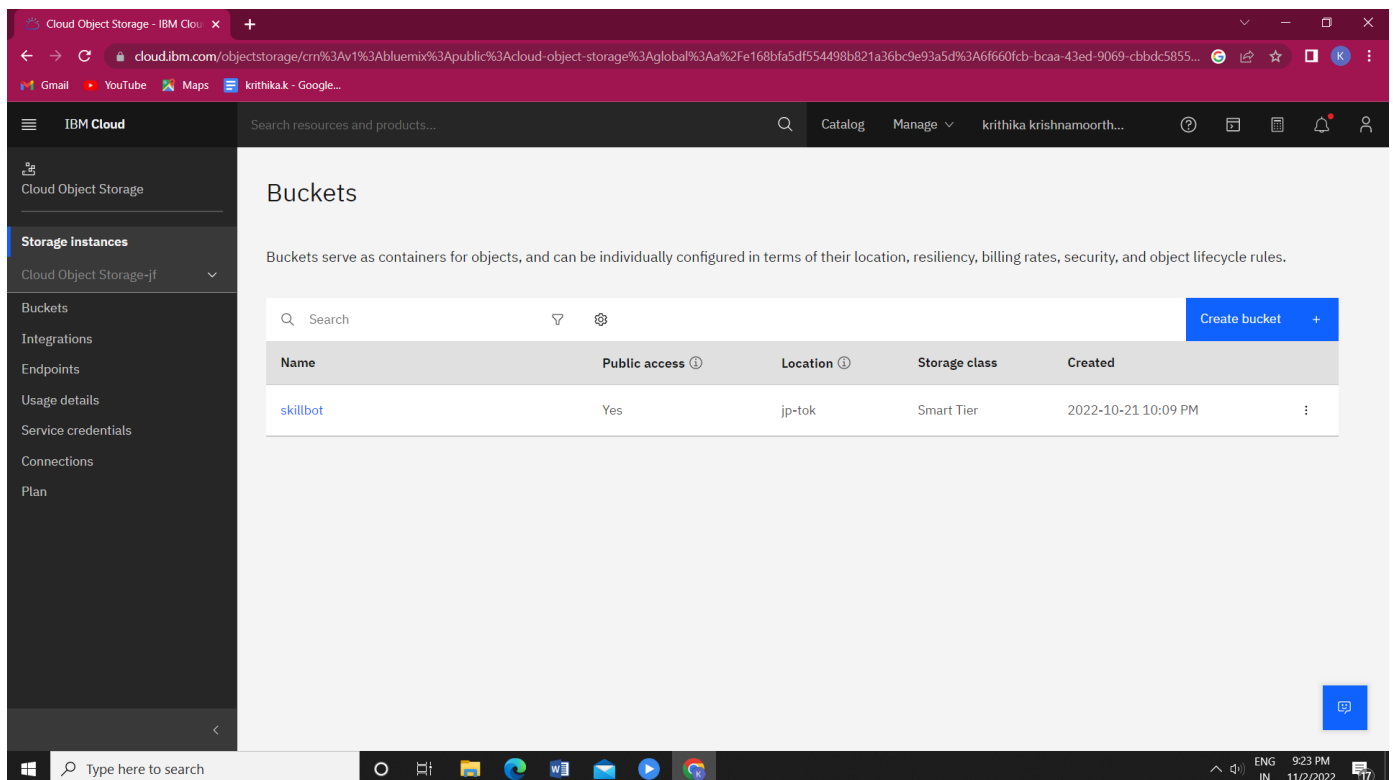


## Assignment-3

|              |                                       |
|--------------|---------------------------------------|
| Date         | 10 October 2022                       |
| Team ID      | PNT2022TMID305715                     |
| Project Name | Skill and Job Recommender Application |

### 1. CREATE A BUCKET IN IBM OBJECT STORAGE.



**Upload an 5 images to ibm object storage and make it public.  
Write html code todisplaying all the 5 images.**

Cloud Object Storage - IBM Cloud

cloud.ibm.com/objectstorage/cm%3Av1%3Abluemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe168bfa5df554498b821a36bc9e93a5d%3A6f660fcb-bcaa-43ed-9069-cbbdc5855...

