Get started with Azure Storage in five minutes

By Tamra Myers (https://github.com/tamram)

Updated: 02/14/2016

In this article:

Overview

It's easy to get started developing with Azure Storage. This tutorial shows you how to get an Azure Storage application up and running quickly. You'll use the Quick Start templates included with the Azure SDK for .NET. These Quick Starts contain ready-to-run code that demonstrates some basic programming scenarios with Azure Storage.

To learn more about Azure Storage before diving into the code, see Next Steps.

Prerequisites

You'll need the following prerequisites before you start:

1. To compile and build the application, you'll need a version of Visual Studio (https://www.visualstudio.com/) installed on your computer.
2. Install the latest version Azure SDK for .NET (https://azure.microsoft.com/downloads/). The SDK includes the Azure QuickStart sample projects, the Azure storage emulator, and the Azure Storage Client Library for .NET (https://msdn.microsoft.com/library/azure/dn261237.aspx).

3. Make sure that you have .NET Framework 4.5 (http://www.microsoft.com/download/details.aspx?id=30653) installed on your computer, as it is required by the Azure QuickStart sample projects that we'll be using in this tutorial.

   If you are not sure which version of .NET Framework is installed in your computer, see How to: Determine Which .NET Framework Versions Are Installed (https://msdn.microsoft.com/vstudio/hh925568.aspx). Or, press the **Start** button or the Windows key, type **Control Panel**. Then, click **Programs** > **Programs and Features**, and determine whether the .NET Framework 4.5 is listed among the installed programs.

4. You'll need an Azure subscription and an Azure storage account.

   - To get an Azure subscription, see Free Trial (https://azure.microsoft.com/pricing/free-trial/), Purchase Options (https://azure.microsoft.com/pricing/purchase-options/), and Member Offers (https://azure.microsoft.com/pricing/member-offers/) (for members of MSDN, Microsoft Partner Network, and BizSpark, and other Microsoft programs).

   - To create a storage account in Azure, see How to create a storage account (../storage-create-storage-account/#create-a-storage-account).

