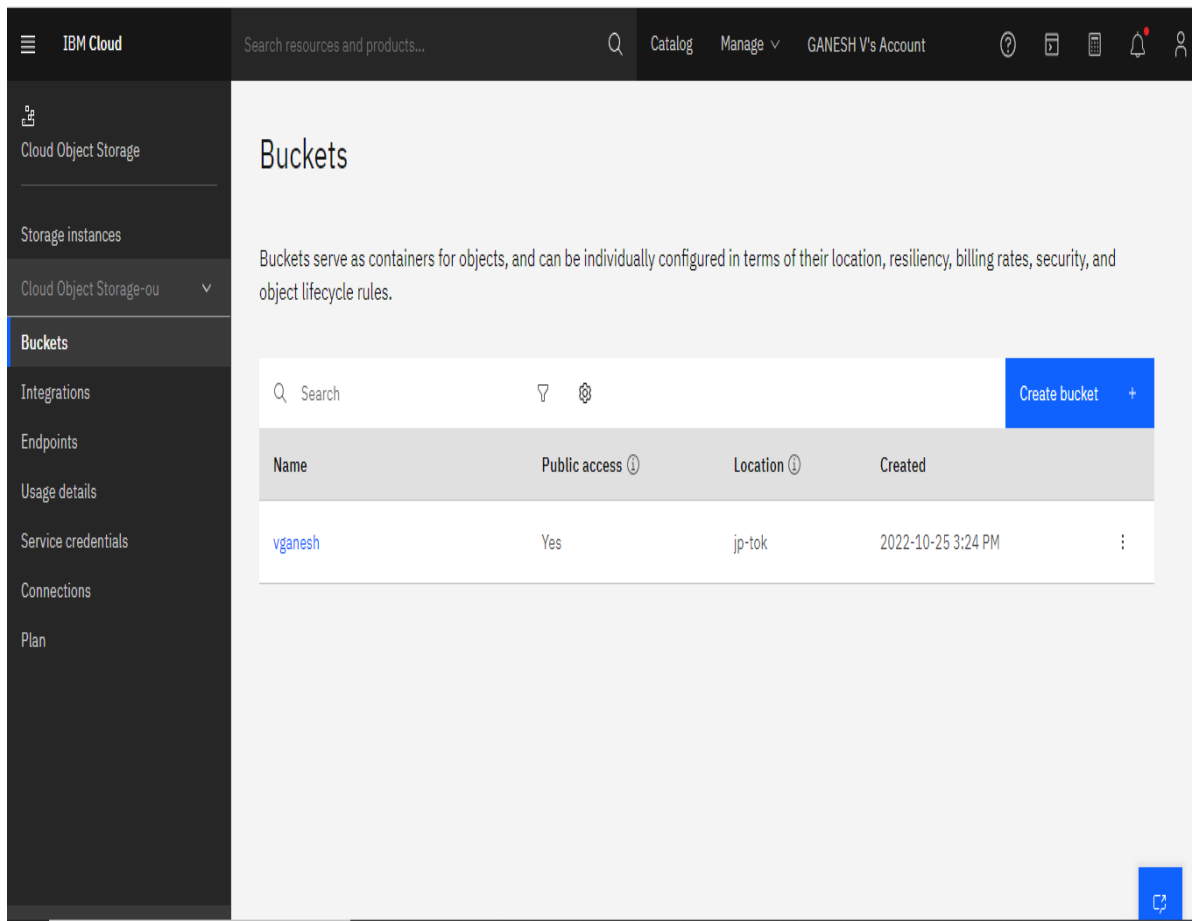


# IBM CLOUD

## ASSIGNMENT - 3

Ganesh V  
813819205017

1. Create a Bucket in IBM object storage.



2. Upload 5 images to IBM object storage and make it public. Write HTML code to display all the 5 images.

**Code File: display.html**

Uploading Images

IBM Cloud

Search resources and products...

Q

Catalog

Manage

GANESH V's Account

?