GmailYouTubeMaps

krithika.k - Google...

IBM Cloud

Search resources and products...

CatalogManagekrithika.krishnamoorth...

Cloud Object Storage

Storage instances

Cloud Object Storage-jf

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

skillbot

TransfersDetailsActions...

ObjectsConfigurationPermissions

Warning: All objects in this bucket have public view access.

If you're seeing more usage than expected, versions count towards your usage or you may have incomplete uploads [Learn more](#)

Prefix filter

Upload

| Object name  | Archived | Size     | Last modified       |
|--------------|----------|----------|---------------------|
| ima...jfif   |          | 13.0 KB  | 2022-10-21 10:15 PM |
| nat...jfif   |          | 7.4 KB   | 2022-10-21 10:14 PM |
| nat...L.jfif |          | 5.0 KB   | 2022-10-21 10:15 PM |
| nat...L.jpg  |          | 181.0 KB | 2022-10-21 10:15 PM |
| nat...3.jfif |          | 7.1 KB   | 2022-10-21 10:15 PM |

Type here to search

ENG 9:24 PM IN 11/2/2022

Cloud Object Storage - IBM Cloud

cloud.ibm.com/objectstorage/cm%3Av1%3Abluemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe168bfa5df554498b821a36bc9e93a5d%3A6f660fcb-bcaa-43ed-9069-cbbdc5855...

GmailYouTubeMaps

krithika.k - Google...

IBM Cloud

Search resources and products...

CatalogManagekrithika.krishnamoorth...

Cloud Object Storage

Storage instances

Cloud Object Storage-jf

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

Manage access to this bucket by creating IAM policies for users and service IDs. Users and service IDs must also have an instance level viewer role (or higher) to use the console or to list buckets using the REST API.

Access policies

Public access

Warning: Access policy update

Access group policy created

A new access policy for this bucket was created for the group: Public Access  
To delete/edit go to the [IAM console](#).

Status: Enabled

Role for this policy: Public Access

Content Reader

As a Content Reader, one can read and list objects in the bucket.

