# How to Upload and Download Files Pragmatically to Azure Blob Storage using VB.Net

In this tutorial I'm going to show how to:

- Connect to Azure Storage
- List files/folders (CloudBlockBlob/CloudBlobDirectory) within an Azure File Storage
- Create a Shared Access Signature for a Blob
- Obtain Blob properties
- Read content from a Blob
- Write content to a Blob
- Upload a Blob to Azure File Storage
- Delete a Blob
- Rename/move a Blob
- Download a Blob

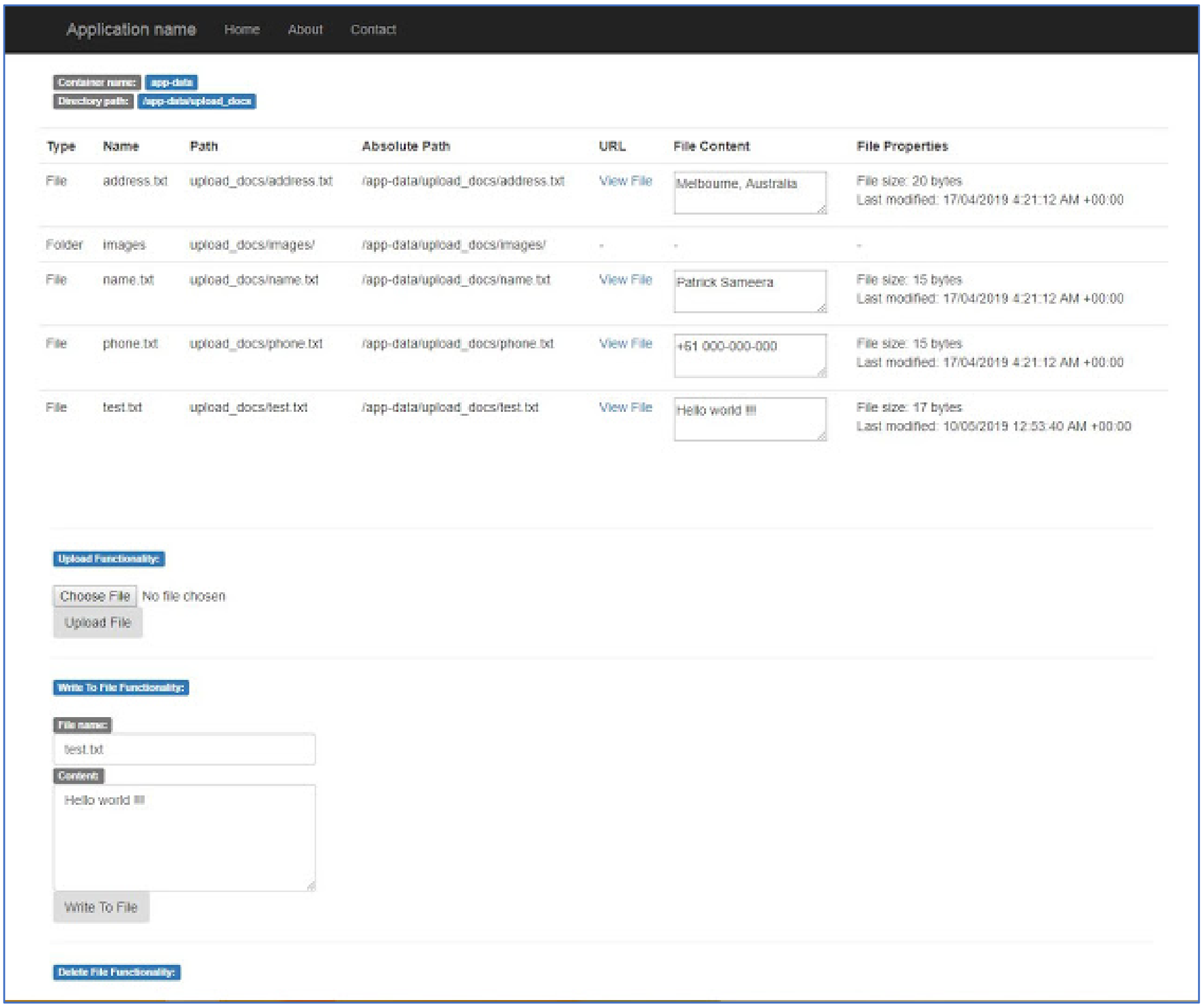
Source code for this Project can be found here:
*https://github.com/patricksameerajayalath/Azure.FileStorage*

You can fid the Part 1 of this tutorial here:

*https://github.com/patricksameerajayalath/AzureTutorials/blob/master/Azure%20File%20Storage%20Overview.pdf*

Patrick Sameera Jayalath

At the end of this tutorial we will be ending up a UI like bellow.



Container name: app-data
Directory path: /app-data/upload_docs

| Type | Name | Path | Absolute Path | URL | File Content | File Properties |
|---|---|---|---|---|---|---|
| File | address.txt | upload_docs/address.txt | /app-data/upload_docs/address.txt | View File | Melbourne, Australia | File size: 20 bytes<br>Last modified: 17/04/2019 4:21:12 AM +00:00 |
| Folder | images | upload_docs/images/ | /app-data/upload_docs/images/ | - | - | - |
| File | name.txt | upload_docs/name.txt | /app-data/upload_docs/name.txt | View File | Patrick Sameera | File size: 15 bytes<br>Last modified: 17/04/2019 4:21:12 AM +00:00 |
| File | phone.txt | upload_docs/phone.txt | /app-data/upload_docs/phone.txt | View File | +61 000-000-000 | File size: 15 bytes<br>Last modified: 17/04/2019 4:21:12 AM +00:00 |
| File | test.txt | upload_docs/test.txt | /app-data/upload_docs/test.txt | View File | Hello world !!! | File size: 17 bytes<br>Last modified: 10/05/2019 12:53:40 AM +00:00 |

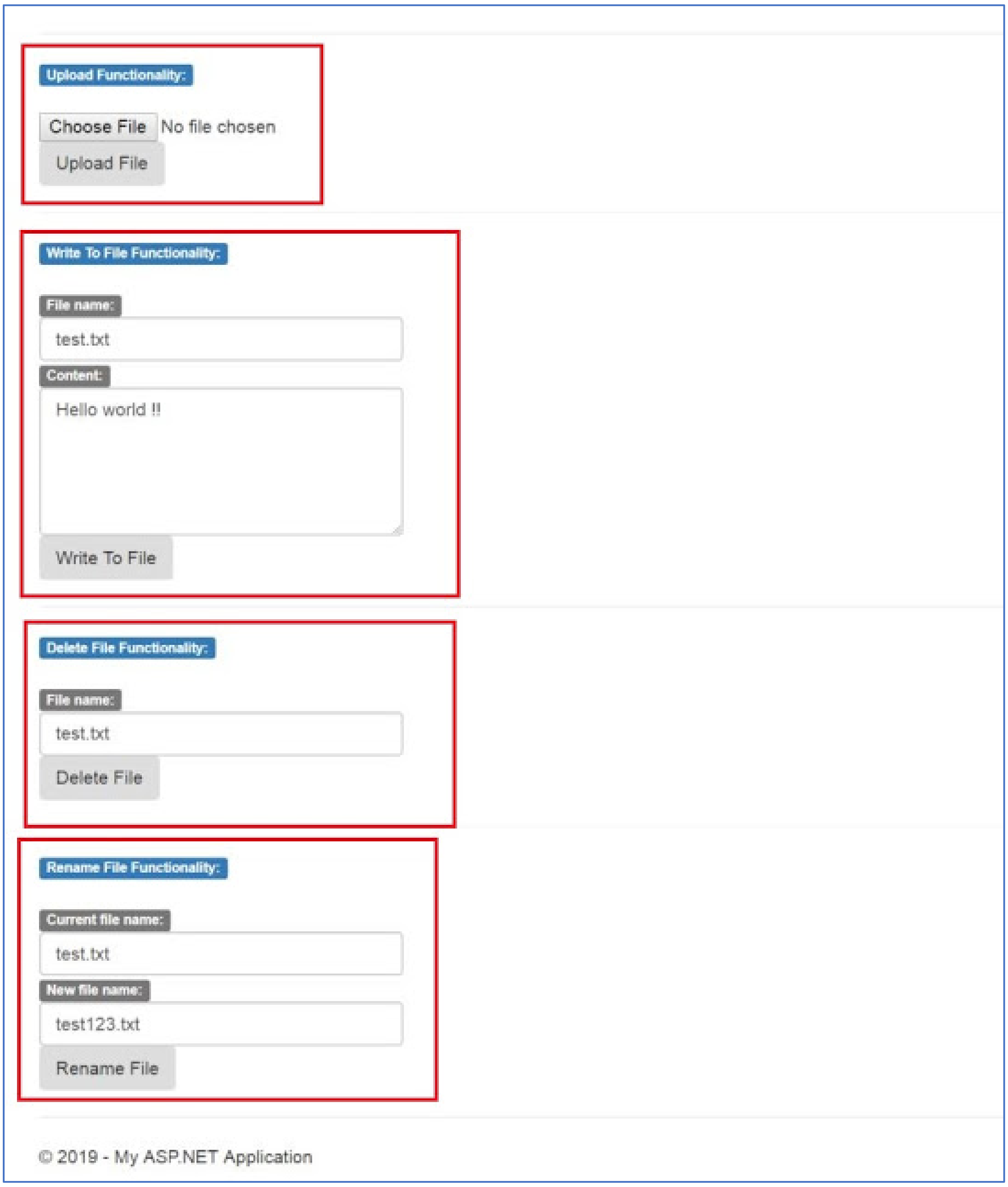
Upload Functionality:

Write To File Functionality:

File name: test.txt

Content: Hello world !!!

Delete File Functionality:



Upload Functionality:
Choose File
Upload File
Write To File Functionality:
Write To File
Delete File Functionality:
Delete File
Rename File Functionality:
Rename File

No file chosen

File name: test.txt

Content: Hello world !!

