Setup IBM Cloud Object Storage to store your Files

by Malika Goyal, Srishti Srivastava

Note: if the interface for Watson Studio has changed, please post a question in the form, we are working on updating this document.

This section will teach you how to setup IBM Object Storage and store your dashboard files hosted on the Cloud in Python. You use IBM Cloud Object Storage, an affordable, reliable, and secure Cloud storage solution.This article will help you get started with IBM Cloud Object Storage.

What is Object Storage and why should you use it?

The "Storage" part of object storage is pretty straightforward, but what exactly is an object and why would you want to store one? An object is basically any conceivable data. It could be a text file, a song, or a picture. For the purposes of this tutorial, our objects will all be HTML files.

Unlike a typical filesystem (like the one used by the device you're reading this article on) where files are grouped in hierarchies of directories/folders, object storage has a flat structure. All objects are stored in groups called buckets. This structure allows for better performance, massive scalability, and cost-effectiveness.

By the end of this article, you will know how to store your files on IBM Cloud Object Storage and easily access them using Python.

Provisioning an Object Storage Instance on IBM Cloud

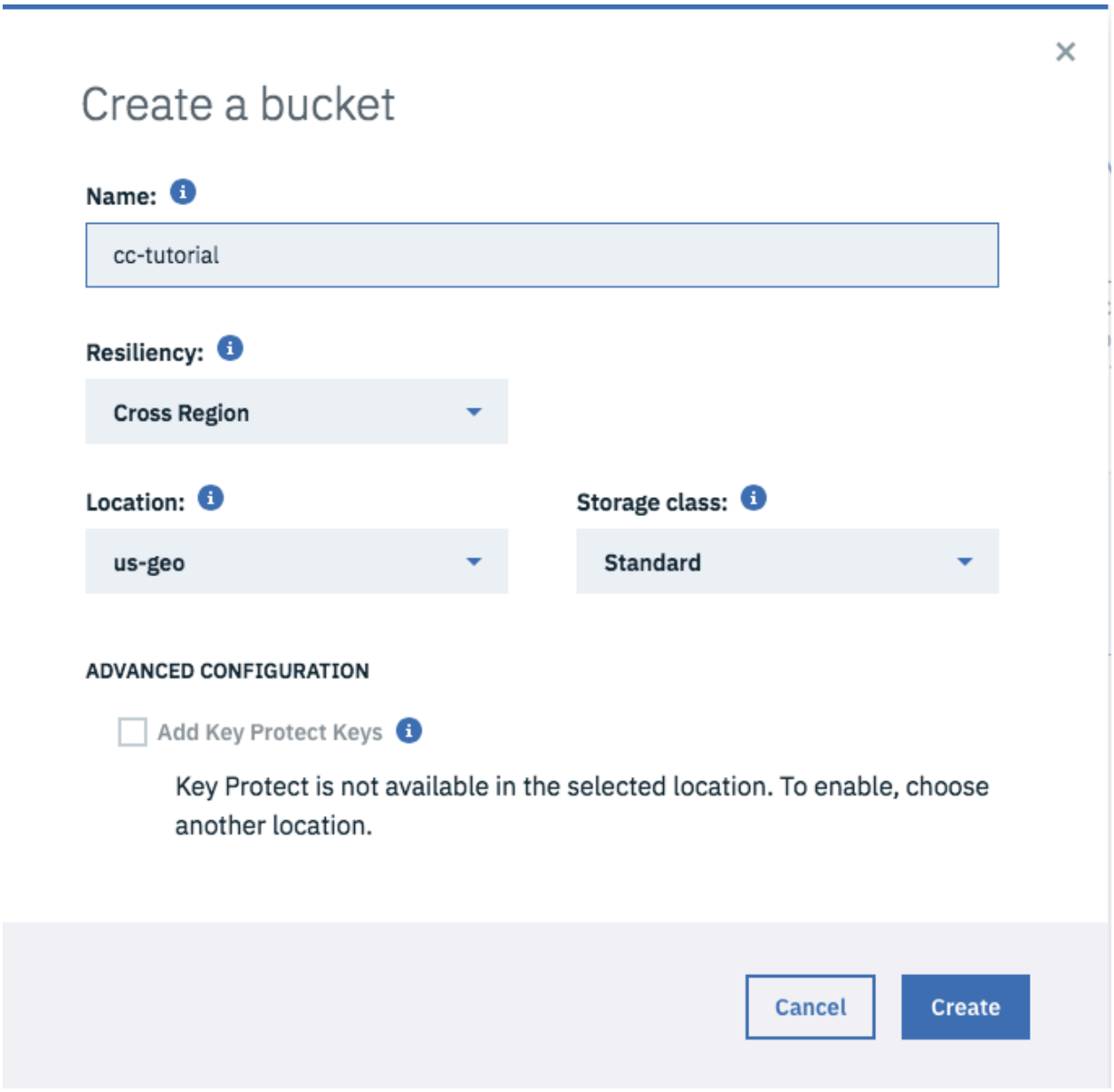
Sign up or log in with your IBM Cloud account here (it's free) to begin provisioning your Object Storage instance. Feel free to use the Lite plan, which is free and allows you to store up to 25 GB per month. You can customize the Service Name if you wish, or just leave it as the default. You can also leave the resource group to the default. Resource groups are useful to organize your resources on IBM Cloud, particularly when you have many of them running. When you're ready, click the **Create** button to finish provisioning your Object Storage instance.

