&lcid=1033>

[https //www.office.com/URL](https://www.office.com/URL)



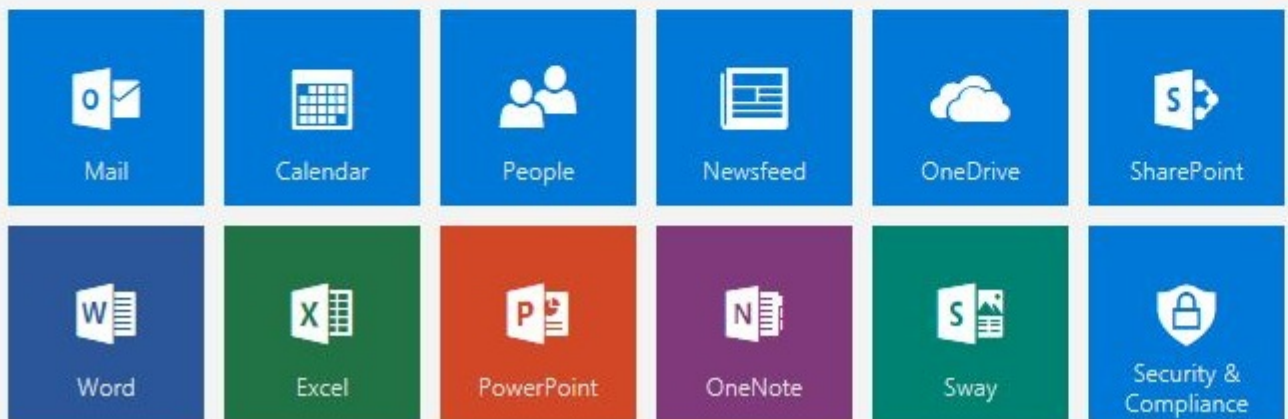


Good afternoon,

Search online documents

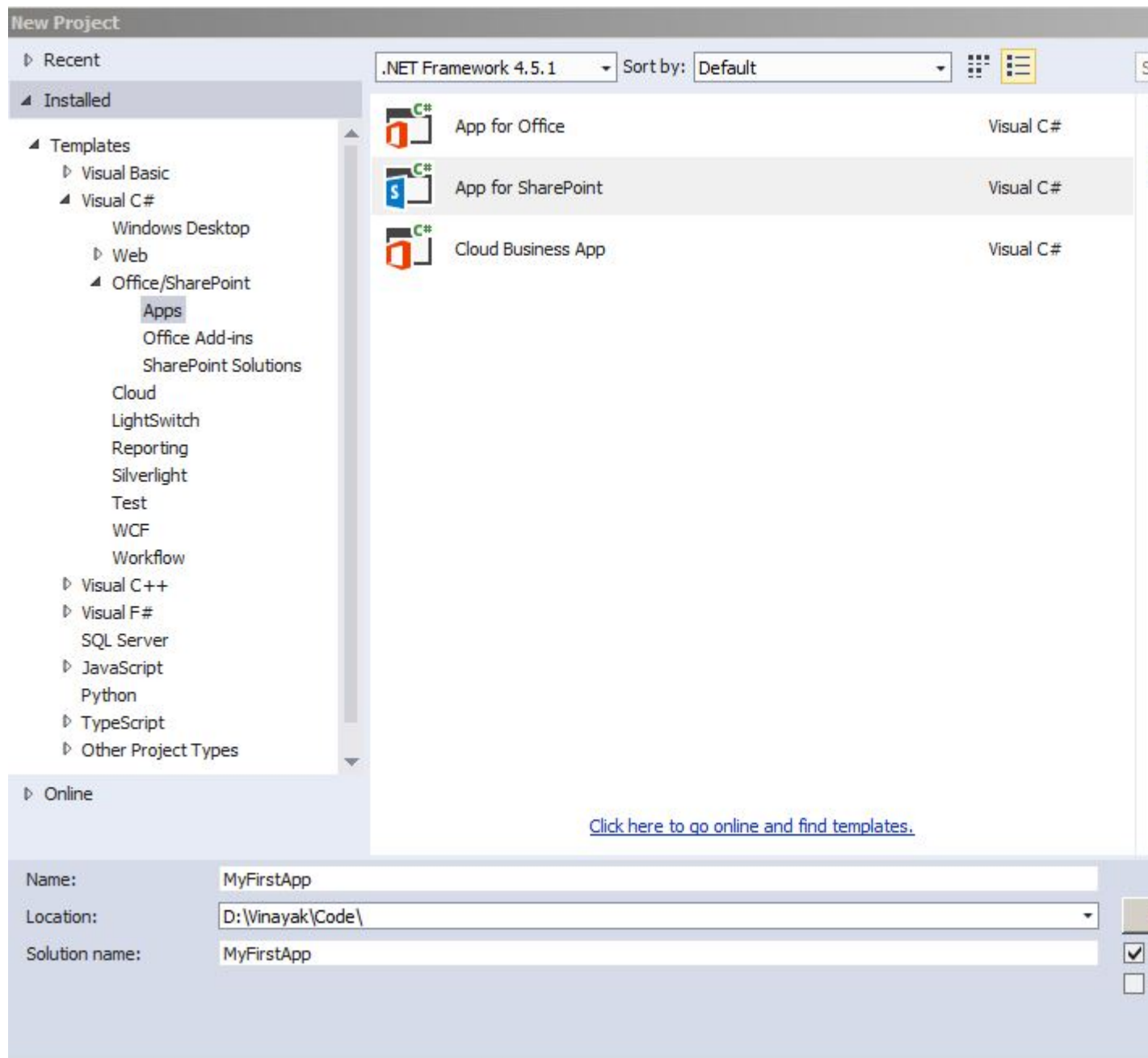


### Use the online apps




## Visual Studio

1. visual studio>
- 2.



3.

New app for SharePoint ? ×

 Specify the app for SharePoint settings