Create access policy

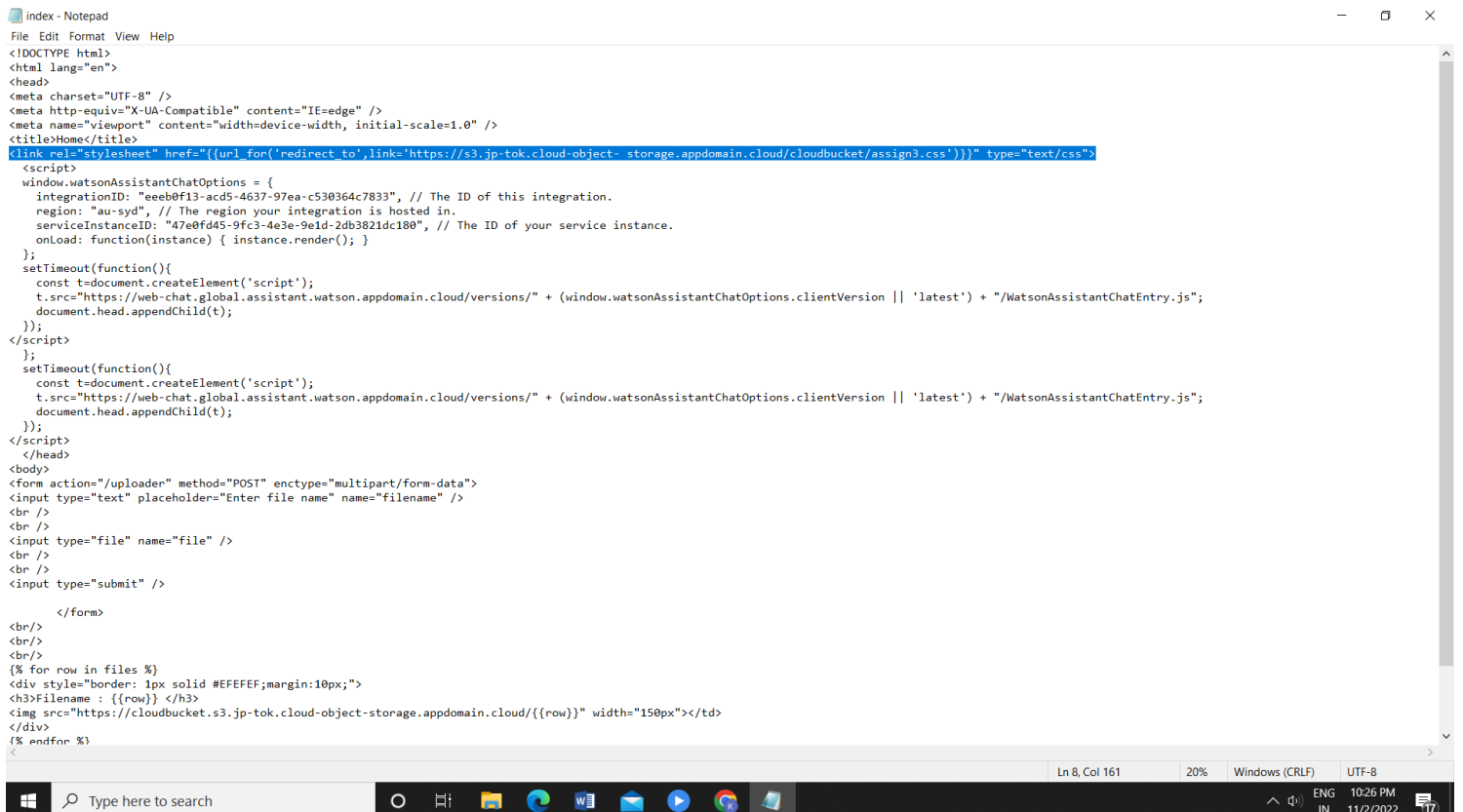
