1. In a web browser, navigate to http://portal.azure.com, and if prompted, sign in using the Microsoft account that is associated with your Azure subscription.

2. In the Microsoft Azure portal, in the Hub Menu (on the left edge of the page), click **New**. Then in the **Storage** menu, click **Storage account**.

* + 3. In the **Create storage account** blade, enter the following settings, and then click **Create**: • **Name**: *Enter a unique name for your storage account (and make a note of it!)*
  + • **Deployment model:** Resource manager
  + • **Account kind**: General purpose
  + • **Performance:** Standard
  + • **Replication:** *Select* ***Locally redundant storage (LRS)***
  + • **Secure transfer required**: Disabled
  + • **Subscription**: *Your subscription*
  + • **Resource group**: *Select* ***Create New*** *and enter a name for a new resource group (and make a note of it!)*
  + • **Location:** *Select any available region*
  + • **Pin to dashboard:** *Unselected*
  + 4. At the top of the page, click **Notifications** and verify that deployment has started. Wait until your storage account has been created. This should take a few minutes.
  + 5. In the Hub menu, click **All resources**, and then click your storage account.
  + 6. In the blade for your storage account, click the **Blobs** tile, and note that you don’t have any containers yet.
  + 7. In the blob service blade, click **+Container**, and create a new container with the following properties: • **Name**: bigdata
  + • **Access type**: Private
  + 8. After the **bigdata** blob container has been created, click it and verify that it contains no blobs.
  + 9. In the bigdata blade, click **Properties** and view the **URL** for the blob container, which should be in the form **https://<*your\_account\_name*>.blob.core.windows.net/bigdata**. This is the URL that client applications can use to access your blob container using HTTP protocol.

**Note**: Azure blob storage also supports the WASB protocol, which is specific to Azure storage. Some big data processing technologies use this protocol to work with Azure storage.

10. Return to the blade for your storage account, and under **Settings**, click **Access keys**. Then on the **Access Keys** page, note that two keys have been generated. These are used to authenticate client applications that connect to your storage account.

**Use the Azure Portal to Upload a File to Azure Storage**

The Azure portal includes a rudimentary graphical interface that you can use to work with your Azure storage account. You can use this to transfer files between your local computer and your blob containers, and to browse the data in your storage account.

1. In the blade for your storage account, view the **Overview** page, and then click the **Blobs** tile.

2. Click the **bigdata** container that you created previously, and then click **Upload**.

* 1. 3. In the **Upload blob** blade, browse to the folder where you extracted the lab files for this course, and select **products.txt**. Then verify the following settings and click **Upload**: • **Files**: “products.txt”
  2. • **Blob type**: Block blob
  3. • **Block size**: 100 MB

**Note**: Azure storage supports three blob types (*block*, *page*, and *append*). Block blobs are formed of one or more blocks of data based on the specified block size, and provide an efficient format for uploading and transferring blobs. For more details about blob types in Azure storage, see https://docs.microsoft.com/en-us/rest/api/storageservices/fileservices/Understanding-Block-Blobs--Append-Blobs--and-Page-Blobs.

4. After the blob has been uploaded, note that it is listed in the **bigdata** container blade. Then click the **Products.txt** blob and in the **Blob** properties blade, note its **URL**, which should be similar to **https://<*your\_account\_name*>.blob.core.windows.net/bigdata/products.txt**.

Use Azure Storage Explorer to Upload files to Azure Storage

The blob container interface in the Azure portal enables you to upload, browse, and download blobs; but it lacks many of the features expected in a modern file management tool. There are various graphical Azure storage management tools available, including support for exploring your Azure storage in Microsoft Visual Studio. However, if you do not need the full Visual Studio environment, you can install Azure Storage Explorer, which is available for Windows, Mac OSX, and Linux.

1. Open a new browser tab and browse to http://storageexplorer.com.

2. Download and install the latest version of Azure Storage Explorer for your operating system (Windows, Mac OSX, or Linux).

3. When the application is installed, launch it. Then add your Azure account, signing in with your Azure credentials when prompted, and configure Storage Explorer to show resources from the Azure subscription in which you created your storage account.

4. After your subscription has been added to the **Explorer** pane expand your storage account, expand **Blob Containers**, and select the **bigdata** container. Note that the **products.txt** file you uploaded previously is listed.

5. In the **Upload** drop-down menu, note that you can choose to upload individual files or folders. Then select **Upload Folder** and browse to the folder where you extracted the lab files for this course and select the **data** folder, and upload it as a block blob.

6. After the upload operation is complete, double-click the **data** folder in your blob container to open it, and verify that it contains files named **customers.txt** and **reviews.txt**.

7. Click the button to navigate back up to the root of the **bigdata** container, and select the **products.txt** file. Then click **Copy**.

8. Open the **data** folder, and then click **Paste** to copy the **product.txt** file to this folder.

9. Navigate back up to the root of the **bigdata** container, and select the **products.txt** file. Then click **Delete**, and when prompted to confirm the deletion, click **Yes**.

10. Verify that the **bigdata** container now contains only a folder named **data**, which in turn contains files named **customers.txt**, **products.txt**, and **reviews.txt**.