Tutorial on how to create and manage storage account on Microsoft Azure

Mujtaba Ahmed Abbasi

February 2017

Contents

Requirements

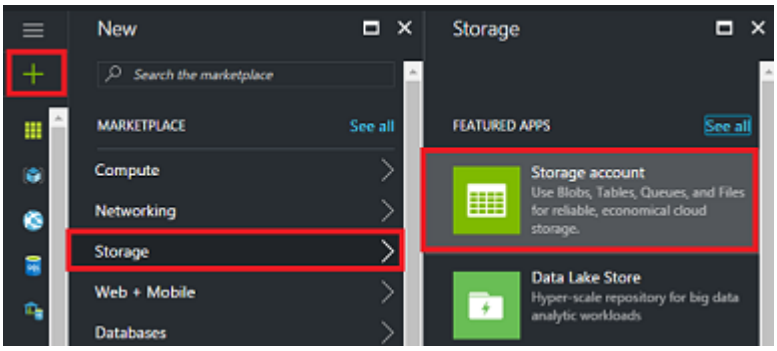
- Sign in for 30 day free trial account from Microsoft Azure subscription.
- Install the Java Development Kit (JDK)
- Text editor, I will be using sublime 3
- Github account

i. Creating an Azure Blob Storage account

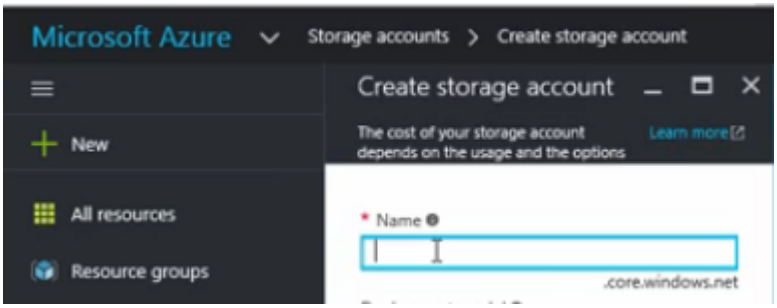
We will be creating blob storage account in this step to store our unstructured data as objects in Azure storage.

1. Sign in to the Azure portal.

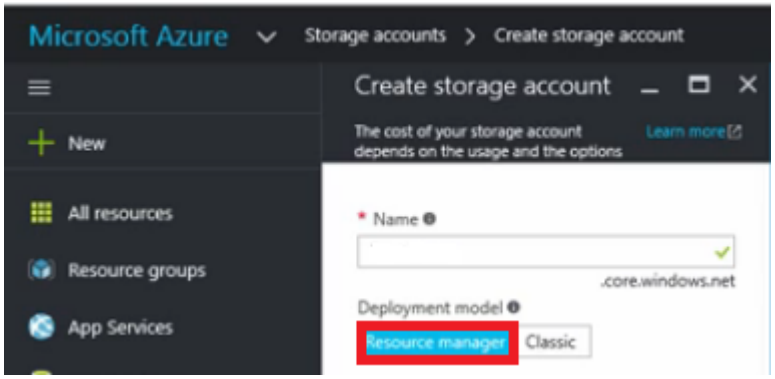
2. On the Hub menu, select **New**, **Storage** and then **Storage account**.



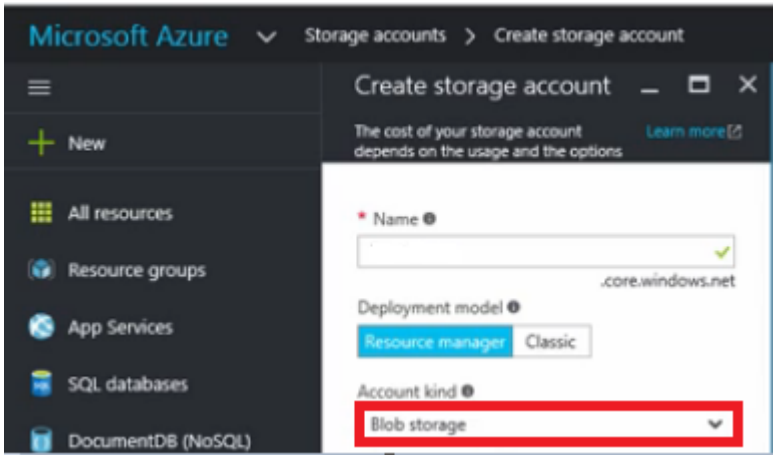
3. Enter a unique name for your storage account. Name must be of 3-24 characters in length and may contain numbers and lowercase letters only.