**What SharePoint site do you want to use for debugging your app?**

**Don't have a developer site?**  
[Sign up for an Office 365 Developer site to develop, test and deploy apps for Office and SharePoint](#)

**How do you want to host your app for SharePoint?**

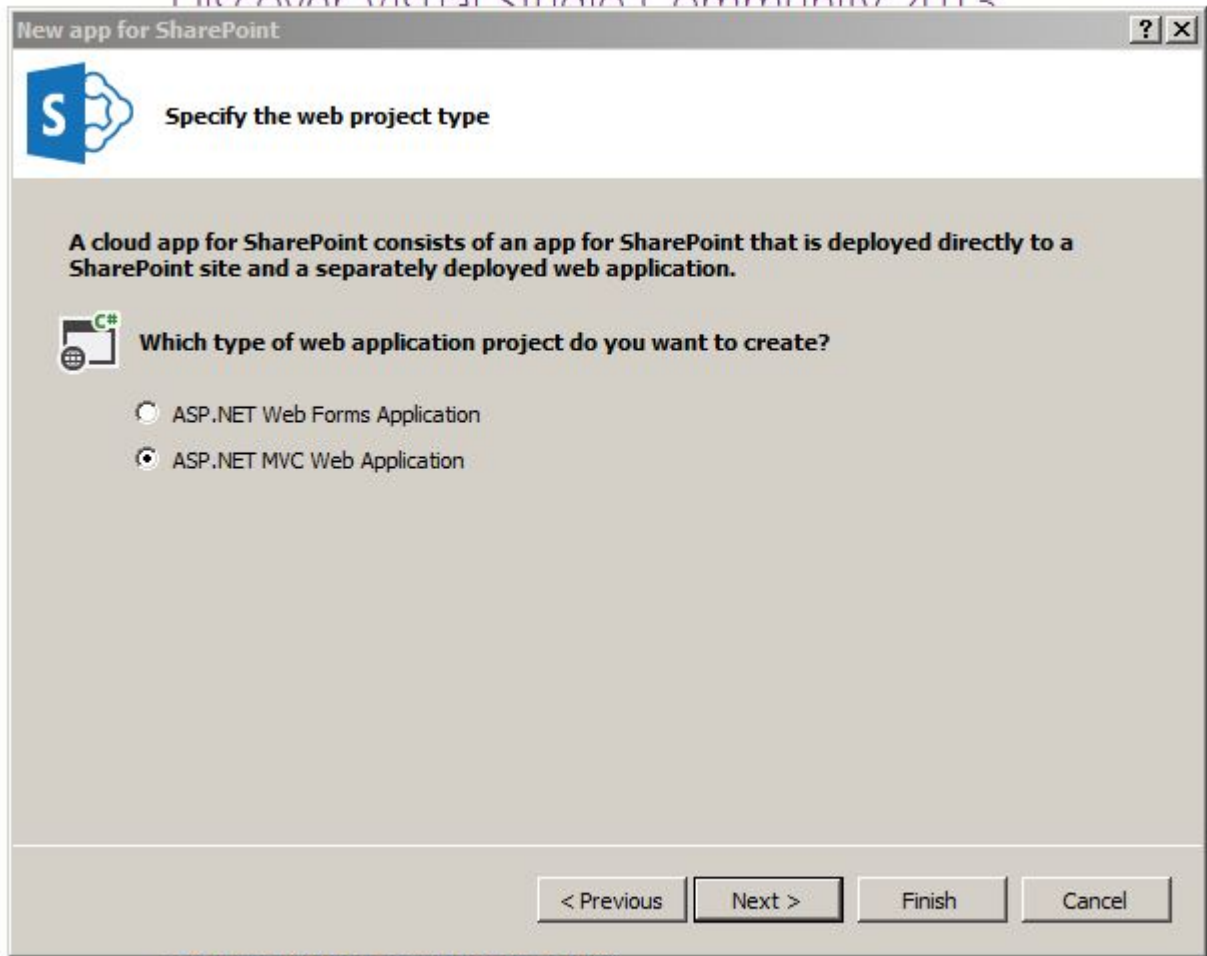
Provider-hosted  
 SharePoint-hosted  
[Learn more about this choice](#)

