# How to log UI traces into Windows Azure table storage (VBAzureJSLogging)

## Introduction

## This sample shows how to add Web UI traces into Azure table storage.

Azure now supports CORS. It’s easy to use JavaScript to create a JS logging Framework based on Azure table storage.

This sample will demonstrate how to use JavaScript to add log to Azure table storage.

## Building the Sample

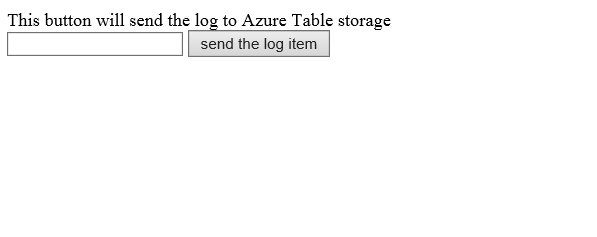
You need rebuild this sample first, then the project will download necessary libraries.

## Running the Sample

You need to open the solution, find AzureClient.cs/vb file, and change the parameters to your Azure storage account and Azure storage account key first.

Then you can run the sample directly.

You will see the images below:



You can add some message into the text box. The text will be uploaded to your Azure storage account logs table.

## Using the Code

This code snippet shows how to enable table storage CORS feature.

|  |
| --- |
| -Code block start-  --C# code snippet start--  /// <summary>  /// Adds CORS rule to the service properties.  /// </summary>  /// <param name="serviceProperties">ServiceProperties</param>  private static void ConfigureCors(ServiceProperties serviceProperties)  {  serviceProperties.Cors = new CorsProperties();  serviceProperties.Cors.CorsRules.Add(new CorsRule()  {  AllowedHeaders = new List<string>() { "\*" },  AllowedMethods = CorsHttpMethods.Put | CorsHttpMethods.Get | CorsHttpMethods.Head | CorsHttpMethods.Post,  AllowedOrigins = new List<string>() { "\*" },  ExposedHeaders = new List<string>() { "\*" },  MaxAgeInSeconds = 1800 // 30 minutes  });  }  --C# code snippet end--  --VB code snippet start--  ''' <summary>  ''' Adds CORS rule to the service properties.  ''' </summary>  ''' <param name="serviceProperties">ServiceProperties</param>  Private Shared Sub ConfigureCors(serviceProperties As ServiceProperties)  serviceProperties.Cors = New CorsProperties()  ' 30 minutes  serviceProperties.Cors.CorsRules.Add(New CorsRule() With { \_  .AllowedHeaders = New List(Of String)() From { \_  "\*" \_  }, \_  .AllowedMethods = CorsHttpMethods.Put Or CorsHttpMethods.[Get] Or CorsHttpMethods.Head Or CorsHttpMethods.Post, \_  .AllowedOrigins = New List(Of String)() From { \_  "\*" \_  }, \_  .ExposedHeaders = New List(Of String)() From { \_  "\*" \_  }, \_  .MaxAgeInSeconds = 1800 \_  })  End Sub  --VB code snippet end--  -Code block end- |

This code snippet shows how to request a SAS URL, then use this URL upload log message to Azure table storage.

|  |
| --- |
| -Code block start-  --JavaScript code snippet start--  <script type="text/javascript">  function addTableEntity(jsonString, tableSasUrl) {  $.ajax({  type: 'POST',  datatype: "json",  data: jsonString,  url: tableSasUrl,  beforeSend: setHeader,  success: function (data, status, xhr) {  alert("upload to Azure table storage success!");  },  error: function (data, status, xhr) {  alert("upload to Azure table storage failed!");  }  });  }  function setHeader(xhr) {  xhr.setRequestHeader('x-ms-version', '2013-08-15');  xhr.setRequestHeader('MaxDataServiceVersion', '3.0');  xhr.setRequestHeader('Accept', 'application/json;odata=nometadata');  xhr.setRequestHeader('Content-Type', 'application/json;odata=nometadata');  }  function addLogtoTableStorage() {  var tableSasApiUrl = '[@Url.RouteUrl("DefaultApi](mailto:@Url.RouteUrl(%22DefaultApi)", New With {.httproute = "", .controller = "tablesas"})';  var entity = {  PartitionKey: 'default',  RowKey: (new Date()).getTime().toString(),  log: $("#txtLogItem").val()  };  alert(tableSasApiUrl);  $.ajax({  type: 'GET',  url: tableSasApiUrl,  dataType:'json',  success: function (data, states, xhr) {  if (xhr.responseText != "") {  alert(data.SASTokenUrl)  addTableEntity(JSON.stringify(entity), data.SASTokenUrl);  }  return xhr.responseText;  },  error: function (data, status, xhr) {  alert("can't get SAS for table");  }  })  }  </script>  --JavaScript code snippet end--  -Code block end- |

## More Information

http://blogs.msdn.com/b/windowsazurestorage/archive/2014/02/03/windows-azure-storage-introducing-cors.aspx