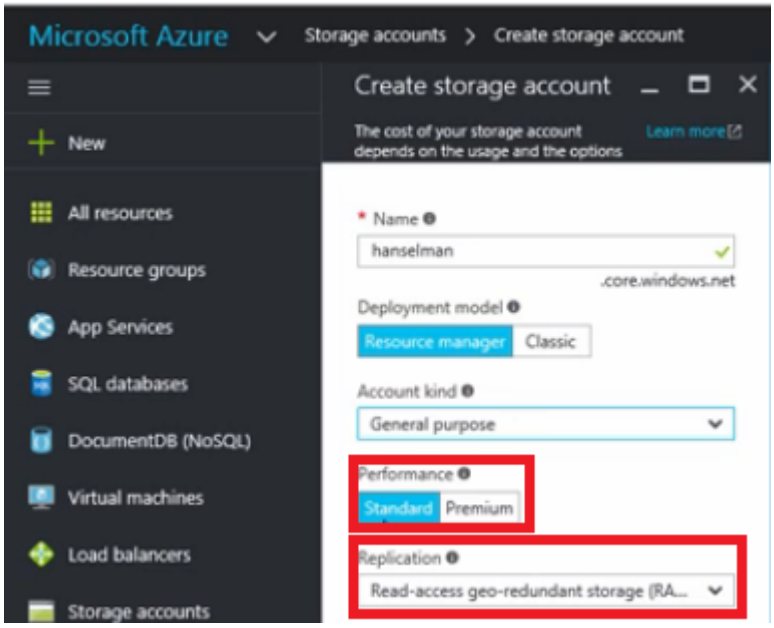
4. Select the deployment model option as **Resource Manager** . Blob storage accounts can only be created using this particular option.



5. Change the type of storage account from General purpose to **Blob Storage**. By default it is General purpose.



6. Now for **Performance** and **Replication** wise we will leverage the default options for this demo.



7. Leave the remaining fields default and hit the pin to dashboard button at the bottom of your screen. Finally click the **Create** button to create the storage account.

ii. Creating a container in that storage account

In this step we will create a container (CloudBlobClient) in the storage account we created in the previous section.

1. Set a connection string to the Azure storage account.

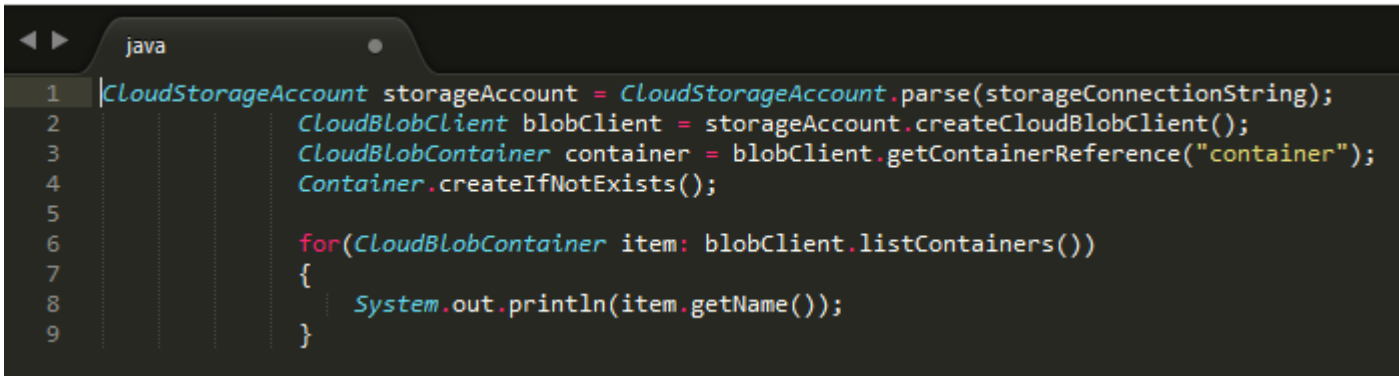
```java
// setting up a connection to Azure storage account
public static final String storageConnectionString =
    "DefaultEndpointsProtocol=http;" +
    "AccountName=your_storage_account_name;" +
    "AccountKey=your_storage_account_key";