Cloud Object Storage

Storage instances

Cloud Object Storage-ou

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

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](#)

Q

Prefix filter

Upload

<input type="checkbox"/>	Object name	Archived ⓘ	Size	Last modified	
<input type="checkbox"/>	ibm...jpg		458.9 KB	2022-10-25 3:51 PM	:
<input type="checkbox"/>	ibm...jpg		2.0 MB	2022-10-25 3:51 PM	:
<input type="checkbox"/>	ibm...r.jpg		89.8 KB	2022-10-25 3:51 PM	:
<input type="checkbox"/>	ib... n.jpg		318.3 KB	2022-10-25 3:51 PM	:
<input type="checkbox"/>	ibm...le.jpg		100.8 KB	2022-10-25 3:51 PM	:

IBM Cloud

Search resources and products...

Q

Catalog

Manage

GANESH V's Account

?

Cloud Object Storage

Storage instances

Cloud Object Storage-ou

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

Writer

As a Writer, one can create/modify/delete buckets. In addition, one can upload and download the objects in the bucket.

Create access policy

Public access

Warning: Granting Public access to this bucket will allow anyone to access the bucket. To revoke public access, remove the "Public access" policy from this bucket within [Access groups](#)

[Learn more](#)

X

Status: Enabled

Role for this bucket:

Content Reader

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

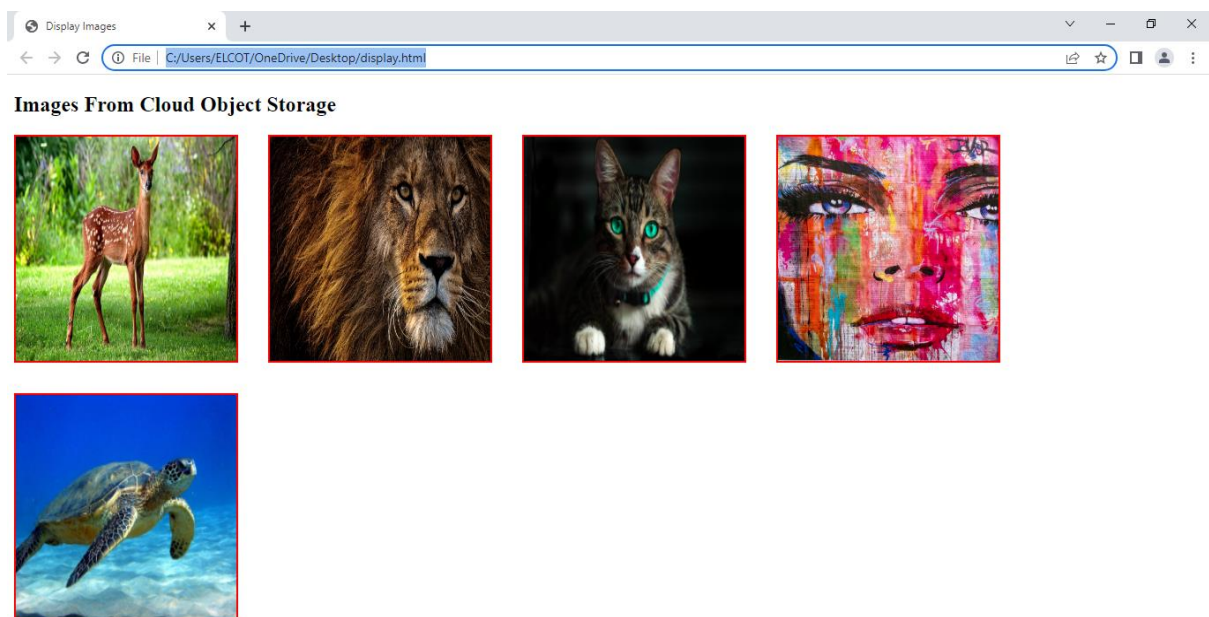
Create access policy

Style file present in local machine

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>
5          Display Images
6      </title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10     <h2>Images From Cloud Object Storage</h2>
11     
17 </html>