4.

5. MVCWebform。 MCV

3

## Discover Visual Studio Community 2013



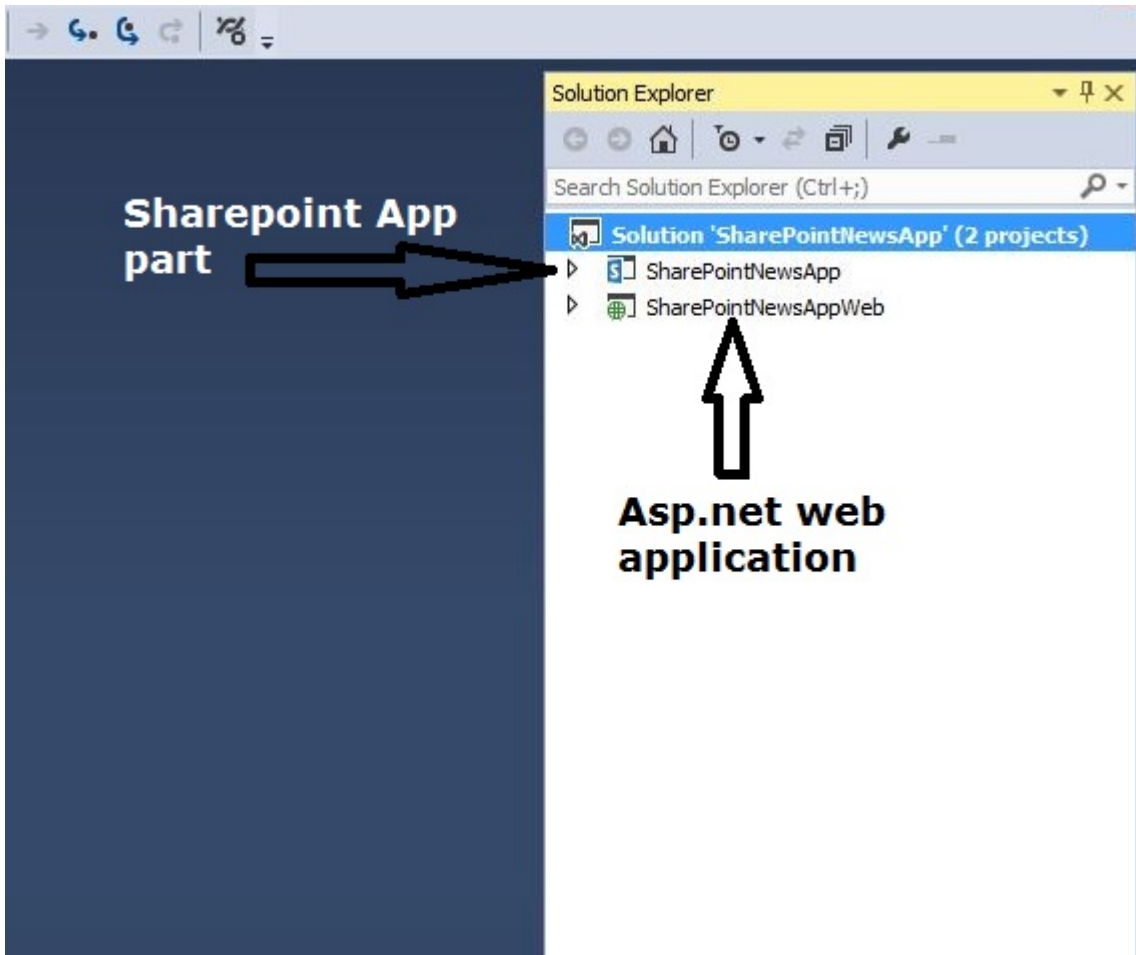
[Exploring dotnet new with .NET Core](#)