File name: test.txt

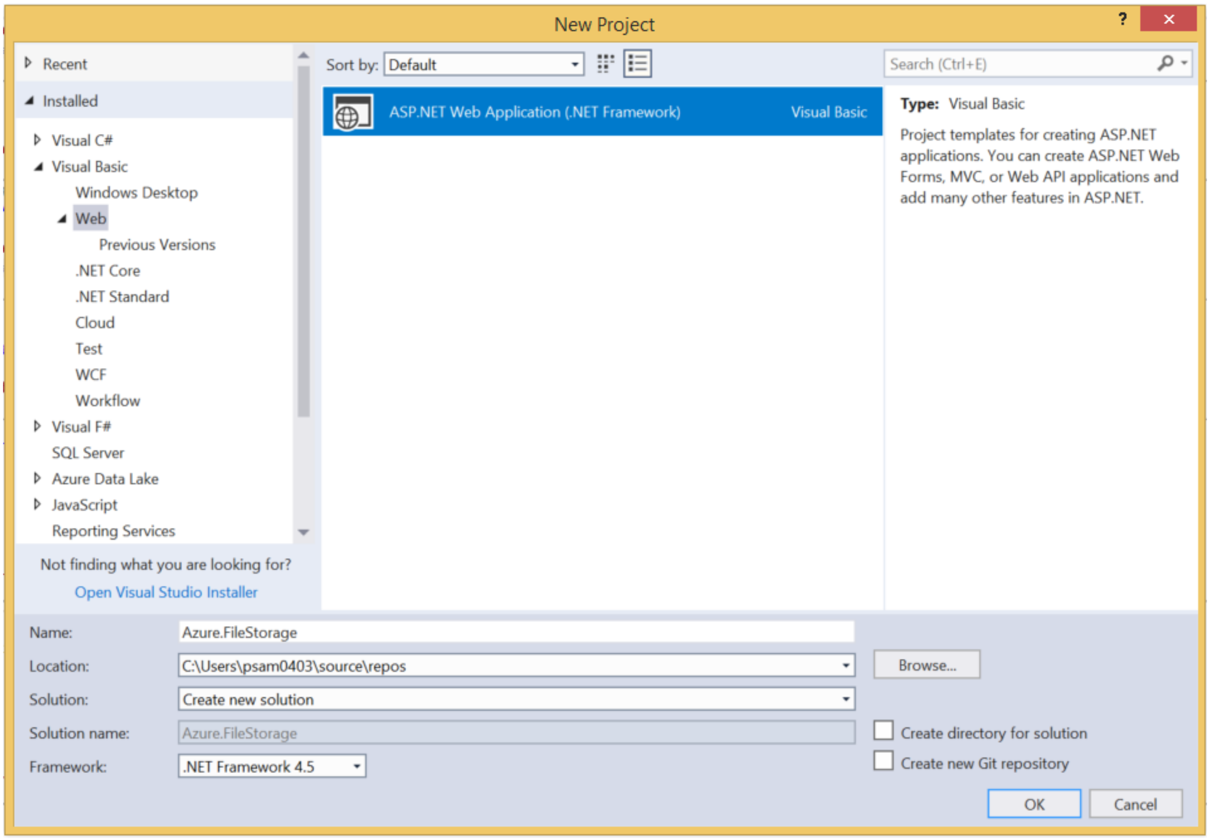
Current file name: test.txt

New file name: test123.txt

© 2019 - My ASP.NET Application

To get start we will be creating an ASP.Net Web Project.

- Language – VB.Net



ASP.NET Web Application (.NET Framework) — Visual Basic

**Type:** Visual Basic

Project templates for creating ASP.NET applications. You can create ASP.NET Web Forms, MVC, or Web API applications and add many other features in ASP.NET.

Name: Azure.FileStorage

Location: C:\Users\psam0403\source\repos

Solution: Create new solution

Solution name: Azure.FileStorage

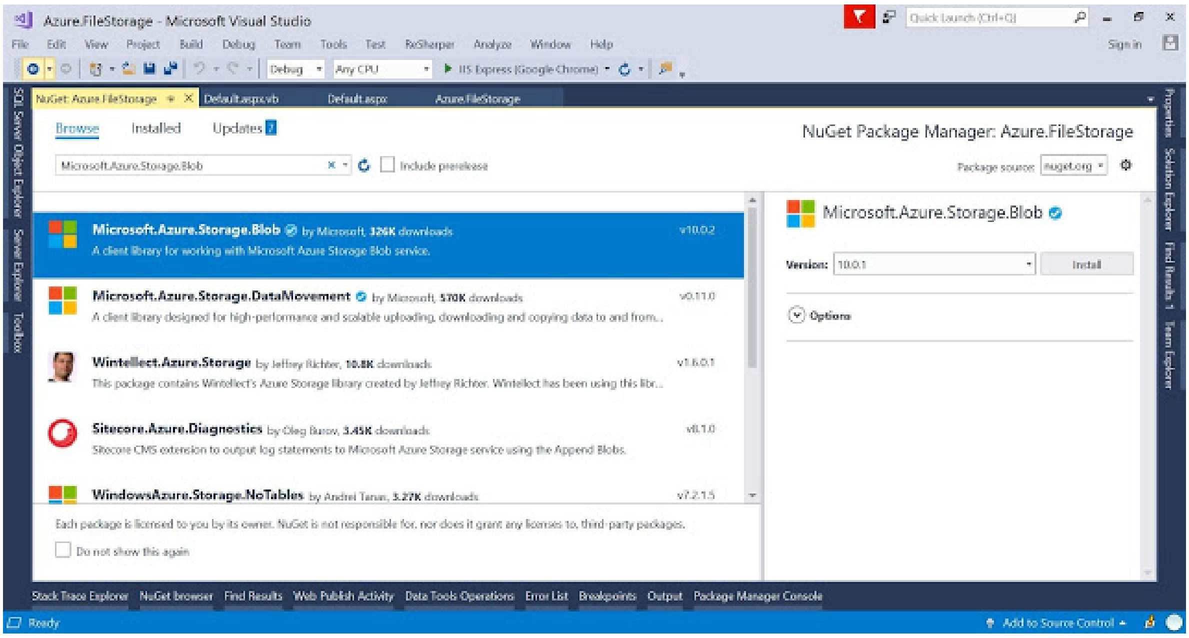
Framework: .NET Framework 4.5

□ Create directory for solution

□ Create new Git repository

In the Solution Explorer, right click on the Project and select Manage NuGet Packages.

Under Browse tab Search for *Microsoft.Azure.Stroage.Blob* and on the results Select it and Click Install.



NuGet Package Manager: Azure.FileStorage

Microsoft.Azure.Storage.Blob

**Microsoft.Azure.Storage.Blob** by Microsoft, **326K** downloads v10.0.2
A client library for working with Microsoft Azure Storage Blob service.

**Microsoft.Azure.Storage.DataMovement** by Microsoft, **570K** downloads v0.11.0
A client library designed for high-performance and scalable uploading, downloading and copying data to and from...

**Wintellect.Azure.Storage** by Jeffrey Richter, **10.8K** downloads v1.6.0.1
This package contains Wintellect's Azure Storage library created by Jeffrey Richter. Wintellect has been using this libr...

**Sitecore.Azure.Diagnostics** by Oleg Burov, **3.45K** downloads v8.1.0
Sitecore CMS extension to output log statements to Microsoft Azure Storage service using the Append Blobs.

**WindowsAzure.Storage.NoTables** by Andrei Tarun, **3.27K** downloads v7.2.1.5

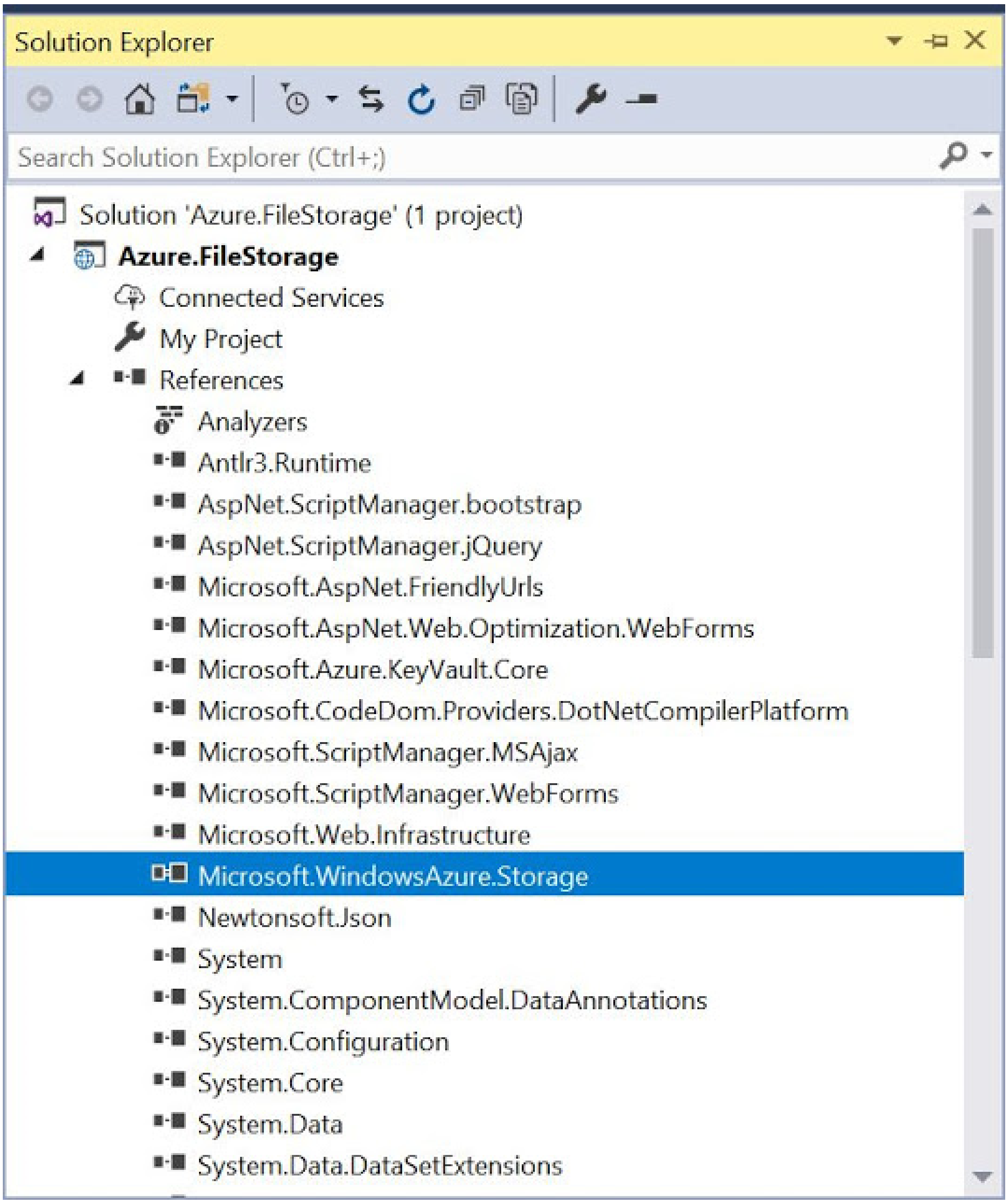
Each package is licensed to you by its owner. NuGet is not responsible for, nor does it grant any licenses to, third-party packages.