```

2. Create a container: Using **CloudBlobClient** we can get reference objects for containers. You can create the container if it doesn't exist with the **createIfNotExists** method, which will otherwise return the existing container.

```java
// Get your storage account from connection-string.
CloudStorageAccount storageAccount = CloudStorageAccount.parse(storageConnection

// Create the blob client.
CloudBlobClient blobClient = storageAccount.createCloudBlobClient();

// Get a reference to a container.
CloudBlobContainer container = blobClient.getContainerReference("mycontainer");

// Create the container if it does not exist.
container.createIfNotExists();
```

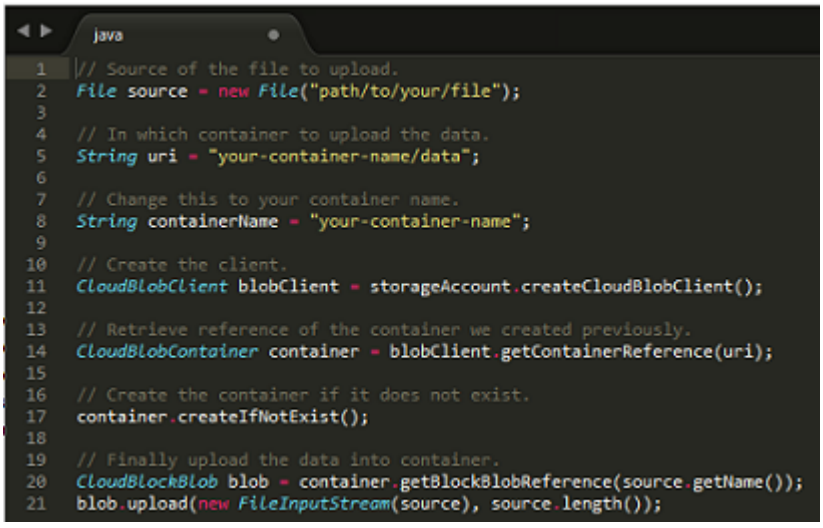
iii. Seeing what containers exist within the storage account

In order to list the containers, we can use **ListContainers()** to list all containers in one storage account, consider the java code:



iv. Uploading data to that container

Now we have created a container, its time to upload data in it. Consider the code:



The above example will upload the data file into the data directory of your container.

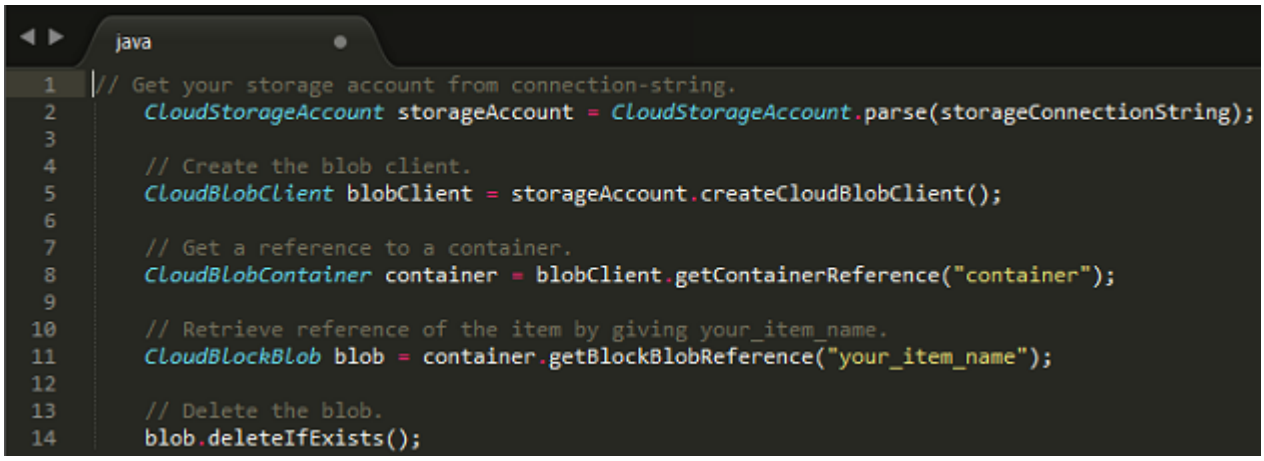
v. Seeing items are inside of that container

