Azure CLI sample script

Next, we'll work with a small shell script that issues a few basic Azure CLI commands to interact with Azure Storage resources. The script first creates a new container in your storage account, then uploads an existing file (as a blob) to that container. It then lists all blobs in the container, and finally, downloads the file to a destination on your local computer that you specify.

bash
```bash
#!/bin/bash
# A simple Azure Storage example script

export AZURE_STORAGE_ACCOUNT=<storage_account_name>
export AZURE_STORAGE_KEY=<storage_account_key>

export container_name=<container_name>
export blob_name=<blob_name>
export file_to_upload=<file_to_upload>
export destination_file=<destination_file>

echo "Creating the container..."
az storage container create --name $container_name

echo "Uploading the file..."
az storage blob upload --container-name $container_name --file $file_to_upload --name $blob_name

echo "Listing the blobs..."
az storage blob list --container-name $container_name --output table

echo "Downloading the file..."
az storage blob download --container-name $container_name --name $blob_name --file $destination_file --output table

echo "Done"
```

Configure and run the script

1. Open your favorite text editor, then copy and paste the preceding script into the editor.
2. Next, update the script's variables to reflect your configuration settings. Replace the following values as specified:
    - **\<storage_account_name\>** The name of your storage account.
    - **\<storage_account_key\>** The primary or secondary access key for your storage account.
    - **\<container_name\>** A name the new container to create, such as "azure-cli-sample-container".
    - **\<blob_name\>** A name for the destination blob in the container.
    - **\<file_to_upload\>** The path to small file on your local computer, such as "~/images/HelloWorld.png".
    - **\<destination_file\>** The destination file path, such as "~/downloadedImage.png".
3. After you've updated the necessary variables, save the script and exit your editor. The next steps assume you've named your script **my_storage_sample.sh**.
4. Mark the script as executable, if necessary: `chmod +x my_storage_sample.sh`
5. Execute the script. For example, in Bash: `./my_storage_sample.sh`

You should see output similar to the following, and the **\<destination_file\>** you specified in the script should appear on your local computer.

```
Creating the container...
{
  "created": true
}
Uploading the file...
Percent complete: %100.0
Listing the blobs...
Name       Blob Type      Length   Content Type            Last Modified
---------  -----------    -------- ----------------------  ------------------------
README.md  BlockBlob         6700  application/octet-stream 2017-05-12T20:54:59+00:00
Downloading the file...
Name
---------
README.md
Done
```