Working with Buckets

Since you just created the instance, you'll now be presented with options to create a bucket. You can always find your Object Storage instance by selecting it from your IBM Cloud Dashboard. There's a limit of 100 buckets per Object Storage instance, but each bucket can hold billions of objects. In practice, how many buckets you need will be dictated by your availability and resilience needs. For the purposes of this tutorial, a single bucket will do just fine.

Creating your First Bucket

Click the **Create Bucket** button and you'll be shown a window like the one below, where you can customize some details of your Bucket. All these options may seem overwhelming at the moment, but don't worry, we'll explain them in a moment. They are part of what makes this service so customizable, should you have the need later on. There are several naming conventions including Must start and end in alphanumeric characters (from 3 to 255) limited to: lowercase, numbers and non-consecutive dots, and hyphens. For now all you have to do is make sure that **Resiliency** is set to **Cross Region** and the **location** is set to **us-geo** .



If you don't care about the nuances of bucket configuration, you can type in any unique name you like and press the Create button, leaving all other options to their defaults. You can then skip to the **Putting Objects in Buckets** section below. If you would like to learn about what these options mean, read on.

Configuring your bucket

Resiliency Options

Resiliency Option	Description	Characteristics
Cross Region	Your data is stored across three geographic regions within your selected location	High availability and *very* high durability
Regional	Your data is stored across three different data centers within a single geographic region	High availability and durability, very low latency for regional users
Single Data Center	Your data is stored across multiple devices within a single data center	Data locality

Storage Class Options

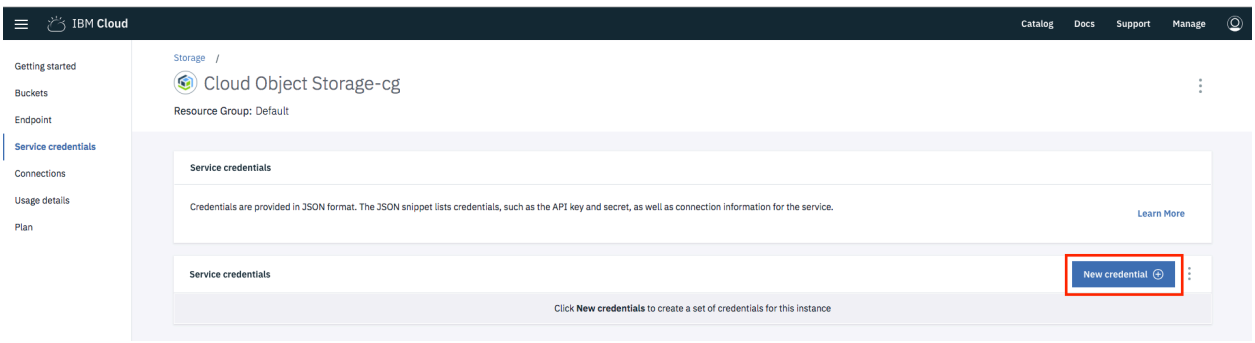
Frequency of Data Access	IBM Cloud Object Storage Class
Continual	**Standard**
Weekly or monthly	**Vault**
Less than once a month	**Cold Vault**
Unpredictable	**Flex**

Feel free to experiment with different configurations, but I recommend choosing "Standard" for your storage class for this tutorial's purposes. Any resilience option will do.