Run your first Azure Storage application against Azure Storage in the cloud

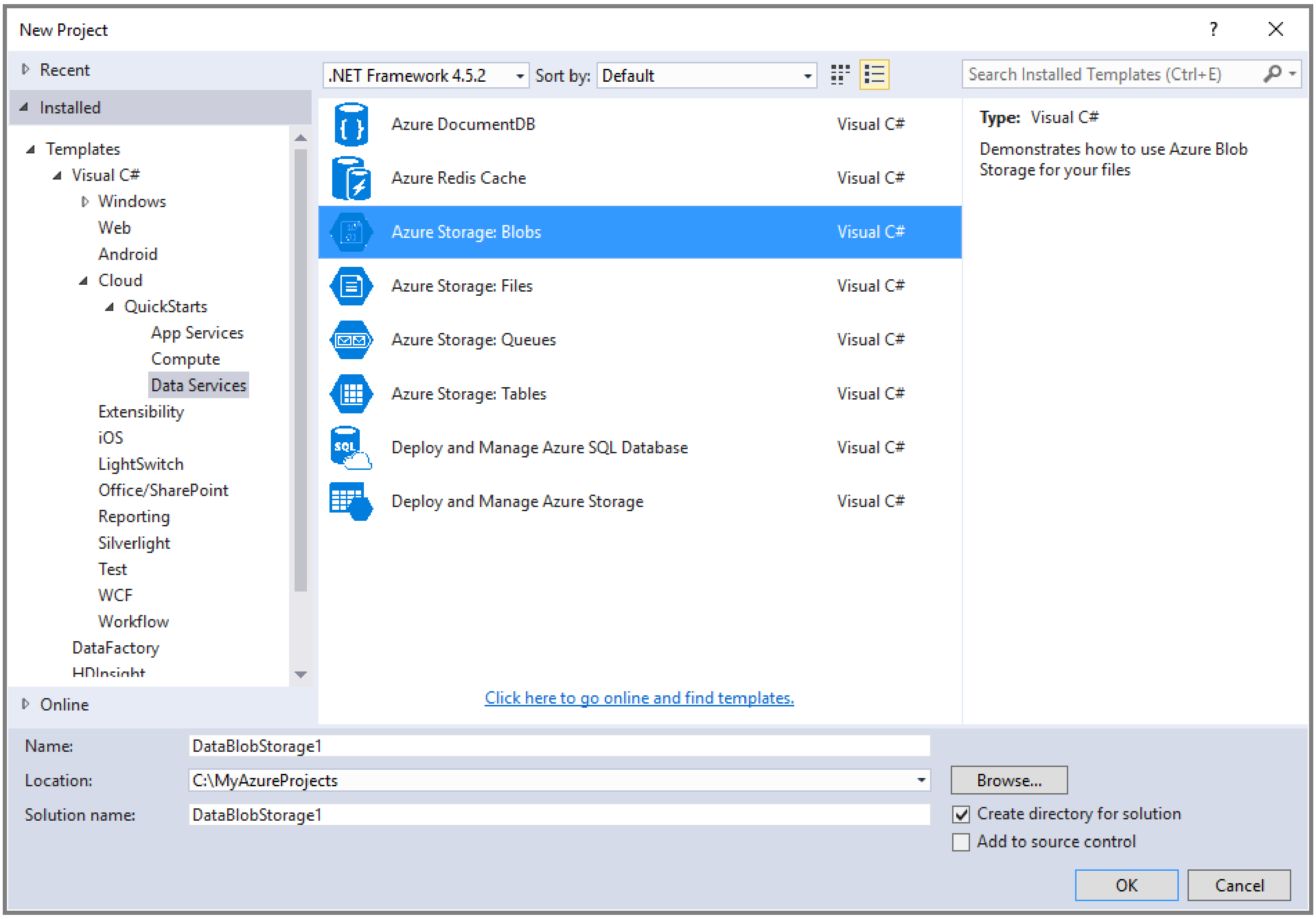
Once you have an account, you can create a simple Azure Storage application using one of the Azure Quick Starts sample projects in Visual Studio. This tutorial focuses on the sample projects for Azure Storage: **Azure Storage: Blobs**, **Azure Storage: Files**, **Azure Storage: Queues**, and **Azure Storage: Tables**:

1. Start Visual Studio.

2. From the **File** menu, click **New Project**.

3. In the **New Project** dialog box, click **Installed** > **Templates** > **Visual C#** > **Cloud** > **QuickStarts** > **Data Services**. a. Choose one of the following templates: **Azure Storage: Blobs**, **Azure Storage: Files**, **Azure Storage: Queues**, or **Azure Storage: Tables**. b. Make sure that **.NET Framework 4.5** is selected as the target framework.

   - 3.c. Specify a name for your project and create the new Visual Studio solution, as shown:


New Project
?
Recent
Installed
Templates
Visual C#
Windows
Web
Android
Cloud
QuickStarts
App Services
Compute
Data Services
Extensibility
iOS
LightSwitch
Office/SharePoint
Reporting
Silverlight
Test
WCF
Workflow
DataFactory
HDInsight
Online
.NET Framework 4.5.2
Sort by: Default
Search Installed Templates (Ctrl+E)
Azure DocumentDB — Visual C#
Azure Redis Cache — Visual C#
Azure Storage: Blobs — Visual C#
Azure Storage: Files — Visual C#
Azure Storage: Queues — Visual C#
Azure Storage: Tables — Visual C#
Deploy and Manage Azure SQL Database — Visual C#
Deploy and Manage Azure Storage — Visual C#
Click here to go online and find templates.
Browse...
OK
Cancel

Type: Visual C#

Demonstrates how to use Azure Blob Storage for your files

Name: DataBlobStorage1

Location: C:\MyAzureProjects

Solution name: DataBlobStorage1

☑ Create directory for solution

□ Add to source control

You may want to review the source code before running the application. To review the code, select **Solution Explorer** on the **View** menu in Visual Studio. Then, double click the Program.cs file.