Wednesday, July 27, 2016

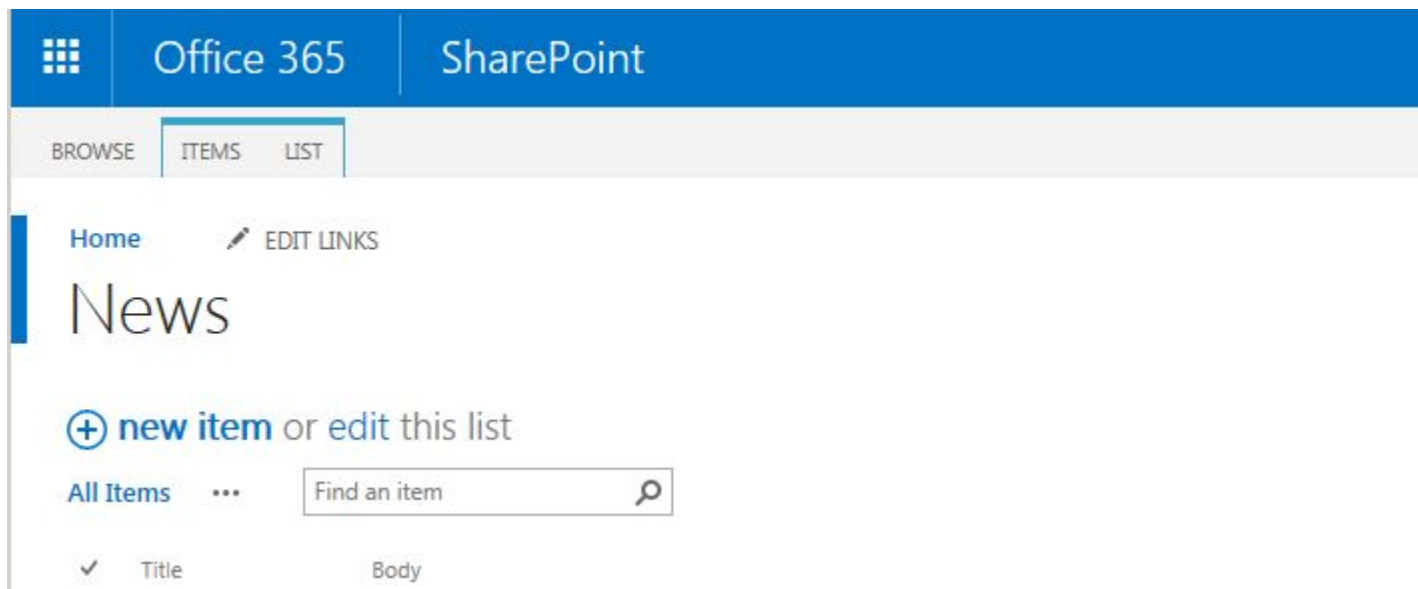
I'm very enjoying the "dotnet" command line. Mostly I do "dotnet new" and then add to the default Hello World app with the Visual Studio Code editor. Recently, though, I realized that the -t "type" and -l "lang" options are there

6. ""Windows Azure""