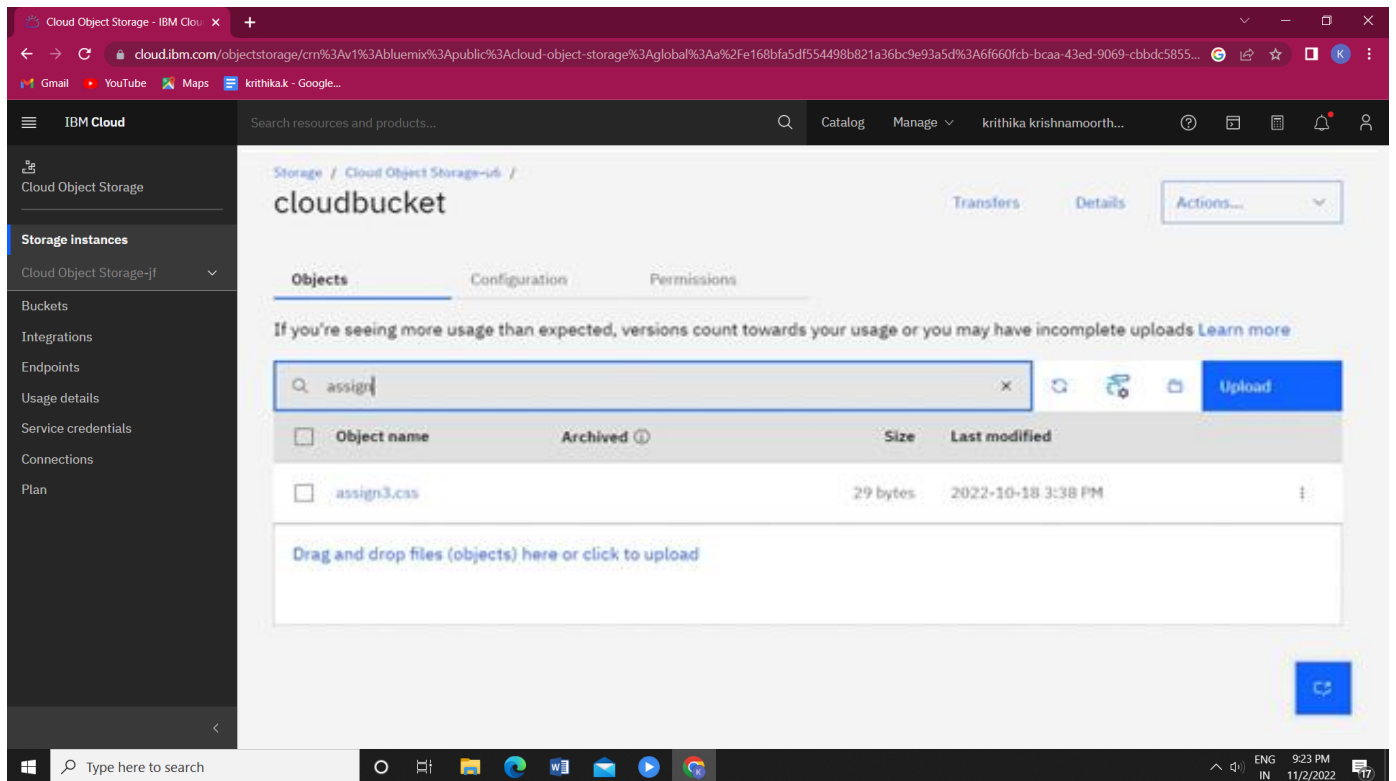
Context-based restrictions