□ Do not show this again

Microsoft.Azure.Storage.Blob

Version: 10.0.1

Patrick Sameera Jayalath

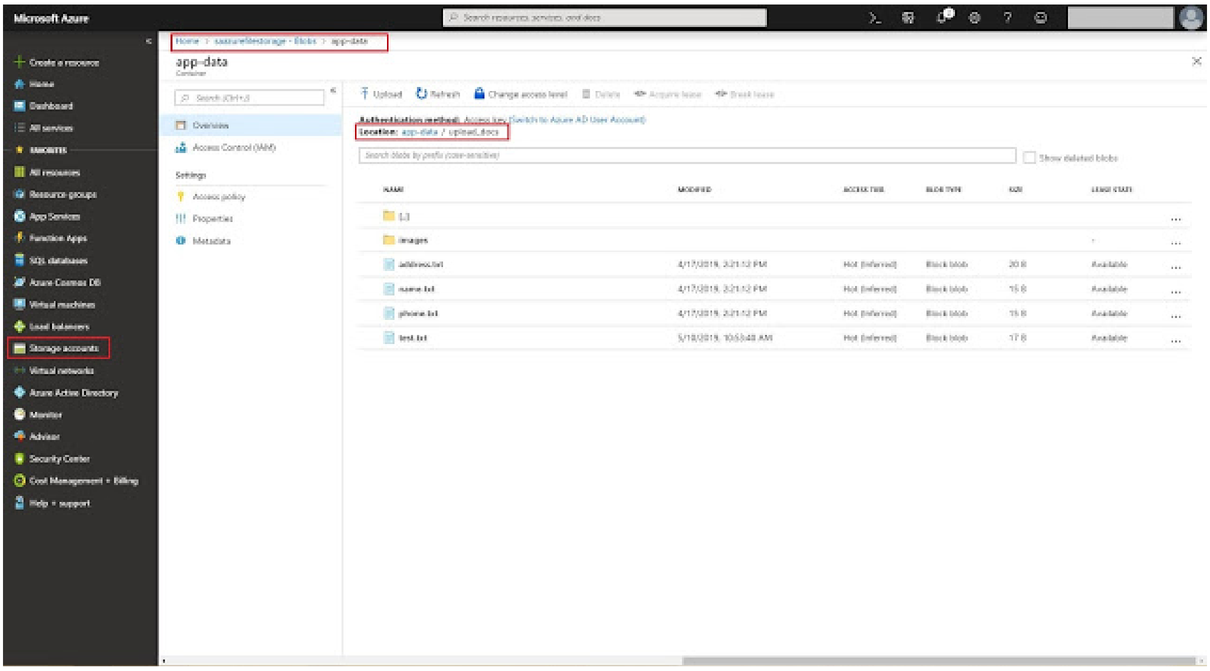


Solution 'Azure.FileStorage' (1 project)

- **Azure.FileStorage**
  - Connected Services
  - My Project
  - References
    - Analyzers
    - Antlr3.Runtime
    - AspNet.ScriptManager.bootstrap
    - AspNet.ScriptManager.jQuery
    - Microsoft.AspNet.FriendlyUrls
    - Microsoft.AspNet.Web.Optimization.WebForms
    - Microsoft.Azure.KeyVault.Core
    - Microsoft.CodeDom.Providers.DotNetCompilerPlatform
    - Microsoft.ScriptManager.MSAjax
    - Microsoft.ScriptManager.WebForms
    - Microsoft.Web.Infrastructure
    - Microsoft.WindowsAzure.Storage
    - Newtonsoft.Json
    - System
    - System.ComponentModel.DataAnnotations
    - System.Configuration
    - System.Core
    - System.Data
    - System.Data.DataSetExtensions

For this tutorial we will be using the same File Storage Account we created earlier tutorial.

*https://github.com/patricksameerajayalath/AzureTutorials/blob/master/Azure%20File%20Storage%20Overview.pdf*



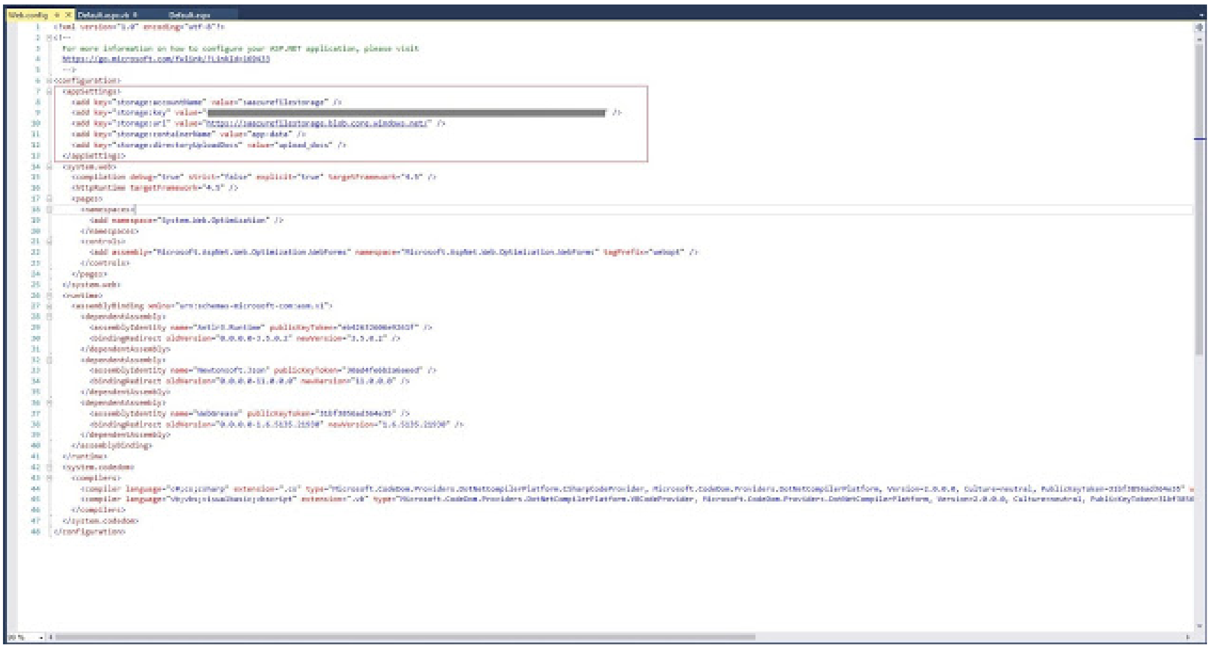
**Authentication method:** Access key (Switch to Azure AD User Account)
**Location:** app-data / upload_docs

| NAME | MODIFIED | ACCESS TIER | BLOB TYPE | SIZE | LEASE STATE | |
|---|---|---|---|---|---|---|
| [..] | | | | | | ... |
| images | | | | | - | ... |
| address.txt | 4/17/2019, 3:21:12 PM | Hot (Inferred) | Block blob | 20 B | Available | ... |
| name.txt | 4/17/2019, 3:21:12 PM | Hot (Inferred) | Block blob | 15 B | Available | ... |
| phone.txt | 4/17/2019, 3:21:12 PM | Hot (Inferred) | Block blob | 15 B | Available | ... |
| test.txt | 5/10/2019, 10:53:40 AM | Hot (Inferred) | Block blob | 17 B | Available | ... |

We will be storing all Storage Account related information in Web.config.