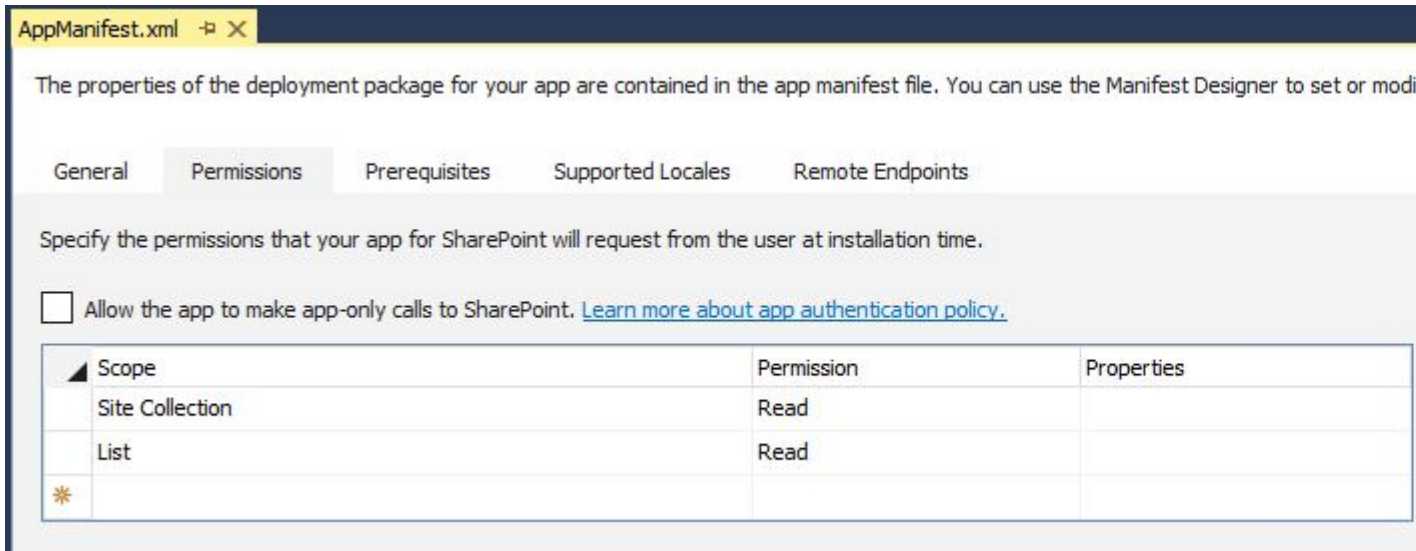
7. 2. SharePointasp.net Web



1. SharePoint
2. 3BodySummeryThumbnaillImageUrl



3. SharePointAppManifest.xml.



#### 4. WebHomeControllerMVC。webformdefault.aspx.cs

5. . . .

```
[SharePointContextFilter]
public ActionResult Index()
{
    User spUser = null;

    var spContext = SharePointContextProvider.Current.GetSharePointContext(HttpContext);
    List<NewsList> newsList = new List<NewsList>();
    using (var clientContext = spContext.CreateUserClientContextForSPHost())
    {
        if (clientContext != null)
        {
            spUser = clientContext.Web.CurrentUser;

            clientContext.Load(spUser, user => user.Title);

            clientContext.ExecuteQuery();

            ViewBag.UserName = spUser.Title;

            List lst = clientContext.Web.Lists.GetByTitle("News");
            CamlQuery queryNews = CamlQuery.CreateAllItemsQuery(10);
            ListItemCollection newsItems = lst.GetItems(queryNews);
            clientContext.Load(newsItems, includes => includes.Include(i => i.Id, i =>
i.DisplayName, i => i["ThumbnailImageUrl"], i => i["Summery"]));

            clientContext.ExecuteQuery();

            if (newsItems != null)
            {
                foreach (var lstProductItem in newsItems)
                {
                    newsList.Add(
                        new NewsList
                        {
                            Id = Convert.ToInt32(lstProductItem.Id.ToString()),
                            Title = lstProductItem.DisplayName.ToString(),
                            Summery = lstProductItem["Summery"].ToString(),
                            Thumbnail = lstProductItem["ThumbnailImageUrl"].ToString()
                        }
                    );
                }
            }
        }
    }
}
```

```

        });
    }
}
}

return View(newsList);
}

```

## 6. IndexAdd View。 “”

## 7. Views> HomeIndex.cshtml

## 8. index.cshtml

```

@model List<SharePointNewsAppWeb.Models.NewsList>
@{
    ViewBag.Title = "My News - browse latest news";
}
<br />
@foreach (var item in Model)
{
    <div class="row panel panel-default">
        <div class="col-xs-3">
            <a href="/home/aticle?ArticleId=@item.Id">
                
            </a>
        </div>
        <div class="col-xs-9 panel-default">
            <div class="panel-heading">
                <h4><a href="/home/aticle?ArticleId=@item.Id">@item.Title.ToUpper()</a></h4>
            </div>
            <div class="panel-body">
                <p>@item.Summary</p>
            </div>
        </div>
    </div>
}