Firewall (legacy)

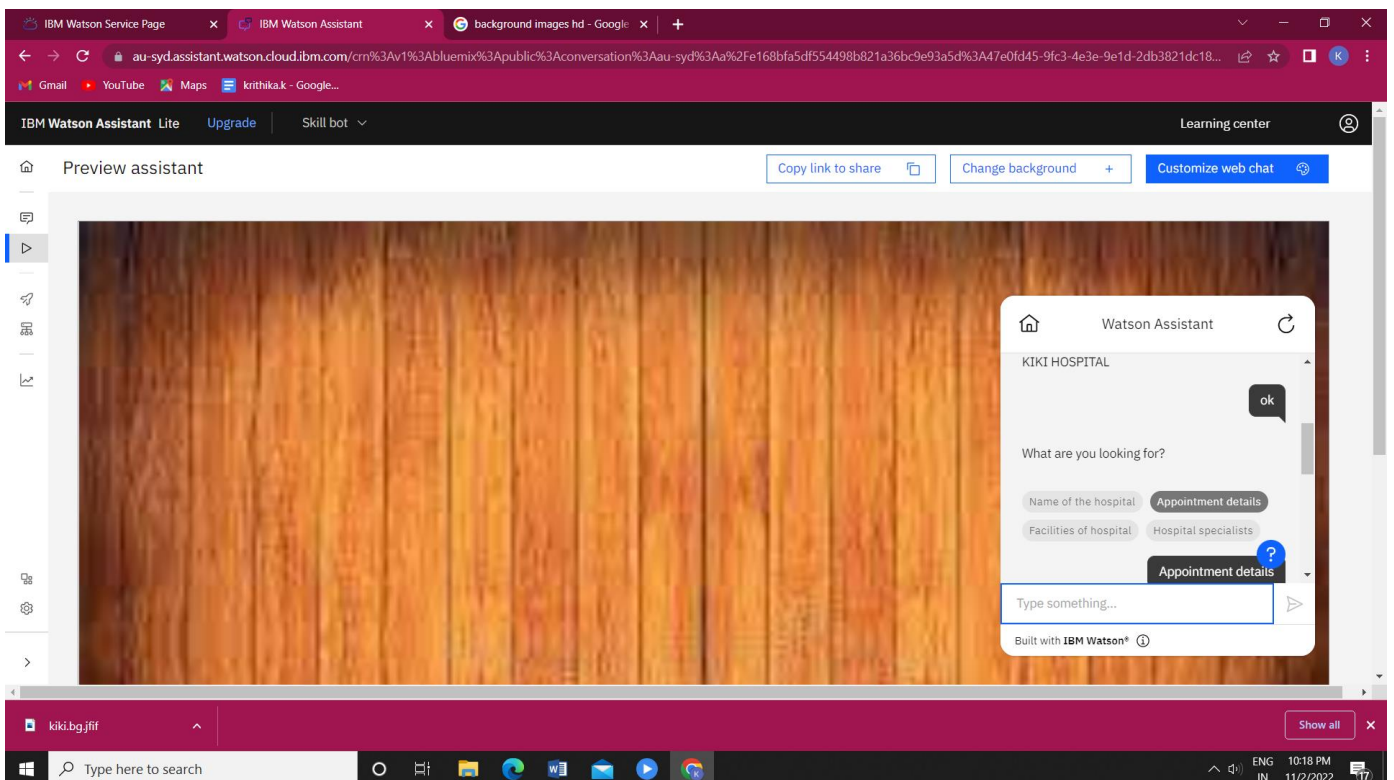
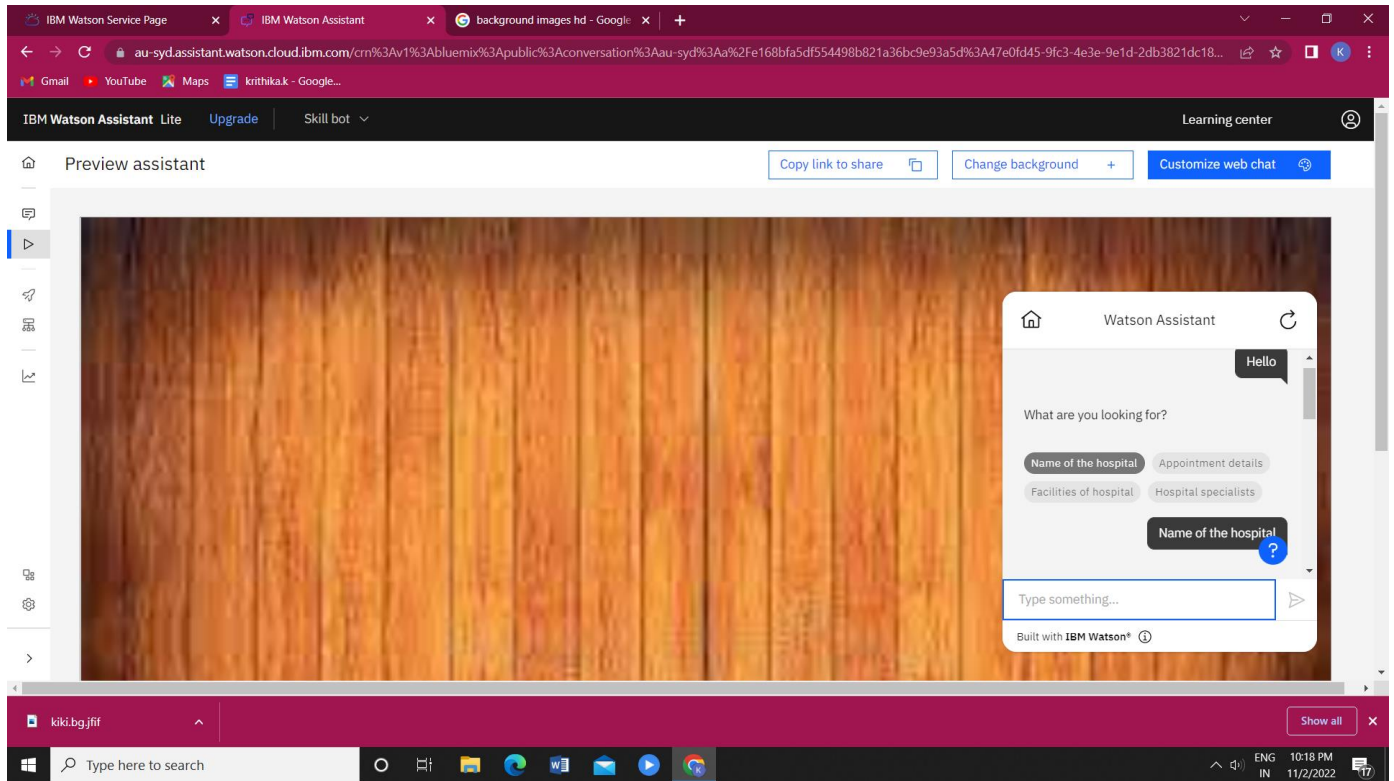
Type here to search