To see the blobs in a container, we need to get reference of the container first like we did while uploading a blob. Using **listBlobs** method with a for loop, we can print the uri of each item in a container. Consider the java code:

```java
// Get your storage account from connection-string.
    CloudStorageAccount storageAccount = CloudStorageAccount.parse(storageConnection

    // Create the blob client.
    CloudBlobClient blobClient = storageAccount.createCloudBlobClient();

    // Get a reference to a container.
    CloudBlobContainer container = blobClient.getContainerReference("container");

    // Loop over blobs within the container and output the URI to each of them.
    for (ListBlobItem blobItem : container.listBlobs()) {
        System.out.println(blobItem.getUri());
    }
```

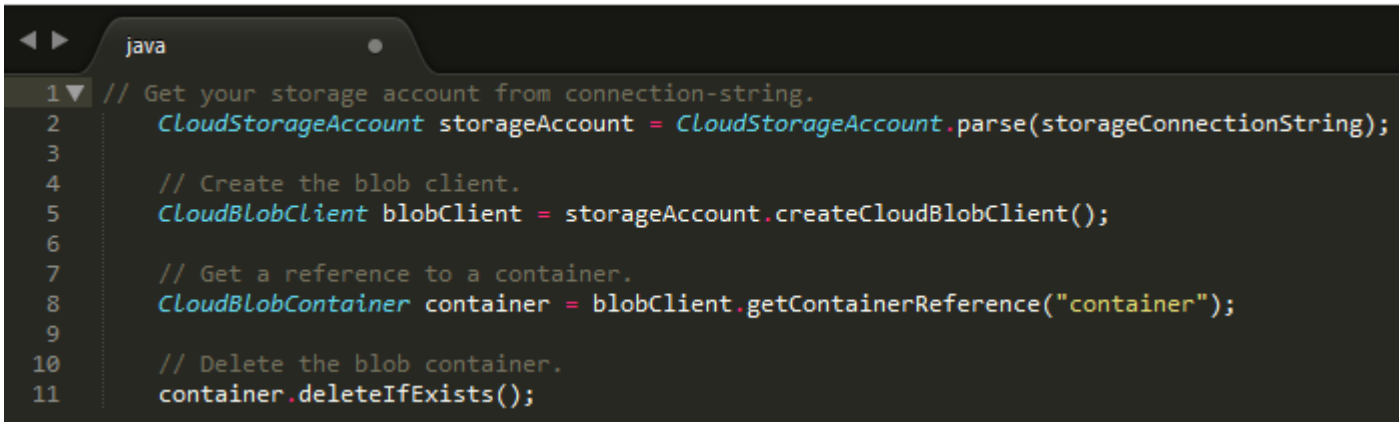
vi. Deleting an item in that container

In order to delete an item, get a reference of that item and call **deleteIfExists**.



vii. Deleting the container

To delete a blob container, consider the following code:



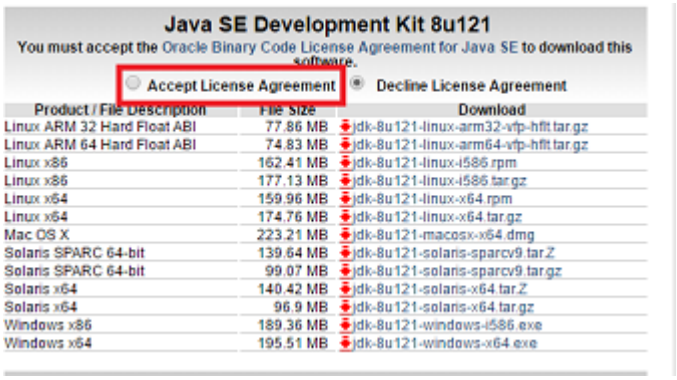
viii. Deleting the storage account

To delete storage account click the **Delete** button on top of your storage account in Azure portal. Deleting a storage account deletes the entire account, including all data in the account.