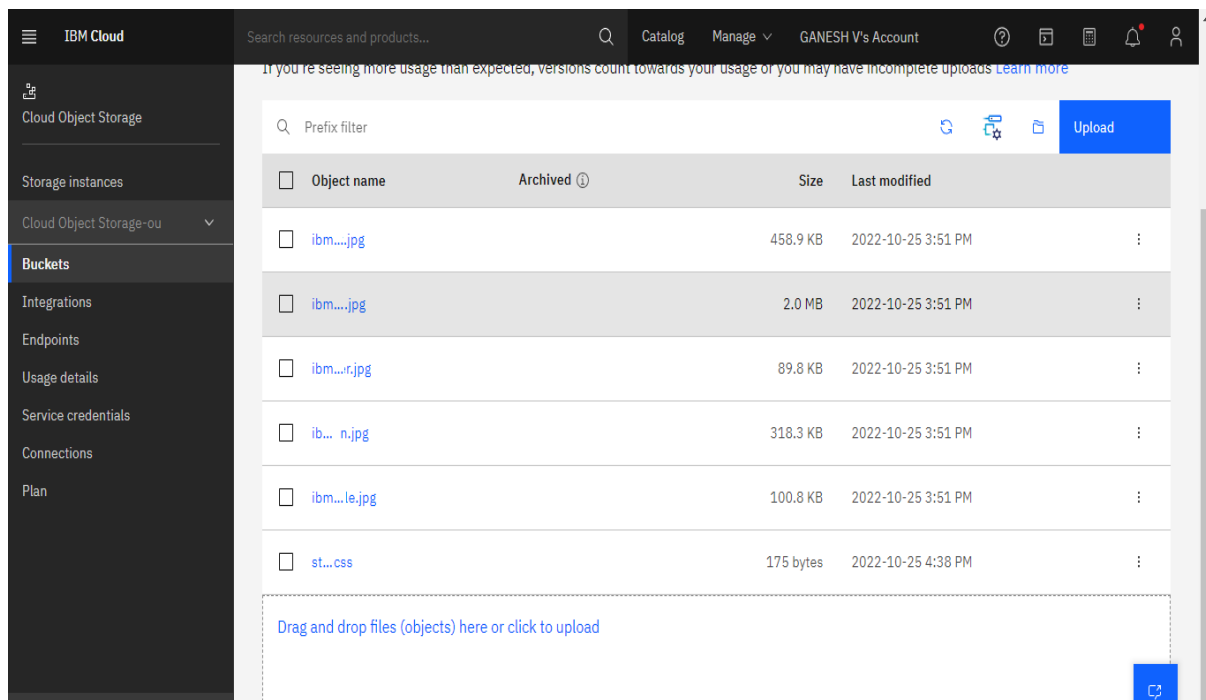
```



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

### Code File: display\_que3.html

Uploading Style page to cloud object storage

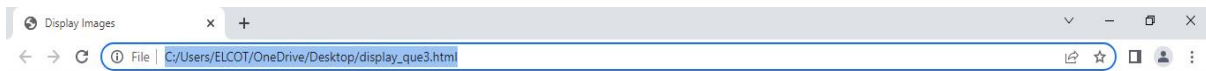


Using the style page from cloud object storage

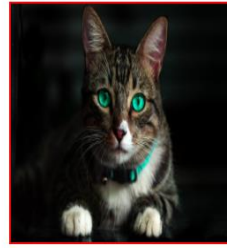
```