ENG 9:24 PM IN 11/2/2022

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



### 3. Design a chatbot using IBM Watson assistant for hospital.



### Web URL for Assistant:

<https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Fau-syd.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-47e0fd45-9fc3-4e3e-9e1d-2db3821dc180%3A%3A086906ae-3a86-4691-9bae-3ce8e98fc0f5&integrationID=eeeb0f13-acd5-4637-97ea-c530364c7833&region=au-syd&serviceInstanceID=47e0fd45-9fc3-4e3e-9e1d-2db3821dc180>

### 4. Create Watson assistant service with 10 steps and use 3 conditions in it. Load that script in HTML page.

The screenshot displays the IBM Watson Assistant console interface. On the left, a list of steps is visible, with step 10 highlighted. The main area shows the configuration for Step 10, which is titled "Step 10 is taken" and includes a "with conditions" dropdown. The conditions section shows a single condition: "If All of this is true: 9. Other than appoi... is Helpline number". Below the conditions, there is a "New condition group +" button. The "Assistant says" section contains a text input field with the value "999999999". A "Preview" button is located at the bottom right of the configuration area. The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 10:27 PM on 11/2/2022.

## Included 3 conditions in steps:

The screenshot displays the IBM Watson Assistant configuration interface. On the left, the 'Conversation steps' panel shows a sequence of steps. Step 1 is 'What are you looking for?' with a 'Continue to next step' button. Step 2 is 'Name of the hospital' with a 'Free text' input field and a 'Re-ask previous step(s)' button. Step 3 is 'Appointment details' with a 'Free text' input field and a 'Re-ask previous step(s)' button. The main area shows 'Step 2 is taken' with a dropdown menu set to 'with conditions'. Below this, the 'Conditions' section shows a single condition: 'If All of this is true: 1. What are you looking for is Name of the hospital'. The 'Assistant says' section shows the response 'KIKI HOSPITAL'. A 'Preview' button is visible at the bottom right.

The screenshot displays the IBM Watson Assistant configuration interface. On the left, the 'Conversation steps' panel shows a sequence of steps. Step 1 is 'Appointment details' with a 'Free text' input field and a 'Re-ask previous step(s)' button. Step 2 is 'Facilities of hospital' with a 'Free text' input field and a 'Re-ask previous step(s)' button. Step 3 is 'Hospital specialists' with a 'Free text' input field and a 'Continue to next step' button. The main area shows 'Step 4 is taken' with a dropdown menu set to 'with conditions'. Below this, the 'Conditions' section shows a single condition: 'If All of this is true: 1. What are you looking for is Facilities of hospital'. The 'Assistant says' section shows the response 'We provide you with good and healthy environment including patient clothes, beds and all other needs.' A 'Preview' button is visible at the bottom right.