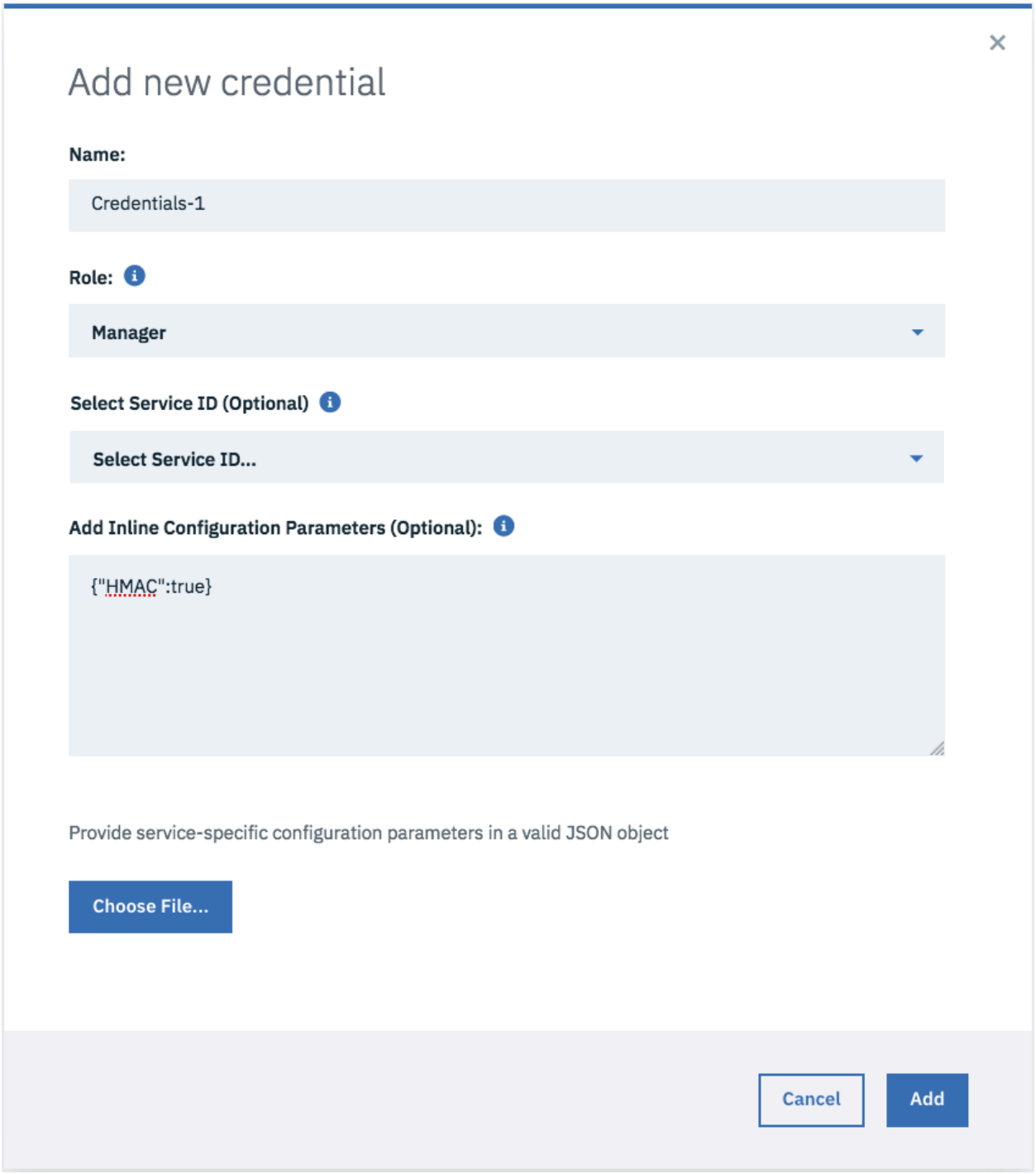
After you've created your bucket,later you will have to store the name of the bucket into the Python variable `bucket_name` (replace `cc-tutorial` with the name of your bucket) in your Jupyter notebook.

Creating Service Credentials

To access your IBM Cloud Object Storage instance from anywhere other than the web interface, you will need to create credentials. Click the **New credential** button under the **Service credentials** section to get started.



In the next window, select `Manager` as your role, and add `{"HMAC":true}` to the `Add Inline Configuration Parameters (Optional)` field. You can leave all other fields as their defaults and click the **Add** button to continue.



You'll now be able to click on **View credentials** to obtain the JSON object containing the credentials you just created. You'll want to store everything you see in

a `credentials` variable like the one below (obviously, replace the placeholder values with your own). Take special note of your `access_key_id` and `secret_access_key` which you will need for the **Python** section below

Note: If you're following along within a notebook be careful not to share this notebook after adding your credentials!

Putting Objects in Buckets

Adding files to IBM Cloud Object Storage with Python

In the assignment, much of the code will be provided. When you call the function make_dashboard the function will create and save a file `index.html`; This file contains the dashboard, you must place this file in IBM object storage, you can then view the dashboard via a `URL`. You will need the variable `file_name`.

file_name= "index.html"

There is option to view the credentials you have created. Click **â€œView Credentialsâ€□**, you will get credentials in the format given below. Copy your Credentials and assign them to the variable `credentials` given in the notebook in **Question 5 #Hidden cell**. Format for credentials as follow:

```
credentials = {
  "apikey": "your-api-key",
  "cos_hmac_keys": {
  "access_key_id": "your-access-key-here",
  "secret_access_key": "your-secret-access-key-here"
  },
 "endpoints": "your-endpoints",
  "iam_apikey_description": "your-iam_apikey_description",
  "iam_apikey_name": "your-iam_apikey_name",
  "iam_role_crn": "your-iam_apikey_name",
  "iam_serviceid_crn": "your-iam_serviceid_crn",
 "resource_instance_id": "your-resource_instance_id"
}
```

There will be several other steps, but don't worry all the instructions will be provided in the lab.

Note: 1) While creating a bucket, make sure **Resiliency** is set to **Cross Region** and the **location** is set to **us-geo**.

2) Bucket name should be same as that you have created in the Object Storage.

3) endpoint url given in the notebook should remain same.