Installing Java Development Kit (JDK)

1. Go to http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html and select the highlighted box.

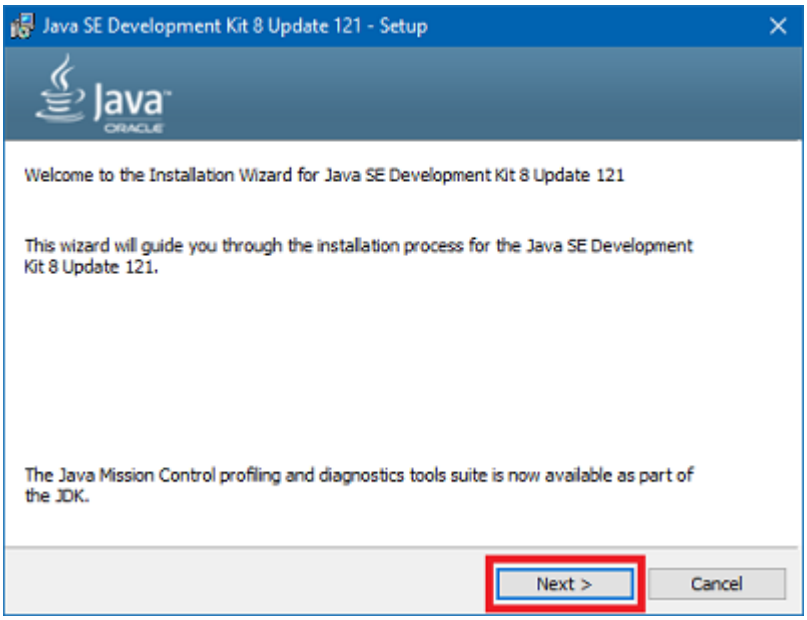
2. Select you preferred option and check the Accept license agreement, i am going for windows x84 depending upon my system specs.



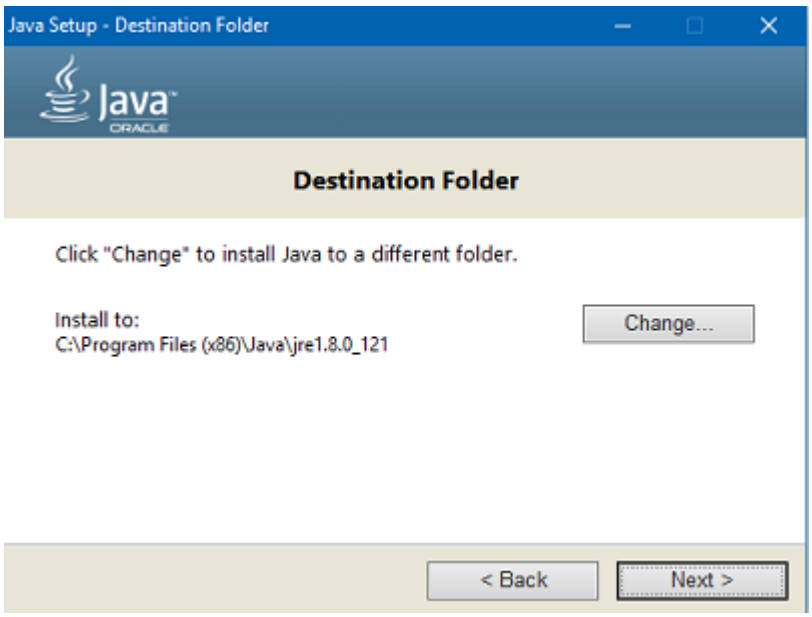
3. Accepting the license agreement and selecting the preferred option, your file will start downloading.

4. Now run the exe file and follow the steps.

5. Click Next



6. Select the folder where you want to save and hit next.



7. It will start installing Java development kit.