IBM Watson Assistant

au-sydassistant.watson.cloud.ibm.com/cm%3Av1%3Abluemix%3Apublic%3Aconversation%3Aau-syd%3Aa%2Fe168bfa5df554498b821a36bc9e93a5d%3A47e0fd45-9fc3-4e3e-9e1d-2db3821dc18...

IBM Watson Assistant Lite Upgrade Skill bot Learning center

Hello

Continue to next step

Are you looking for an appointment?

6 Confirmation

Continue to next step

6 is Yes

which specialist do you prefer?

7 Radiology Cardiology + 2

Continue to next step

7 is Others

Please enter your preferred specialist

8 Free text

Continue to next step

New step +

Step 7 is taken with conditions f<sub>x</sub>

Conditions 1 condition

If All of this is true:

6. Are you looking f... is Yes

and Add condition +

New condition group +

Assistant says

which specialist do you prefer?

Preview

kiki.bg.jfif

Type here to search

10:28 PM 11/2/2022

## Index.html

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Home</title>

    <link rel="stylesheet" href="{ {url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/cloudbucket/assign3.css')} }" type="text/css">

    <script>

      window.watsonAssistantChatOptions = {

        integrationID: "eeeb0f13-acd5-4637-97ea-c530364c7833", // The ID of this integration.

        region: "au-syd", // The region your integration is hosted in.

        serviceInstanceID: "47e0fd45-9fc3-4e3e-9e1d-2db3821dc180", // The ID of your service instance.

        onLoad: function(instance) { instance.render(); }

      };

      setTimeout(function(){

        const t=document.createElement('script');

        t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" +
        (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";

        document.head.appendChild(t);

      });

    </script>

  </head>

  <body>

    <form action="/uploader" method="POST" enctype="multipart/form-data">

      <input type="text" placeholder="Enter file name" name="filename" />

      <br />

      <br />

      <input type="file" name="file" />

      <br />

      <br />

      <input type="submit" />

    </form>

  </body>

</html>
```



```

</form>

<br/>

<br/>

<br/>

{ % for row in files % }

    <div style="border: 1px solid #EFEFEF;margin:10px;">
        <h3>Filename : {{row}} </h3>
        </td>
    </div>

{ % endfor % }

</body>

</html>

```

## App.py

```

import io

from flask import Flask,redirect,url_for,render_template,request

import ibm_boto3

from ibm_botocore.client import Config, ClientError

COS_ENDPOINT="https://s3.jp-tok.cloud-object-storage.appdomain.cloud"

COS_API_KEY_ID=""

COS_INSTANCE_CRN=""


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
)

```

```

app=Flask(__name__)


@app.route('/')
def index():
    try:
        files = cos.Bucket('cloudbucket').objects.all()
        files_names = []
        for file in files:
            files_names.append(file.key)
            print(file)
            print("Item: {0} ({1} bytes)".format(file.key, file.size))
        return render_template('index.html',files=files_names)

    except ClientError as be:
        print("CLIENT ERROR: {0}\n".format(be))
        return render_template('index.html')
    except Exception as e:
        print("Unable to retrieve bucket contents: {0}".format(e))
        return render_template('index.html')


@app.route('/uploader',methods=['POST'])
def upload():
    name_file=request.form['filename']
    f = request.files['file']
    try:
        part_size = 1024 * 1024 * 5

        file_threshold = 1024 * 1024 * 15

        transfer_config = ibm_boto3.s3.transfer.TransferConfig(
            multipart_threshold=file_threshold,

```

```
        multipart_chunksize=part_size
    )

    content = f.read()
    cos.Object('cloudbucket', name_file).upload_fileobj(
        Fileobj=io.BytesIO(content),
        Config=transfer_config
    )
    return redirect(url_for('index'))
```

```
except ClientError as be:
    print("CLIENT ERROR: {0}\n".format(be))
    return redirect(url_for('index'))
```

```
except Exception as e:
    print("Unable to complete multi-part upload: {0}".format(e))
    return redirect(url_for('index'))
```

```
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080, debug=True)
```