

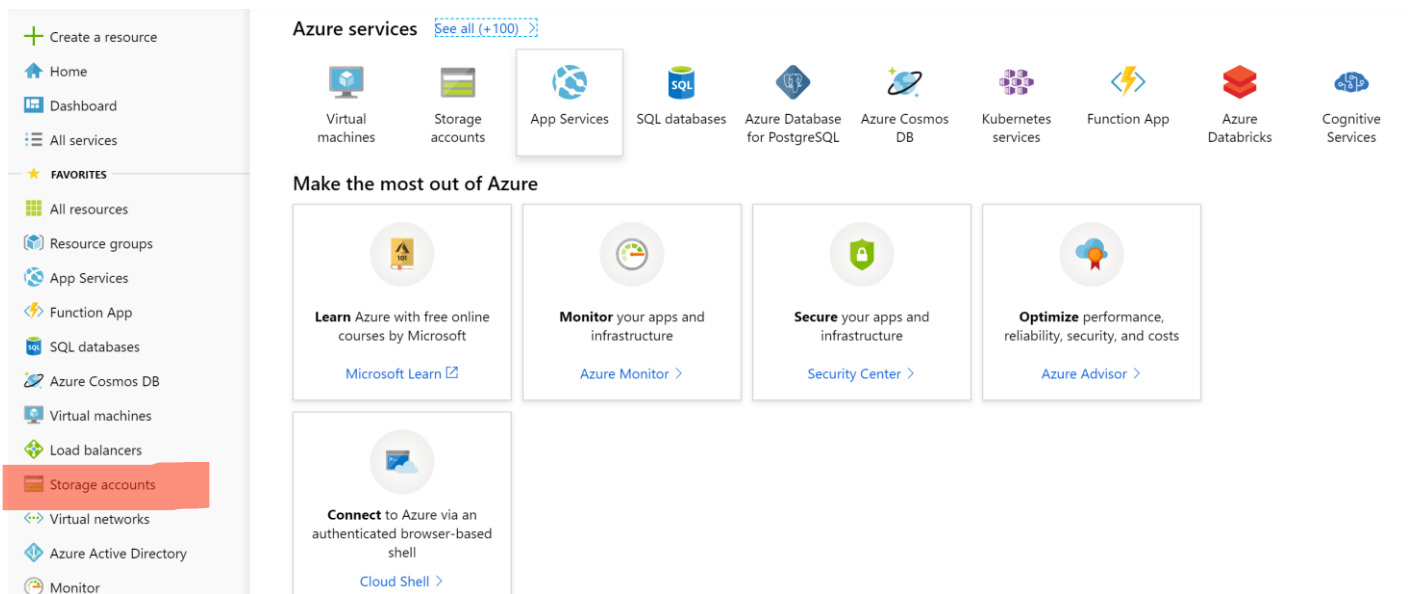
# Azure Blob Storage using ASP.NET Web API

## Step 1: Set up Azure Storage Account

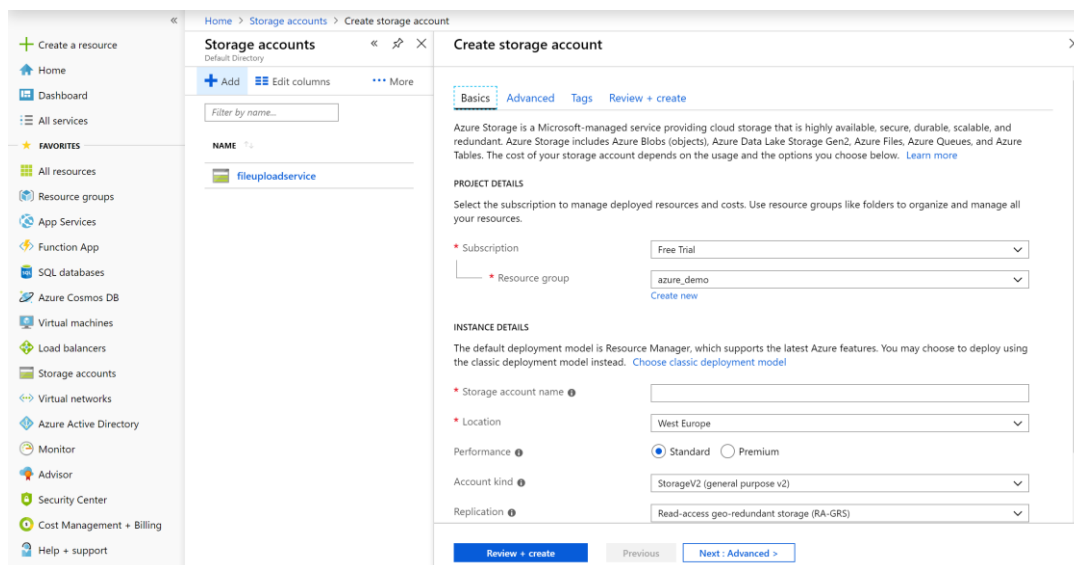
This step assumes the prerequisite that we already have signed up for Azure and have an account set up for some credit or Free Trial subscription.