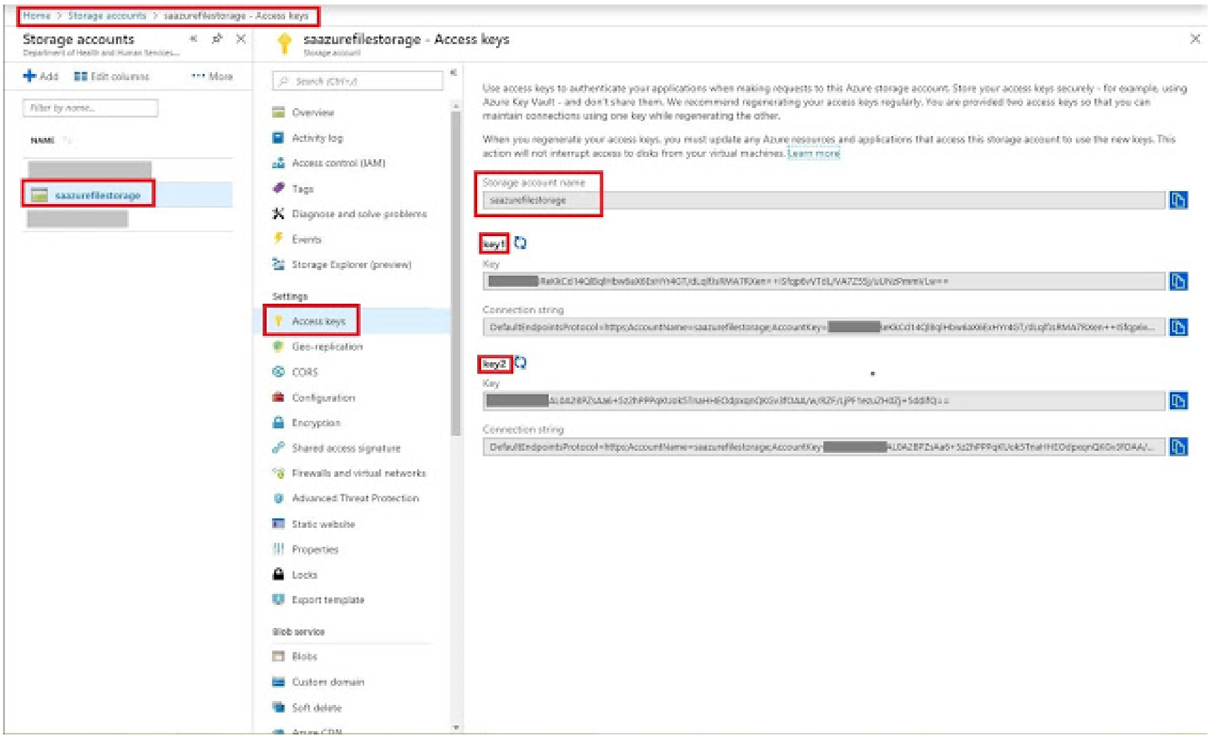
Web.config

*https://github.com/patricksameerajayalath/Azure.FileStorage/blob/master/Web.config*



```xml
<?xml version="1.0" encoding="utf-8"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  https://go.microsoft.com/fwlink/?LinkId=169433
  -->
<configuration>
  <appSettings>
    <add key="storage:accountName" value="saazurefilestorage" />
    <add key="storage:key" value="[redacted]" />
    <add key="storage:url" value="https://saazurefilestorage.blob.core.windows.net/" />
    <add key="storage:containerName" value="app-data" />
    <add key="storage:directoryUploadDocs" value="upload_docs" />
  </appSettings>
  <system.web>
    <compilation debug="true" strict="false" explicit="true" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
    <pages>
      <namespaces>
        <add namespace="System.Web.Optimization" />
      </namespaces>
      <controls>
        <add assembly="Microsoft.AspNet.Web.Optimization.WebForms" namespace="Microsoft.AspNet.Web.Optimization.WebForms" tagPrefix="webopt" />
      </controls>
    </pages>
  </system.web>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="Antlr3.Runtime" publicKeyToken="eb42632606e9261f" />
        <bindingRedirect oldVersion="0.0.0.0-3.5.0.2" newVersion="3.5.0.2" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="Newtonsoft.Json" publicKeyToken="30ad4fe6b2a6aeed" />
        <bindingRedirect oldVersion="0.0.0.0-11.0.0.0" newVersion="11.0.0.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="0.0.0.0-1.6.5135.21930" newVersion="1.6.5135.21930" />
      </dependentAssembly>
    </assemblyBinding>
  </runtime>
  <system.codedom>
    <compilers>
      <compiler language="c#;cs;csharp" extension=".cs" type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.CSharpCodeProvider, Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.8.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" w
      <compiler language="vb;vbs;visualbasic;vbscript" extension=".vb" type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.VBCodeProvider, Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.8.0, Culture=neutral, PublicKeyToken=31bf3856
    </compilers>
  </system.codedom>
</configuration>
```

You can get the connection string from "Access Keys" tab.



Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. Learn more

Storage account name
saazurefilestorage

key1

Key
[redacted]RaKKCd14QBqHbw6aX6EbHYr4GT/dLqf/tsRMATRXen++iSfqp6VVTdL/VA7Z5Sj/uUNzPmmkLw==

Connection string
DefaultEndpointsProtocol=https;AccountName=saazurefilestorage;AccountKey=[redacted]RaKKCd14QBqHbw6aX6EbHYr4GT/dLqf/tsRMATRXen++iSfqpl...

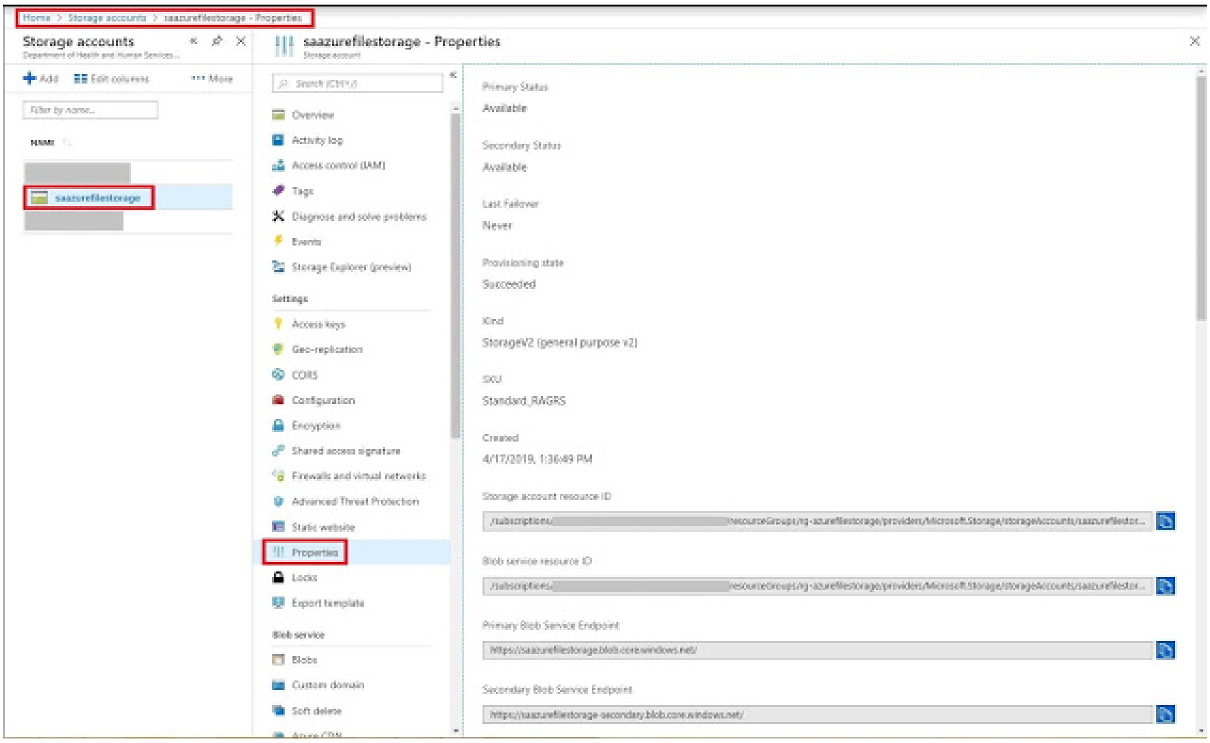
key2

Key
[redacted]AL0A2BFZsAa6+5z2hPPRqKUok5TnaHHEOdpxqnQKGv3fOAA/w/6ZF/jPF1eouZH0Zj+5ddfQ==

Connection string
DefaultEndpointsProtocol=https;AccountName=saazurefilestorage;AccountKey=[redacted]AL0A2BFZsAa6+5z2hPPRqKUok5TnaHHEOdpxqnQKGv3fOAA/...

Patrick Sameera Jayalath

You can get the Blob URL from "Properties" tab.



Primary Status
Available

Secondary Status
Available

Last Failover
Never

Provisioning state
Succeeded

Kind
StorageV2 (general purpose v2)

SKU
Standard_RAGRS

Created
4/17/2019, 1:36:49 PM

Storage account resource ID
/subscriptions/[redacted]/resourceGroups/rg-azurefilestorage/providers/Microsoft.Storage/storageAccounts/saazurefilestor...

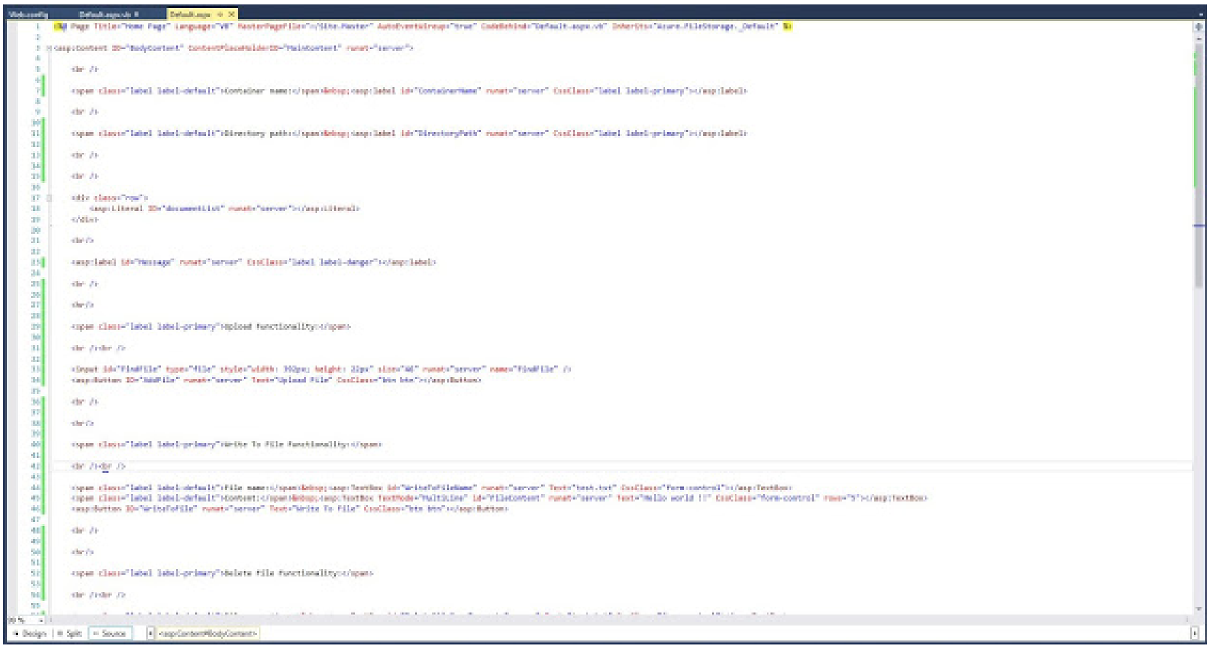
Blob service resource ID
/subscriptions/[redacted]/resourceGroups/rg-azurefilestorage/providers/Microsoft.Storage/storageAccounts/saazurefilestor...