1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>
5      Display Images
6    </title>
7    <link rel="stylesheet" href="https://vganesh.s3.jp-tok.cloud-object-storage.appdomain.cloud/style.css">
8  </head>
9  <body>
10   <h2>Images and Style page From Cloud Object Storage </h2>
11   
17 </html>
18

```



#### Images and Style page From Cloud Object Storage

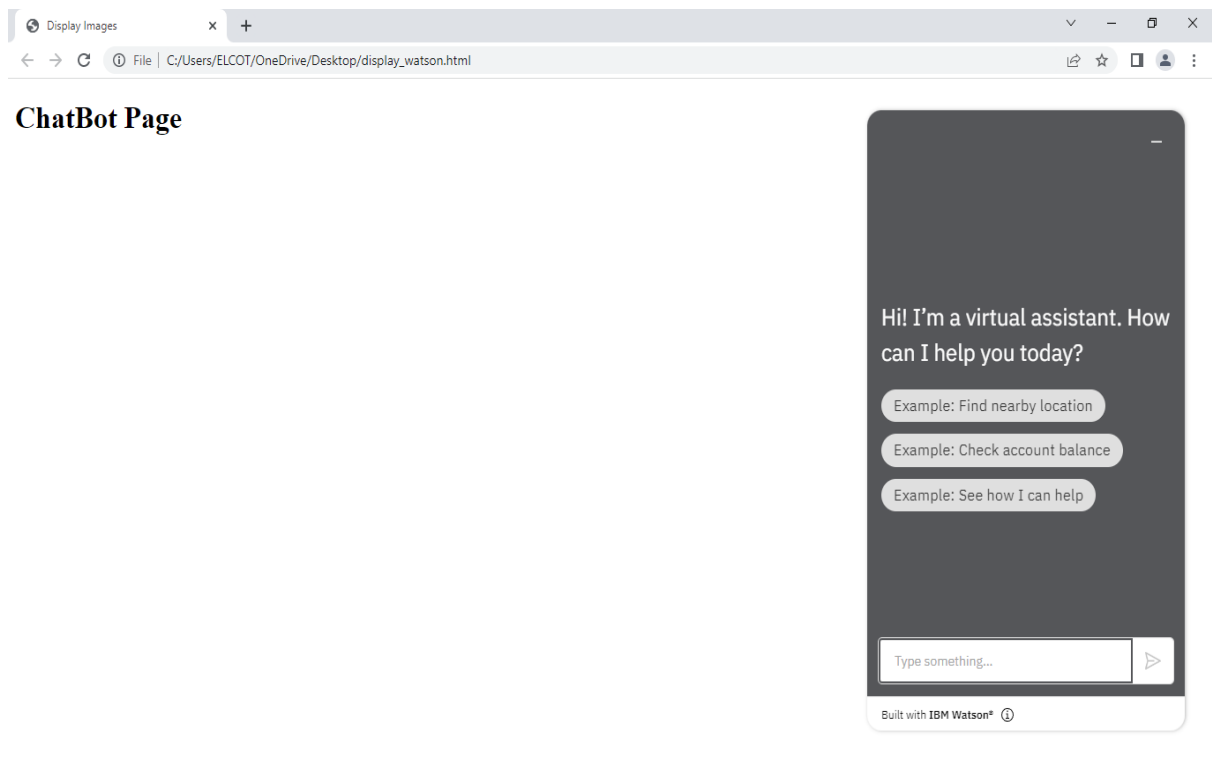


- 
4. Design a chatbot using IBM Watson assistant for hospitals. Ex: User comes with a query to know the branches for that hospital in your city. Submit the web URL of that chat bot as an assignment.

<https://ausyd.assistant.watson.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Aconversation%3Aausyd%3Aa%2Fbc771e983d7047dfb13eb10eaf26321%3Ae062c3a4-0fd2-42ae-8045-6e5a96b5a0a7%3A%3A/assistants/244703ef-066e-4fdb-982f-22e2b8e05e87/preview>

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

**Code File: display\_watson.html**



```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>
5          Display Images
6      </title>
7      <link rel="stylesheet" href="style.css">
8      <script>
9          window.watsonAssistantChatOptions = {
10             integrationID: "3fed0fba-6685-4ca9-ae6a-35122fb3484f", // The ID of this integration.
11             region: "au-syd", // The region your integration is hosted in.
12             serviceInstanceID: "e062c3a4-0fd2-42ae-8045-6e5a96b5a0a7", // The ID of your service instance.
13             onLoad: function(instance) { instance.render(); }
14         };
15         setTimeout(function(){
16             const t=document.createElement('script');
17             t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssist
18             document.head.appendChild(t);
19         });
20     </script>
21 </head>
22 <body>
23     <h1>ChatBot Page </h1>
24 </body>
25 </html>
```