Create a storage account in Microsoft Azure

Go to Azure, on left menu select Storage Account

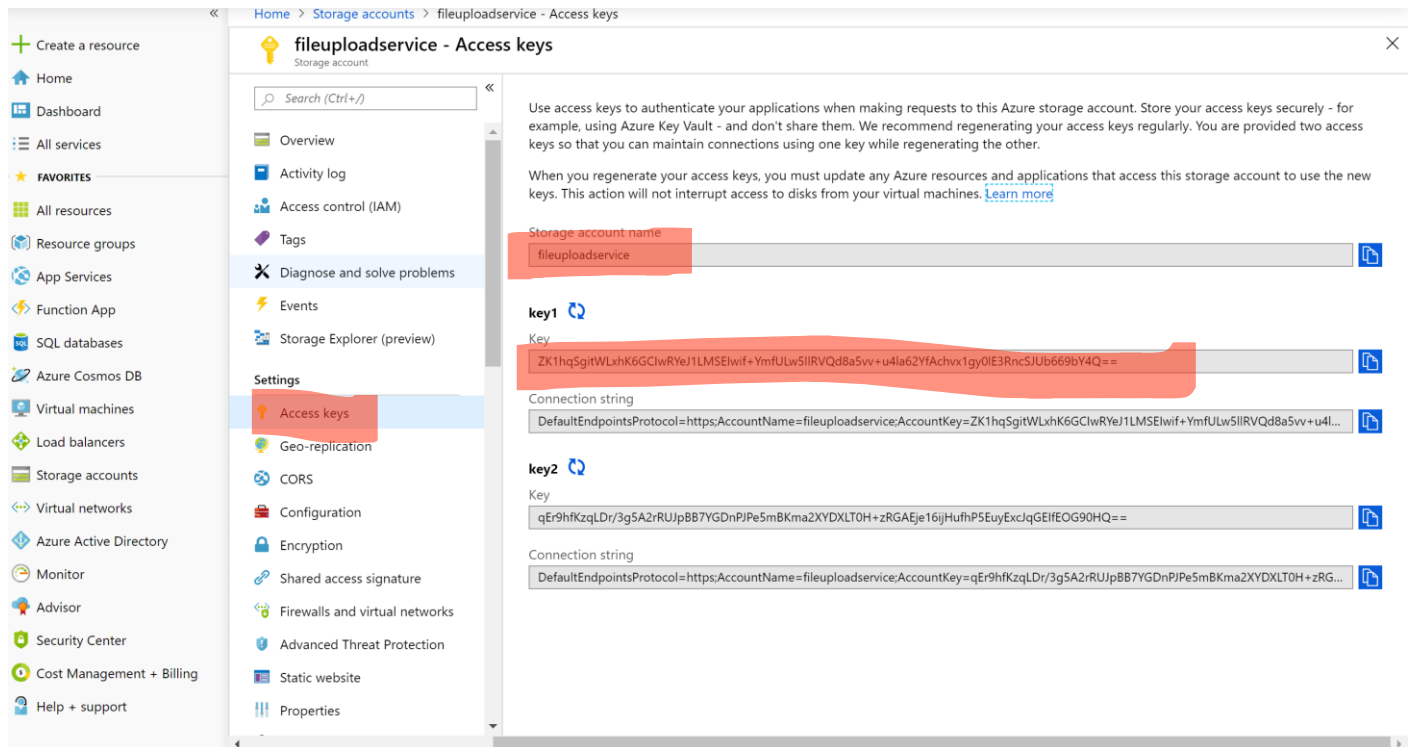


Click the + button on the taskbar so to enter the details of the Storage Account to be created.



Please note that Storage Account name has to be lower case letters and numbers only. This name has to be unique across entire Azure so, we might have to give it several tries before the name gets accepted.

Once the account is created, we have to get the Access Keys so that our Web Api can access the storage account and in turn it's blob storage using those keys



Step 2: Configure the Web Api application to access Azure Storage Account.  
In the web.config, put the Storage account name and Access Key as shown below

```
TableManager.cs  AzureStorageMultipl...StreamProviders.cs  Web.config  UploadController.cs
1 <?xml version="1.0" encoding="utf-8"?>
2 <!--
3 For more information on how to configure your ASP.NET application, please visit
4 http://go.microsoft.com/fwlink/?LinkID=381879
5 -->
6 <configuration>
7 <appSettings>
8 <add key="webpages:Version" value="3.0.0.0" />
9 <add key="webpages:Enabled" value="false" />
10 <add key="ClientValidationEnabled" value="true" />
11 <add key="UnobtrusiveJavaScriptEnabled" value="true" />
12
13 <add key="storage:account:name" value="fileuploadservice" />
14 <add key="storage:account:key" value="{access key here}" />
15 </appSettings>
16
17 <system.web>
18 <compilation debug="true" targetFramework="4.6" />
19 <httpRuntime targetFramework="4.6" />
20 </system.web>
21 <system.webServer>
22 <handlers>
23 <remove name="ExtensionlessUrlHandler-Integrated-4.0" />
24 <remove name="OPTIONSVerbHandler" />
25 <remove name="TRACEVerbHandler" />
26 <add name="ExtensionlessUrlHandler-Integrated-4.0" path="*" verb="*" type="System.Web.Handlers.TransferRequestHandler" preCondition=">
27 </handlers>
28 </system.webServer>
29 <runtime>
30 <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
31 <dependentAssembly>
32 <assemblyIdentity name="Newtonsoft.Json" culture="neutral" publicKeyToken="30ad4fe6b2a6aeed" />
33 <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0" />
34 </dependentAssembly>
35 </assemblyBinding>
36 </runtime>
37 </system.webServer>
38 </configuration>
```

And, that's all we are good to go, we can invoke the API to upload files and create the log for it.

Thank you for reading.