**Assignment -3**

Cloud object storage and Watson Assistant

|  |  |
| --- | --- |
| Assignment Date | 19 September 2022 |
| Student Name | DEVANAND V |
| Student Roll Number | 7376191CS151 |
| Maximum Marks | 2 Marks |

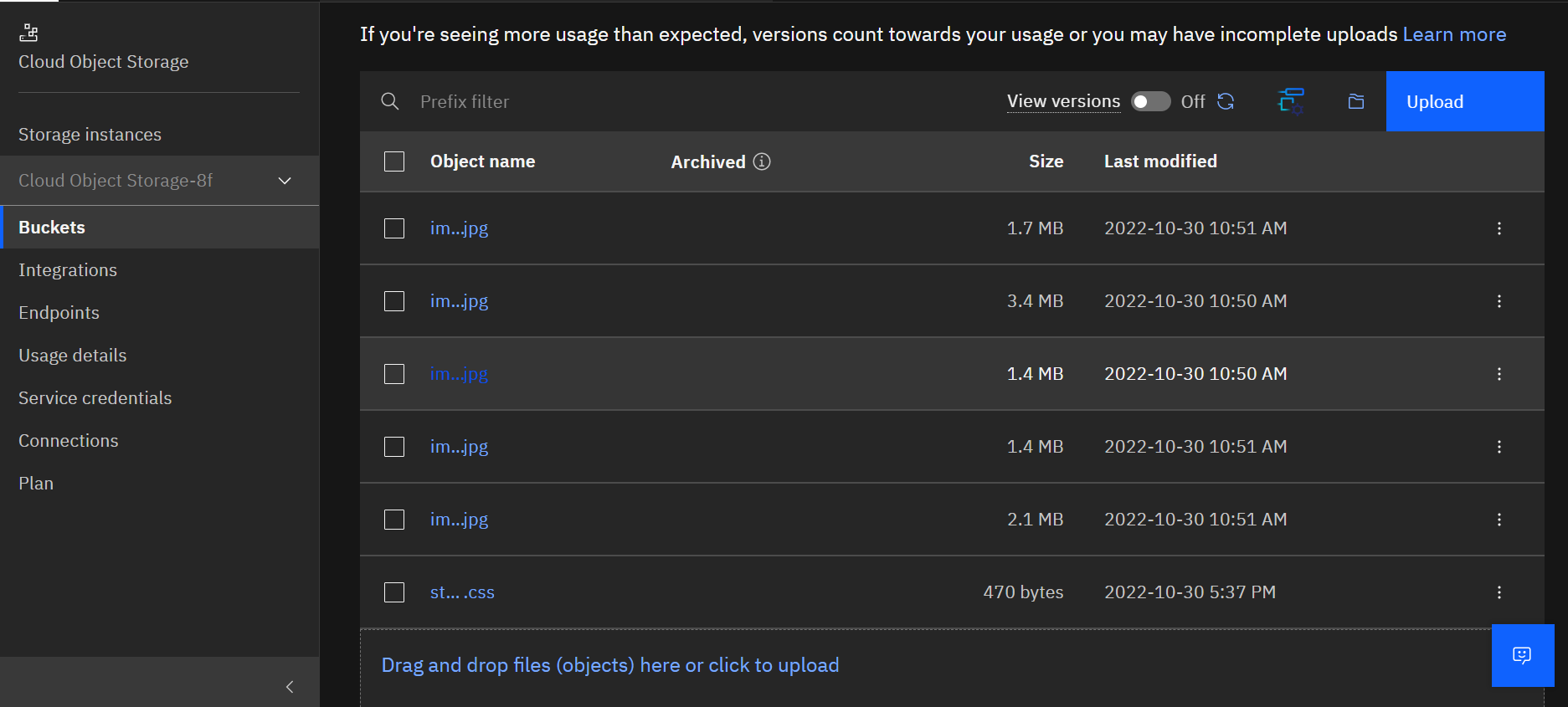
**Question-1:**

1.Create a Bucket in IBM object storage.

2.Upload an 5 images to ibm object storage and make it public. write html code to displaying all the 5 images.

3.Upload a css page to the object storage and use the same page in your HTML code.

**Bucket image:**



**object\_config.py**

import ibm\_boto3

from ibm\_botocore.client import Config, ClientError

# Constants for IBM COS values

COS\_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud" # Current list avaiable at https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints

COS\_API\_KEY\_ID = "Bc32PN-Ec2mpFRVtchtdfkBJBM-uDrjwYdY12L0knCH4" # eg "W00YixxxxxxxxxxMB-odB-2ySfTrFBIQQWanc--P3byk"

COS\_INSTANCE\_CRN = "crn:v1:bluemix:public:cloud-object-storage:global:a/fc99842817e447e4be38d2cca7268d9c:183cba6d-ad4f-4d17-b129-61f93943e367::"

# Create resource

cos = ibm\_boto3.resource("s3",

    ibm\_api\_key\_id=COS\_API\_KEY\_ID,

    ibm\_service\_instance\_id=COS\_INSTANCE\_CRN,

    config=Config(signature\_version="oauth"),

    endpoint\_url=COS\_ENDPOINT

)

def get\_buckets():

    print("Retrieving list of buckets")

    try:

        buckets = cos.buckets.all()

        for bucket in buckets:

            # print("Bucket Name: {0}".format(bucket.name))

            return bucket.name

    except ClientError as be:

        print("CLIENT ERROR: {0}\n".format(be))

    except Exception as e:

        print("Unable to retrieve list buckets: {0}".format(e))

bucket\_name = get\_buckets()

print(bucket\_name)

def get\_bucket\_contents(bucket\_name):

    print("Retrieving bucket contents from: {0}".format(bucket\_name))

    try:

        file\_names = []

        files = cos.Bucket(bucket\_name).objects.all()

        for file in files:

            # print("Item: {0} ({1} bytes).".format(file.key, file.size))

            if 'img' in file.key:

                file\_names.append(file.key)

        return file\_names

    except ClientError as be:

        print("CLIENT ERROR: {0}\n".format(be))

    except Exception as e:

        print("Unable to retrieve bucket contents: {0}".format(e))

object\_files = get\_bucket\_contents(bucket\_name)

**app.py:**

from flask import Flask, render\_template, redirect

from object\_config import object\_files

app = Flask(\_\_name\_\_)

@app.route('/')

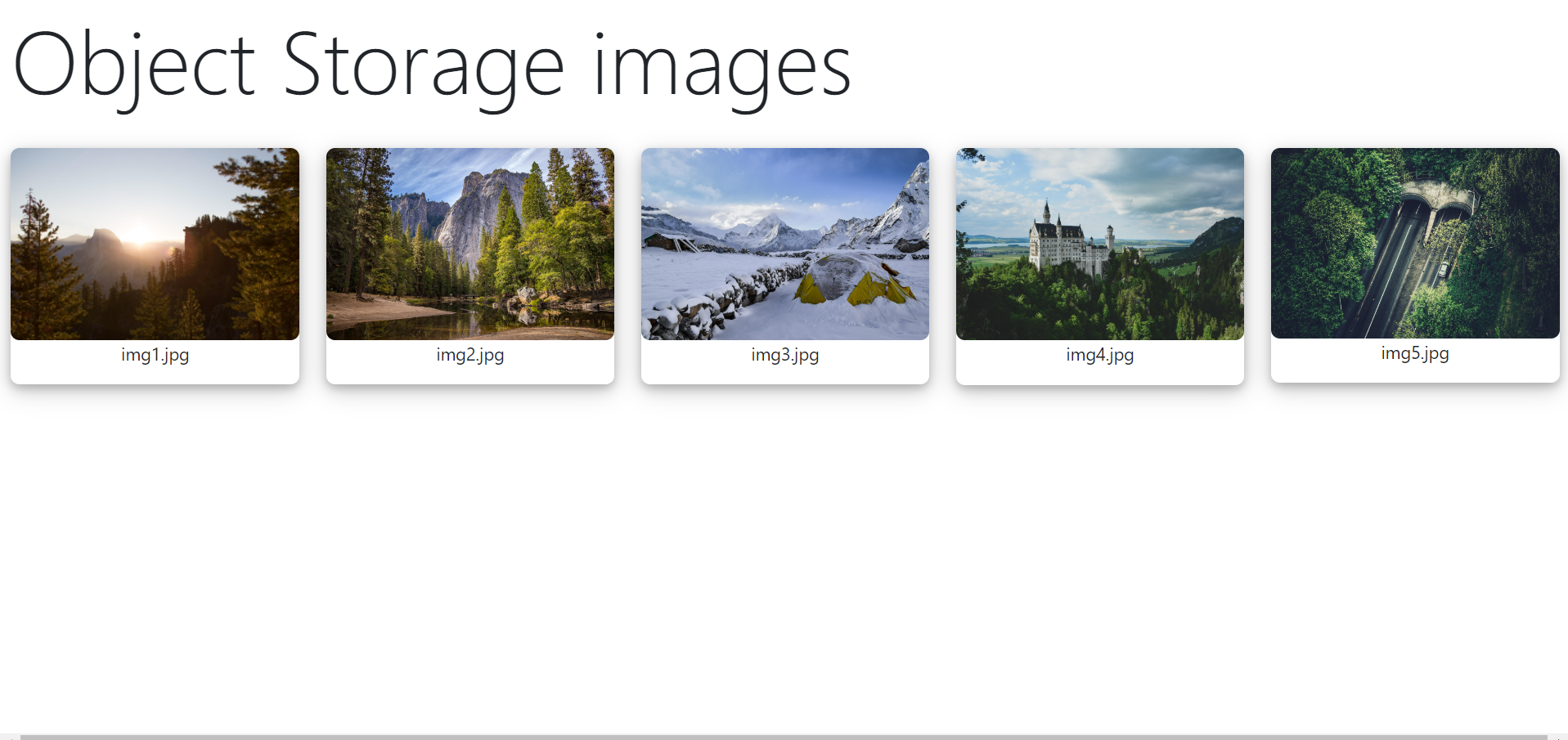
def hello():

    return render\_template('index.html', files = object\_files)

if \_\_name\_\_=='\_\_main\_\_':

    app.run()

**Final output:**

****

**Question-2:**

With a given integral number n, write a program to generate a dictionary that contains (i, i\*i) such that is an integral number between 1 and n (both included). and then the program should print the dictionary.

Suppose the following input is supplied to the program:

8

Then, the output should be:

{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}

|  |
| --- |
| **Solution:** |
|  | n=int(input()) |
|  | d=dict() |
|  | for i in range(1,n+1): |
|  | d[i]=i\*i |
|  |  |
|  | print d |
|  | #----------------------------------------# |
|  | #----------------------------------------# |

Graphical user interface, application, Teams

Description automatically generated