Primary Blob Service Endpoint
https://saazurefilestorage.blob.core.windows.net/

Secondary Blob Service Endpoint
https://saazurefilestorage-secondary.blob.core.windows.net/

Default.aspx

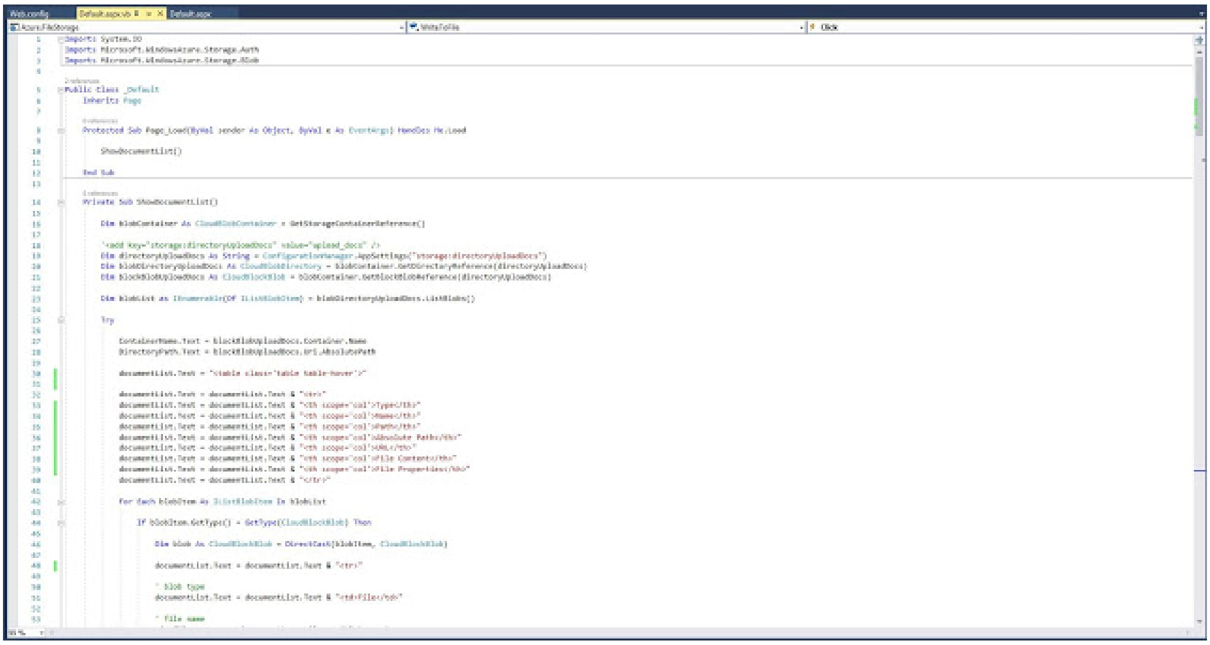
*https://github.com/patricksameerajayalath/Azure.FileStorage/blob/master/Default.aspx*



```
<%@ Page Title="Home Page" Language="VB" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="Default.aspx.vb" Inherits="Azure.FileStorage._Default" %>

<asp:Content ID="BodyContent" ContentPlaceHolderID="MainContent" runat="server">

    <br />

    <span class="label label-default">Container name:</span>&nbsp;<asp:label id="ContainerName" runat="server" CssClass="label label-primary"></asp:label>

    <br />

    <span class="label label-default">Directory path:</span>&nbsp;<asp:label id="DirectoryPath" runat="server" CssClass="label label-primary"></asp:label>

    <br />

    <br />

    <div class="row">
        <asp:Literal ID="documentList" runat="server"></asp:Literal>
    </div>

    <br/>

    <asp:label id="Message" runat="server" CssClass="label label-danger"></asp:label>

    <br />

    <br/>

    <span class="label label-primary">Upload functionality:</span>

    <br /><br />

    <input id="FindFile" type="file" style="width: 392px; height: 22px" size="40" runat="server" name="FindFile" />
    <asp:Button ID="AddFile" runat="server" Text="Upload File" CssClass="btn btn"></asp:Button>

    <br />

    <br/>

    <span class="label label-primary">Write To File Functionality:</span>

    <br /><br />

    <span class="label label-default">File name:</span>&nbsp;<asp:TextBox id="WriteToFileName" runat="server" Text="test.txt" CssClass="form-control"></asp:TextBox>
    <span class="label label-default">Content:</span>&nbsp;<asp:TextBox TextMode="MultiLine" id="FileContent" runat="server" Text="Hello world !!" CssClass="form-control" rows="5"></asp:TextBox>
    <asp:Button ID="WriteToFile" runat="server" Text="Write To File" CssClass="btn btn"></asp:Button>

    <br />

    <br/>

    <span class="label label-primary">Delete file functionality:</span>

    <br /><br />
```

Default.aspx.vb

*https://github.com/patricksameerajayalath/Azure.FileStorage/blob/master/Default.aspx.vb*



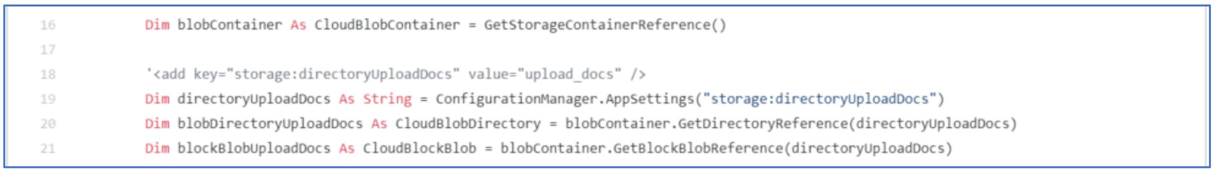
```
Imports System.IO
Imports Microsoft.WindowsAzure.Storage.Auth
Imports Microsoft.WindowsAzure.Storage.Blob

Public Class _Default
    Inherits Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As EventArgs) Handles Me.Load

        ShowDocumentList()

    End Sub

    Private Sub ShowDocumentList()

        Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

        '<add key="storage:directoryUploadDocs" value="upload_docs" />
        Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
        Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
        Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)

        Dim blobList as IEnumerable(Of IListBlobItem) = blobDirectoryUploadDocs.ListBlobs()

        Try

            ContainerName.Text = blockBlobUploadDocs.Container.Name
            DirectoryPath.Text = blockBlobUploadDocs.Uri.AbsolutePath

            documentList.Text = "<table class='table table-hover'>"

            documentList.Text = documentList.Text & "<tr>"
            documentList.Text = documentList.Text & "<th scope='col'>Type</th>"
            documentList.Text = documentList.Text & "<th scope='col'>Name</th>"
            documentList.Text = documentList.Text & "<th scope='col'>Path</th>"
            documentList.Text = documentList.Text & "<th scope='col'>Absolute Path</th>"
            documentList.Text = documentList.Text & "<th scope='col'>URL</th>"
            documentList.Text = documentList.Text & "<th scope='col'>File Content</th>"
            documentList.Text = documentList.Text & "<th scope='col'>File Properties</th>"
            documentList.Text = documentList.Text & "</tr>"

            For Each blobItem As IListBlobItem In blobList

                If blobItem.GetType() = GetType(CloudBlockBlob) Then

                    Dim blob As CloudBlockBlob = DirectCast(blobItem, CloudBlockBlob)

                    documentList.Text = documentList.Text & "<tr>"

                    ' blob type
                    documentList.Text = documentList.Text & "<td>File</td>"

                    ' file name
```

Patrick Sameera Jayalath

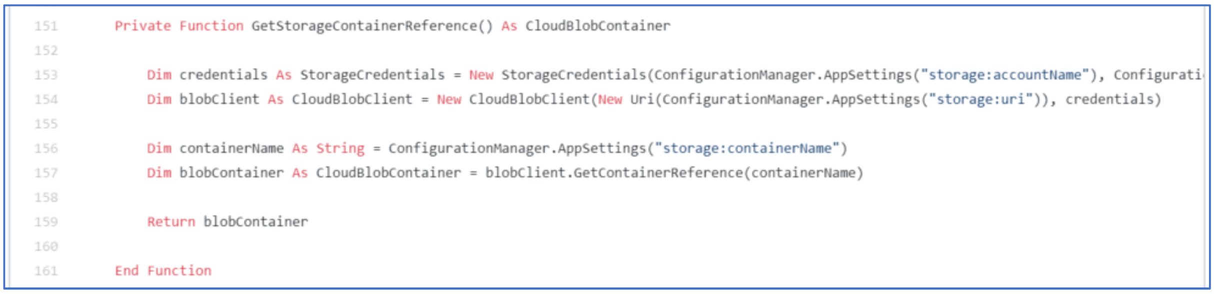
## Code Snippets:

*[https://github.com/patricksameerajayalath/Azure.FileStorage/blob/master/Default.aspx.vb](https://github.com/patricksameerajayalath/Azure.FileStorage/blob/master/Default.aspx.vb)*

### Code 01 - Connect to Azure Storage

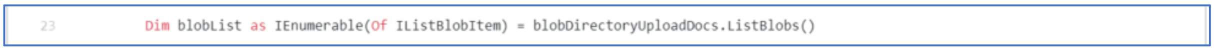


```
        Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

        '<add key="storage:directoryUploadDocs" value="upload_docs" />
        Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
        Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
        Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)
```

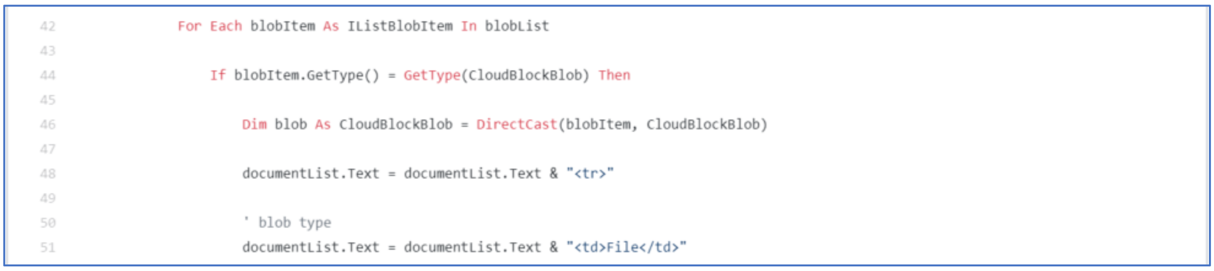


```
    Private Function GetStorageContainerReference() As CloudBlobContainer

        Dim credentials As StorageCredentials = New StorageCredentials(ConfigurationManager.AppSettings("storage:accountName"), Configurati
        Dim blobClient As CloudBlobClient = New CloudBlobClient(New Uri(ConfigurationManager.AppSettings("storage:uri")), credentials)

        Dim containerName As String = ConfigurationManager.AppSettings("storage:containerName")
        Dim blobContainer As CloudBlobContainer = blobClient.GetContainerReference(containerName)

        Return blobContainer

    End Function
```