```

## 9.

## ModelCS。

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace SharePointNewsAppWeb.Models
{
    public class NewsApp
    {
    }
    public class NewsList
    {

public int Id { get; set; }

public string Title { get; set; }

public string Summery { get; set; }

public string Thumbnail { get; set; }
    }
    public class FullArticle
    {

public int Id { get; set; }

public string Title { get; set; }

public string Body { get; set; }

    }
}
}
```

### 10. F5。 “Localhost”。

。 。

#### 1. HomeController

```
[SharePointContextFilter]
public ActionResult Aticle(int ArticleId)
{
    User spUser = null;

    var spContext = SharePointContextProvider.Current.GetSharePointContext(HttpContext);
    FullArticle article = new FullArticle();
    using (var clientContext = spContext.CreateUserClientContextForSPHost())
    {
        if (clientContext != null)
        {
            spUser = clientContext.Web.CurrentUser;

            clientContext.Load(spUser, user => user.Title);

            clientContext.ExecuteQuery();
        }
    }
}
```



```

        ViewBag.UserName = spUser.Title;

        List lst = clientContext.Web.Lists.GetByTitle("News");
        CamlQuery queryNews = new CamlQuery();
        queryNews.ViewXml = @"<View><Query><Where><Eq><FieldRef Name='ID' />" +
"<Value Type='Number'>" + ArticleId + "</Value></Eq></Where></Query>" +
        "<ViewFields><FieldRef Name='ID' /><FieldRef Name='Title' /><FieldRef
Name='Body' /></ViewFields></View>";//
        ListItemCollection newsItems = lst.GetItems(queryNews);
        clientContext.Load(newsItems, includes => includes.Include(i => i.Id, i =>
i.DisplayName, i => i["Body"]));

        clientContext.ExecuteQuery();

        if (newsItems != null)
        {
            foreach (var lstProductItem in newsItems)
            {
                article.Id = Convert.ToInt32(lstProductItem.Id.ToString());
                article.Title = lstProductItem.DisplayName.ToString();
                article.Body = lstProductItem["Body"].ToString();
            }
        }
    }
    return View(article);
}

```

## 2. ActionView。 ViewArticle

```

@model SharePointNewsAppWeb.Models.FullArticle

@{
    ViewBag.Title = "Article";
}

<br />
<div class="panel panel-default">
    <div class="panel-heading"><a style="font-size:20px;" href="/"><i class="glyphicon
glyphicon-chevron-left"></i> <i class="glyphicon glyphicon-home"></i> </a></div>
    <div class="panel-heading"><h1 class="h2">@Model.Title.ToUpper()</h1></div>
    <div class="panel-body">@Html.Raw(@Model.Body)</div>
</div>

```

App <https://riptutorial.com/zh-CN/sharepoint/topic/6301/app>

S. No		Contributors
1	sharepoint	<a href="#">Community</a> , <a href="#">Marco</a> , <a href="#">Ryan Gregg</a> , <a href="#">Thriggle</a> , <a href="#">Tom Resing</a> , <a href="#">Zach Koehne</a>
2	REST	<a href="#">Aaron</a> , <a href="#">Brock Davis</a> , <a href="#">ocelotsloth</a> , <a href="#">R4mbi</a> , <a href="#">Rohit Waghela</a> , <a href="#">Thriggle</a>
3	SharePoint 2013	<a href="#">Rohit Waghela</a> , <a href="#">Yayati</a>
4	SharePoint	<a href="#">Sunil sahu</a>
5		<a href="#">jjr2527</a> , <a href="#">MikhailSP</a>
6	JavaScriptJSOM	<a href="#">Thriggle</a> , <a href="#">yngrdyn</a>
7	JavaScript	<a href="#">Thriggle</a>
8		<a href="#">Lukáš Nešpor</a> , <a href="#">Thriggle</a>
9	CSOM	<a href="#">InvoiceGuy</a> , <a href="#">Lukáš Nešpor</a> , <a href="#">MikhailSP</a> , <a href="#">RamenChef</a> , <a href="#">Thriggle</a> , <a href="#">Zach Koehne</a>
10	App	<a href="#">vinayak hegde</a>