Next, run the sample application:

1. In Visual Studio, select **Solution Explorer** on the **View** menu. Open the App.config file and comment out the connection string for the Azure storage emulator:

   ```
   <!--<add key="StorageConnectionString" value = "UseDevelopmentStorage=true;"/>-->
   ```

2. Uncomment the connection string for the Azure Storage Service and provide the storage account name and access key in the App.config file: `<add key="StorageConnectionString" value="DefaultEndpointsProtocol=https;AccountName=[AccountName];AccountKey=[AccountKey]"`

   To retrieve your storage account access key, see Manage your storage access keys (../storage-create-storage-account/#manage-your-storage-access-keys).

3. After you provide the storage account name and access key in the App.config file, on the **File** menu, click **Save All** to save all the project files.

4. On the **Build** menu, click **Build Solution**.

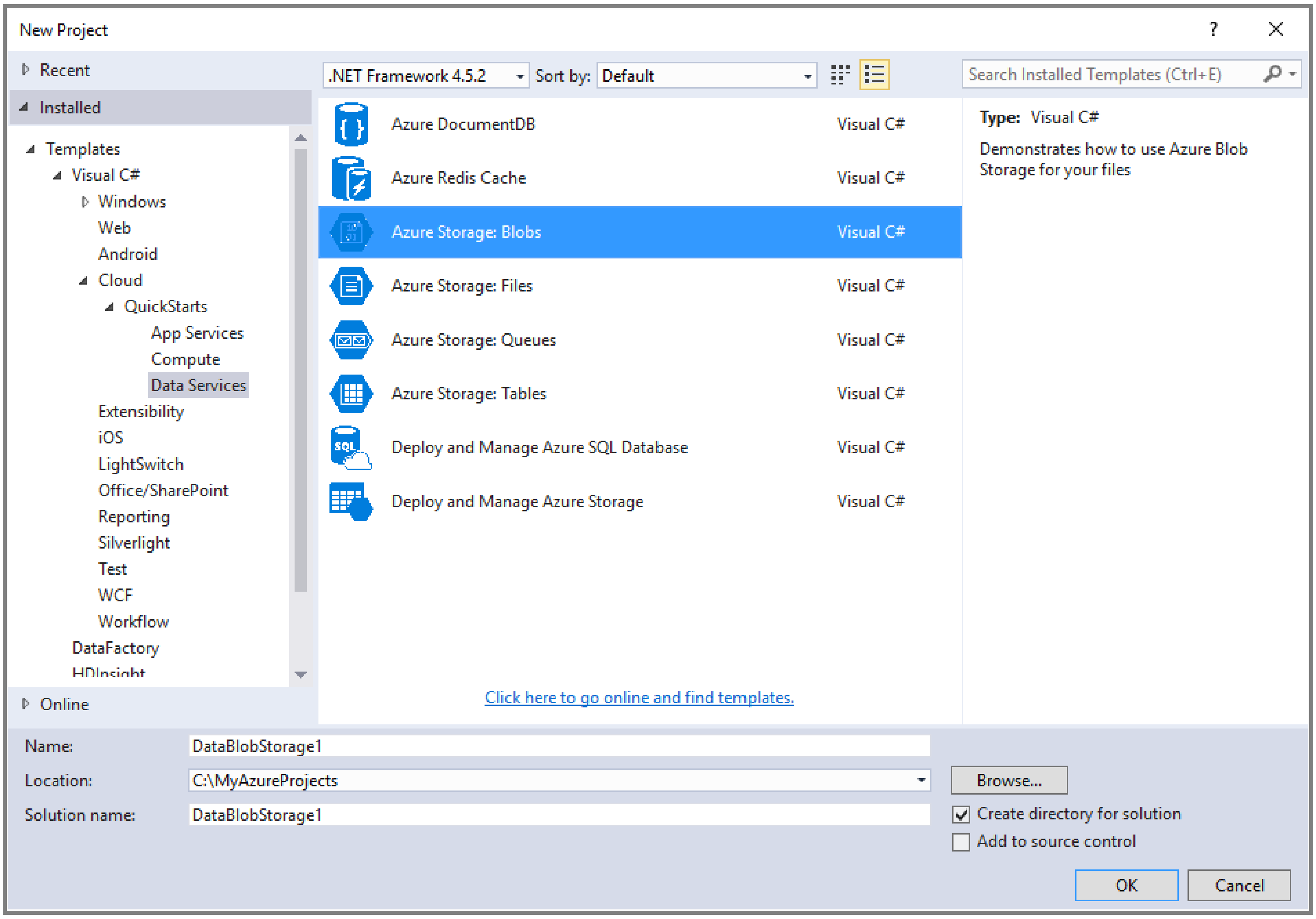
5. On the **Debug** menu, Press **F11** to run the solution step by step or press **F5** to run the solution.