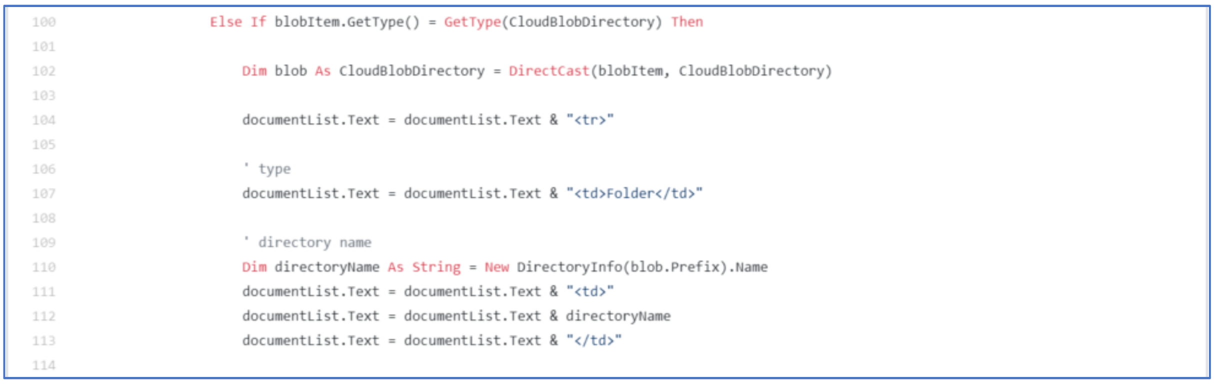
### Code 02 - List files/folders (CloudBlockBlob/CloudBlobDirectory) within an Azure File Storage



```
        Dim blobList as IEnumerable(Of IListBlobItem) = blobDirectoryUploadDocs.ListBlobs()
```

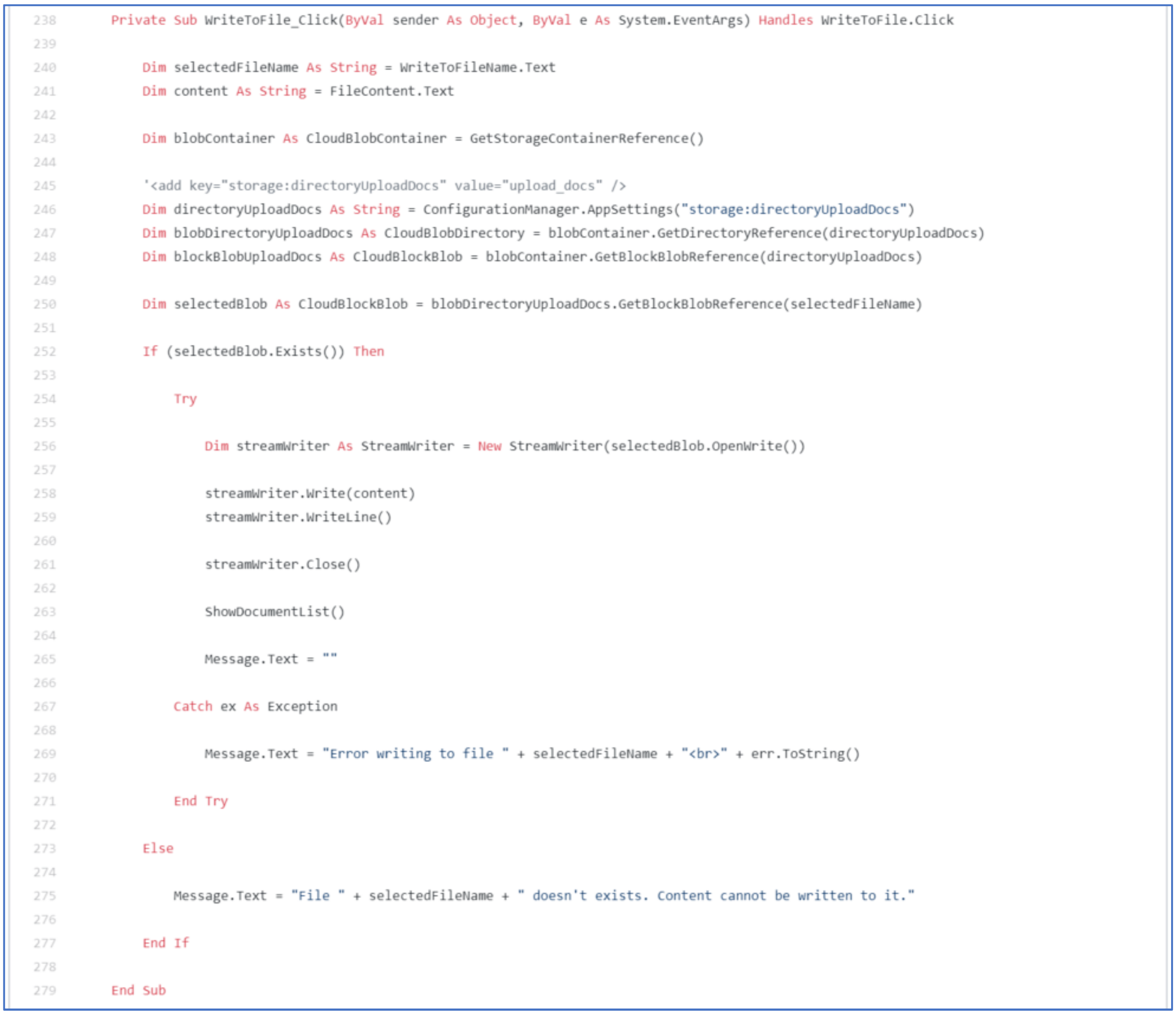


```
            For Each blobItem As IListBlobItem In blobList

                If blobItem.GetType() = GetType(CloudBlockBlob) Then

                    Dim blob As CloudBlockBlob = DirectCast(blobItem, CloudBlockBlob)

                    documentList.Text = documentList.Text & "<tr>"

                    ' blob type
                    documentList.Text = documentList.Text & "<td>File</td>"
```



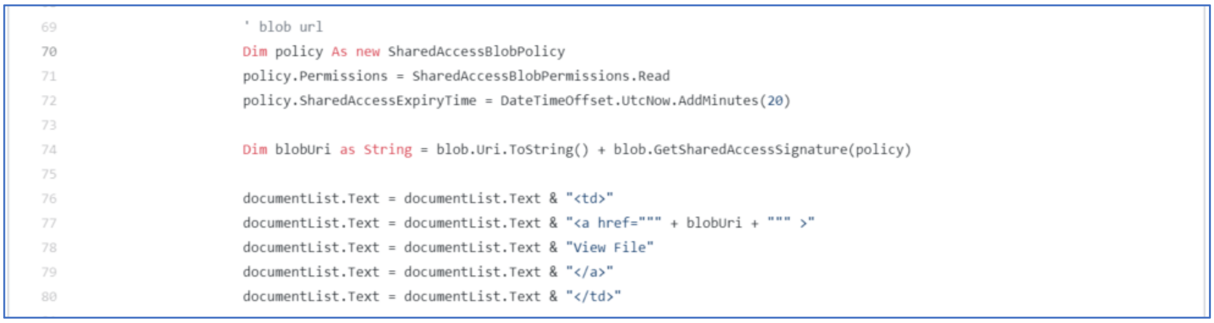
```
                Else If blobItem.GetType() = GetType(CloudBlobDirectory) Then

                    Dim blob As CloudBlobDirectory = DirectCast(blobItem, CloudBlobDirectory)

                    documentList.Text = documentList.Text & "<tr>"

                    ' type
                    documentList.Text = documentList.Text & "<td>Folder</td>"

                    ' directory name
                    Dim directoryName As String = New DirectoryInfo(blob.Prefix).Name
                    documentList.Text = documentList.Text & "<td>"
                    documentList.Text = documentList.Text & directoryName
                    documentList.Text = documentList.Text & "</td>"

```

## Code 03 - Write content to a Blob



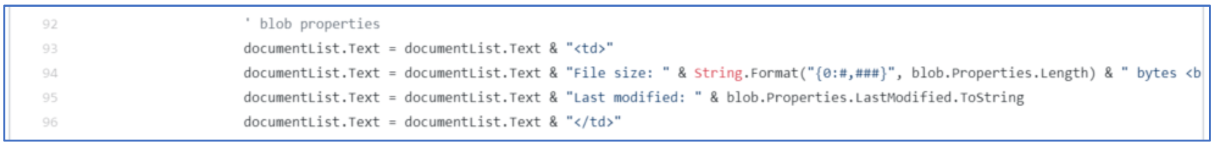
```
Private Sub WriteToFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles WriteToFile.Click

    Dim selectedFileName As String = WriteToFileName.Text
    Dim content As String = FileContent.Text

    Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

    '<add key="storage:directoryUploadDocs" value="upload_docs" />
    Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
    Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
    Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)

    Dim selectedBlob As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(selectedFileName)

    If (selectedBlob.Exists()) Then

        Try

            Dim streamWriter As StreamWriter = New StreamWriter(selectedBlob.OpenWrite())

            streamWriter.Write(content)
            streamWriter.WriteLine()

            streamWriter.Close()

            ShowDocumentList()

            Message.Text = ""

        Catch ex As Exception

            Message.Text = "Error writing to file " + selectedFileName + "<br>" + err.ToString()

        End Try

    Else

        Message.Text = "File " + selectedFileName + " doesn't exists. Content cannot be written to it."

    End If

End Sub
```