Run your first Azure Storage application locally against the Azure Storage Emulator

The Azure Storage Emulator (../storage-use-emulator/) provides a local environment that emulates the Azure Blob, Queue, and Table services for development purposes. You can use the storage emulator to test your storage application locally, without creating an Azure subscription or storage account, and without incurring any cost.

To try it, let's create a simple Azure Storage application using one of the Azure Quick Starts sample projects in Visual Studio. This tutorial focuses on the **Azure Blob Storage**, **Azure Table Storage**, and **Azure Queue Storage** sample projects:

1. Start Visual Studio.
2. From the **File** menu, click **New Project**.
3. In the **New Project** dialog box, click **Installed** > **Templates** > **Visual C#** > **Cloud** > **QuickStarts** > **Data Services**. a. Choose one of the following templates: **Azure Storage: Blobs**, **Azure Storage: Files**, **Azure Storage: Queues**, or **Azure Storage: Tables**. b. Make sure that **.NET Framework 4.5** is selected as the target framework.
   c. Specify a name for your project and create the new Visual Studio solution, as shown:



.NET Framework 4.5.2

Sort by: Default

Azure DocumentDB	Visual C#
Azure Redis Cache	Visual C#
Azure Storage: Blobs	Visual C#
Azure Storage: Files	Visual C#
Azure Storage: Queues	Visual C#
Azure Storage: Tables	Visual C#
Deploy and Manage Azure SQL Database	Visual C#
Deploy and Manage Azure Storage	Visual C#

Type: Visual C#

Demonstrates how to use Azure Blob Storage for your files

Name: DataBlobStorage1

Location: C:\MyAzureProjects

Solution name: DataBlobStorage1

☑ Create directory for solution

□ Add to source control

4. In Visual Studio, select **Solution Explorer** on the **View** menu. Open the App.config file and comment out the connection string for your Azure storage account if you have already added one. Then uncomment the connection string for the Azure storage emulator:

   ```
   <add key="StorageConnectionString" value = "UseDevelopmentStorage=true;"/>
   ```

You may want to review the source code before running the application. To review the code, select **Solution Explorer** on the **View** menu in Visual Studio. Then, double click the Program.cs file.

Next, run the sample application in the Azure Storage Emulator:

1. Press the **Start** button or the Windows key, search for *Microsoft Azure Storage emulator*, and start the application. When the emulator starts, you'll see an icon and a notification in the Windows Task View area.
2. In Visual Studio, click **Build Solution** on the **Build** menu.
3. On the **Debug** menu, Press **F11** to run the solution step by step, or press **F5** to run the solution from start to finish.

Next Steps

See these resources to learn more about Azure Storage:

- Introduction to Microsoft Azure Storage (../storage-introduction/)
- Get started with Azure Blob storage using .NET (../storage-dotnet-how-to-use-blobs/)
- Get started with Azure Table storage using .NET (../storage-dotnet-how-to-use-tables/)
- Get started with Azure Queue Storage using .NET (../storage-dotnet-how-to-use-queues/)
- Get started with Azure File storage on Windows (../storage-dotnet-how-to-use-files/)
- Transfer data with the AzCopy Command-Line Utility (../storage-use-azcopy/)
- Azure Storage Documentation (https://azure.microsoft.com/documentation/services/storage/)
- Microsoft Azure Storage Client Library for .NET (https://msdn.microsoft.com/library/azure/dn261237.aspx)
- Azure Storage Services REST API (https://msdn.microsoft.com/library/azure/dd179355.aspx)