## Code 04 - Create a Shared Access Signature for a Blob



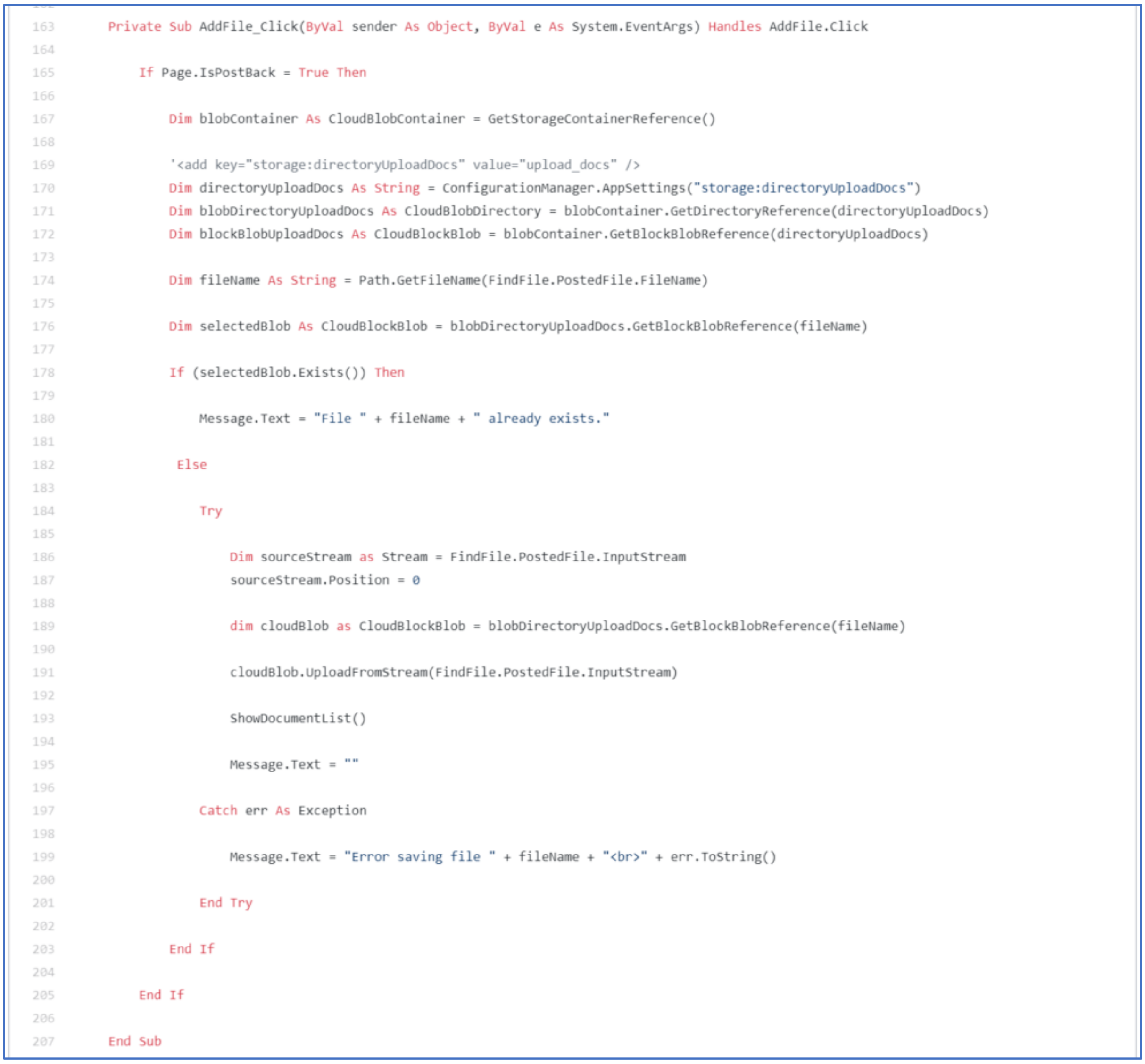
```
' blob url
Dim policy As new SharedAccessBlobPolicy
policy.Permissions = SharedAccessBlobPermissions.Read
policy.SharedAccessExpiryTime = DateTimeOffset.UtcNow.AddMinutes(20)

Dim blobUri as String = blob.Uri.ToString() + blob.GetSharedAccessSignature(policy)

documentList.Text = documentList.Text & "<td>"
documentList.Text = documentList.Text & "<a href=""" + blobUri + """ >"
documentList.Text = documentList.Text & "View File"
documentList.Text = documentList.Text & "</a>"
documentList.Text = documentList.Text & "</td>"
```

## Code 05 - Obtain Blob properties



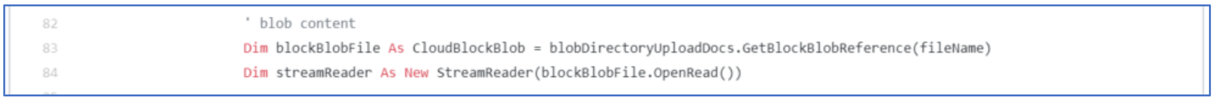
```
' blob properties
documentList.Text = documentList.Text & "<td>"
documentList.Text = documentList.Text & "File size: " & String.Format("{0:#,###}", blob.Properties.Length) & " bytes <b
documentList.Text = documentList.Text & "Last modified: " & blob.Properties.LastModified.ToString
documentList.Text = documentList.Text & "</td>"
```

## Code 06 - Upload a Blob to Azure File Storage



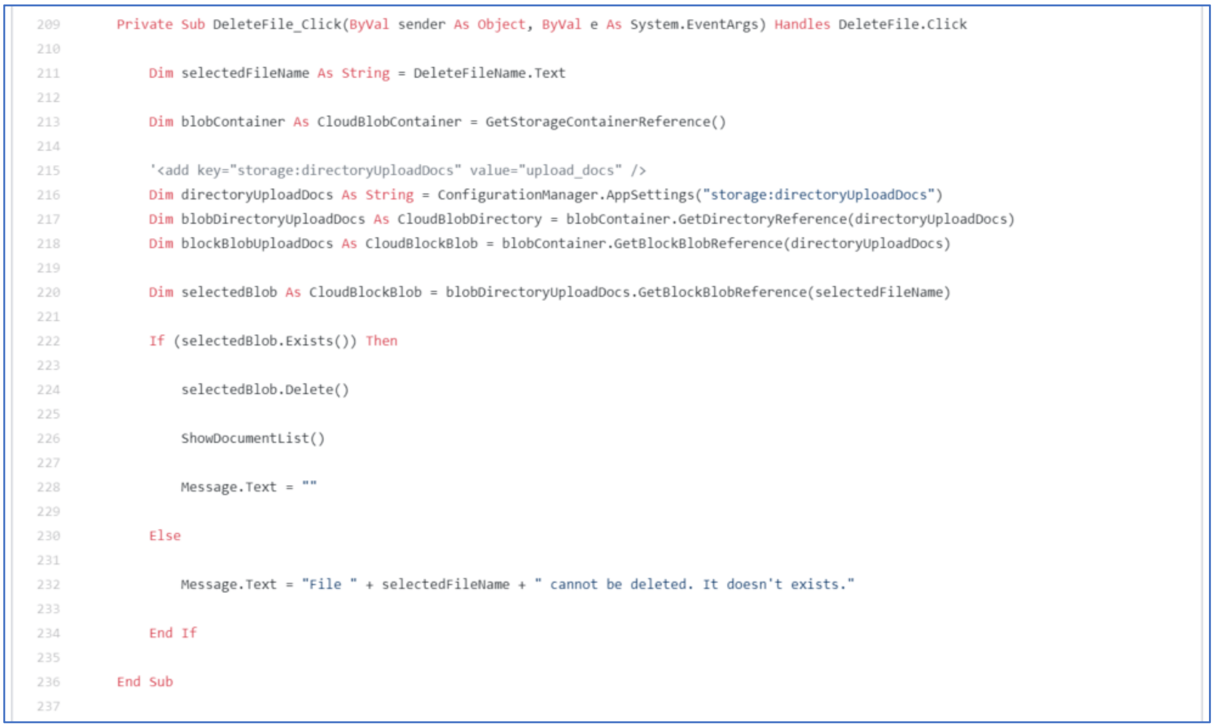
```vb
Private Sub AddFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles AddFile.Click

    If Page.IsPostBack = True Then

        Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

        '<add key="storage:directoryUploadDocs" value="upload_docs" />
        Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
        Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
        Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)

        Dim fileName As String = Path.GetFileName(FindFile.PostedFile.FileName)

        Dim selectedBlob As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(fileName)

        If (selectedBlob.Exists()) Then

            Message.Text = "File " + fileName + " already exists."

         Else

            Try

                Dim sourceStream as Stream = FindFile.PostedFile.InputStream
                sourceStream.Position = 0

                dim cloudBlob as CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(fileName)

                cloudBlob.UploadFromStream(FindFile.PostedFile.InputStream)

                ShowDocumentList()

                Message.Text = ""

            Catch err As Exception

                Message.Text = "Error saving file " + fileName + "<br>" + err.ToString()

            End Try

        End If

    End If

End Sub
```

## Code 07 - Read content from a Blob



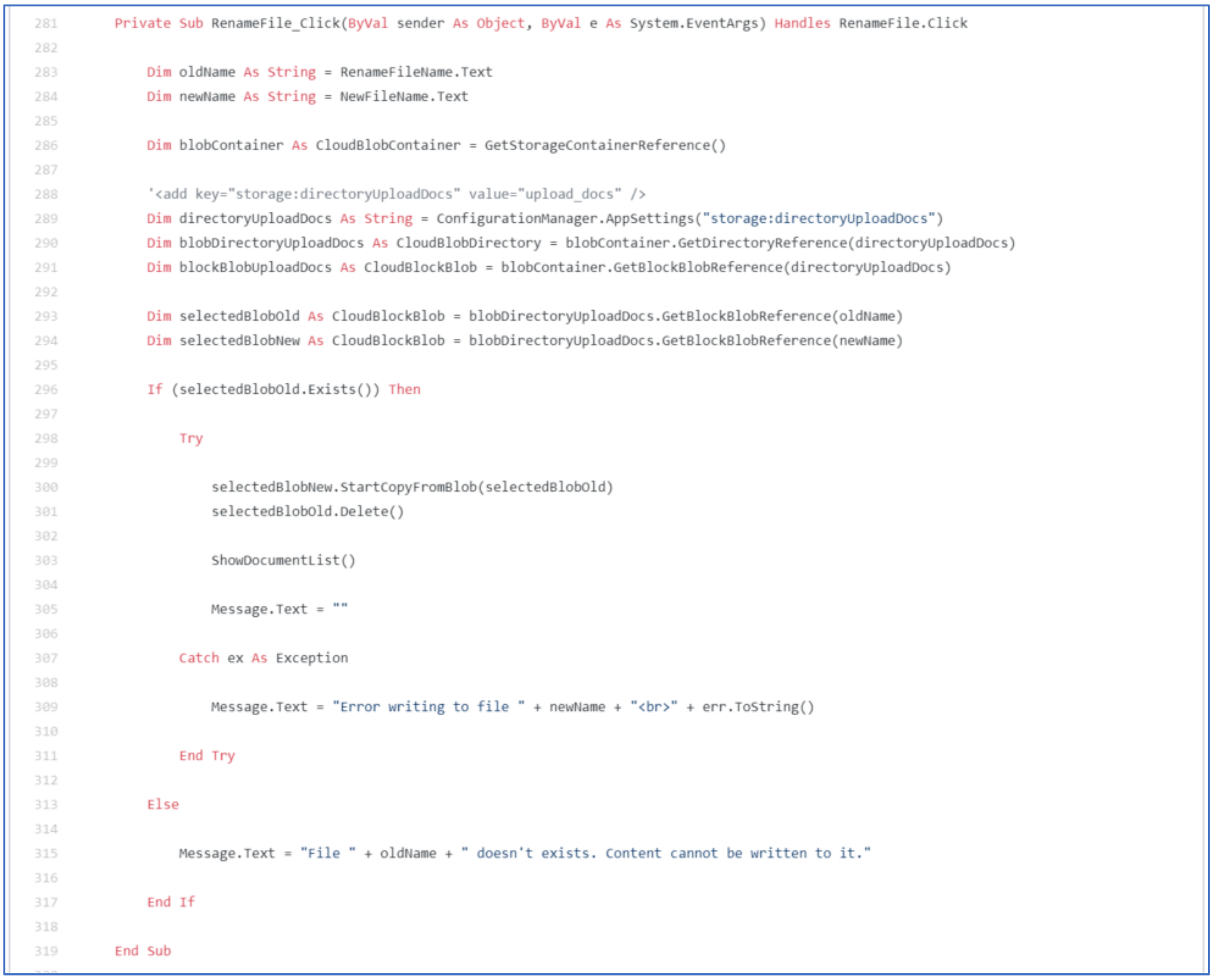
```vb
' blob content
Dim blockBlobFile As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(fileName)
Dim streamReader As New StreamReader(blockBlobFile.OpenRead())
```

## Code 08 - Delete a Blob



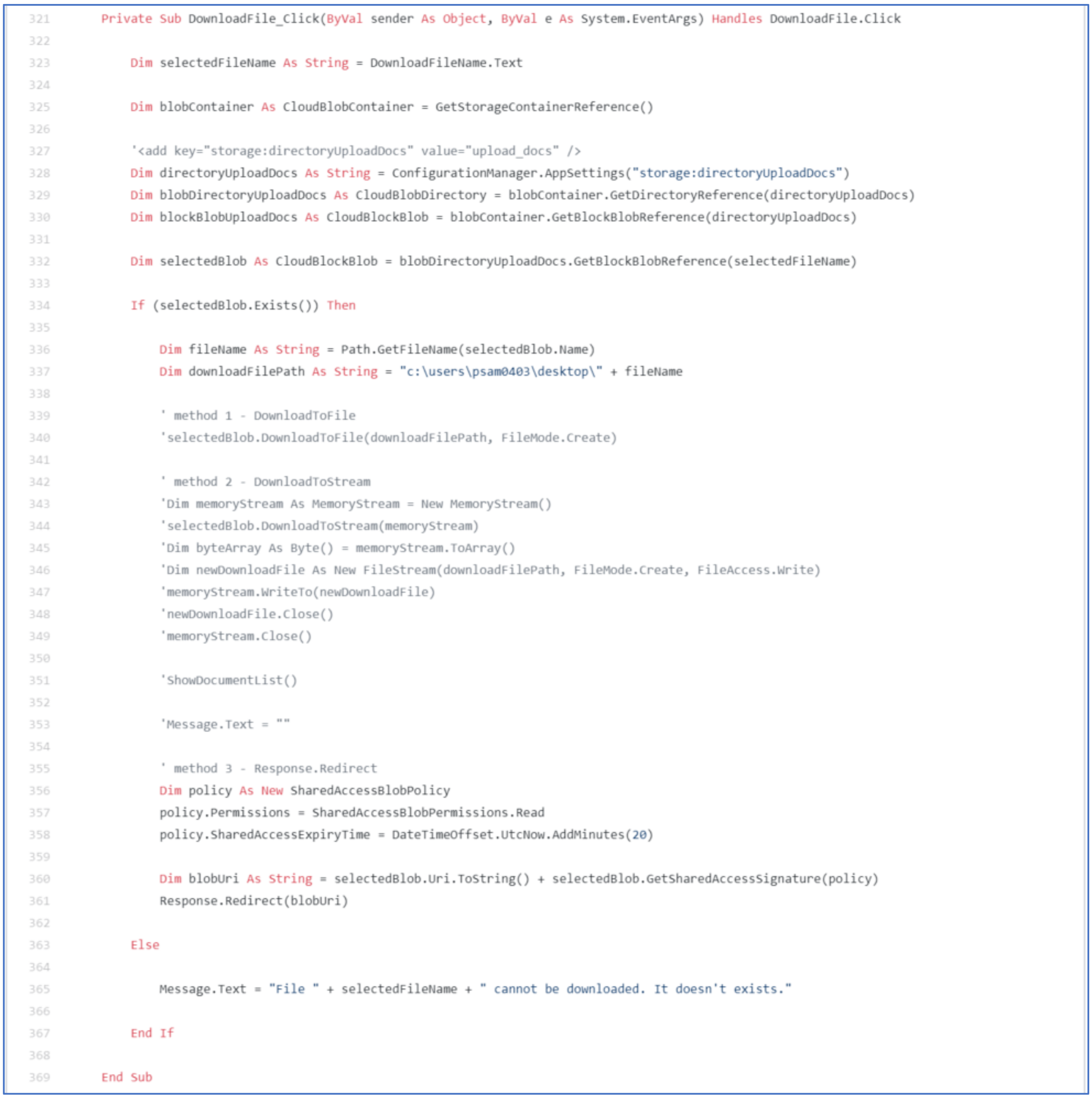
```
Private Sub DeleteFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles DeleteFile.Click

    Dim selectedFileName As String = DeleteFileName.Text

    Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

    '<add key="storage:directoryUploadDocs" value="upload_docs" />
    Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
    Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
    Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)

    Dim selectedBlob As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(selectedFileName)

    If (selectedBlob.Exists()) Then

        selectedBlob.Delete()

        ShowDocumentList()

        Message.Text = ""

    Else

        Message.Text = "File " + selectedFileName + " cannot be deleted. It doesn't exists."

    End If

End Sub
```

Code 09 - Rename/Move a Blob



```
Private Sub RenameFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles RenameFile.Click

    Dim oldName As String = RenameFileName.Text
    Dim newName As String = NewFileName.Text

    Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

    '<add key="storage:directoryUploadDocs" value="upload_docs" />
    Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
    Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
    Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)

    Dim selectedBlobOld As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(oldName)
    Dim selectedBlobNew As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(newName)

    If (selectedBlobOld.Exists()) Then

        Try

            selectedBlobNew.StartCopyFromBlob(selectedBlobOld)
            selectedBlobOld.Delete()

            ShowDocumentList()

            Message.Text = ""

        Catch ex As Exception

            Message.Text = "Error writing to file " + newName + "<br>" + err.ToString()

        End Try

    Else

        Message.Text = "File " + oldName + " doesn't exists. Content cannot be written to it."

    End If

End Sub
```

## Code 10 - Download a Blob



```
Private Sub DownloadFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles DownloadFile.Click

    Dim selectedFileName As String = DownloadFileName.Text

    Dim blobContainer As CloudBlobContainer = GetStorageContainerReference()

    '<add key="storage:directoryUploadDocs" value="upload_docs" />
    Dim directoryUploadDocs As String = ConfigurationManager.AppSettings("storage:directoryUploadDocs")
    Dim blobDirectoryUploadDocs As CloudBlobDirectory = blobContainer.GetDirectoryReference(directoryUploadDocs)
    Dim blockBlobUploadDocs As CloudBlockBlob = blobContainer.GetBlockBlobReference(directoryUploadDocs)

    Dim selectedBlob As CloudBlockBlob = blobDirectoryUploadDocs.GetBlockBlobReference(selectedFileName)

    If (selectedBlob.Exists()) Then

        Dim fileName As String = Path.GetFileName(selectedBlob.Name)
        Dim downloadFilePath As String = "c:\users\psam0403\desktop\" + fileName

        ' method 1 - DownloadToFile
        'selectedBlob.DownloadToFile(downloadFilePath, FileMode.Create)

        ' method 2 - DownloadToStream
        'Dim memoryStream As MemoryStream = New MemoryStream()
        'selectedBlob.DownloadToStream(memoryStream)
        'Dim byteArray As Byte() = memoryStream.ToArray()
        'Dim newDownloadFile As New FileStream(downloadFilePath, FileMode.Create, FileAccess.Write)
        'memoryStream.WriteTo(newDownloadFile)
        'newDownloadFile.Close()
        'memoryStream.Close()

        'ShowDocumentList()

        'Message.Text = ""

        ' method 3 - Response.Redirect
        Dim policy As New SharedAccessBlobPolicy
        policy.Permissions = SharedAccessBlobPermissions.Read
        policy.SharedAccessExpiryTime = DateTimeOffset.UtcNow.AddMinutes(20)

        Dim blobUri As String = selectedBlob.Uri.ToString() + selectedBlob.GetSharedAccessSignature(policy)
        Response.Redirect(blobUri)

    Else

        Message.Text = "File " + selectedFileName + " cannot be downloaded. It doesn't exists."

    End If

End Sub
```