

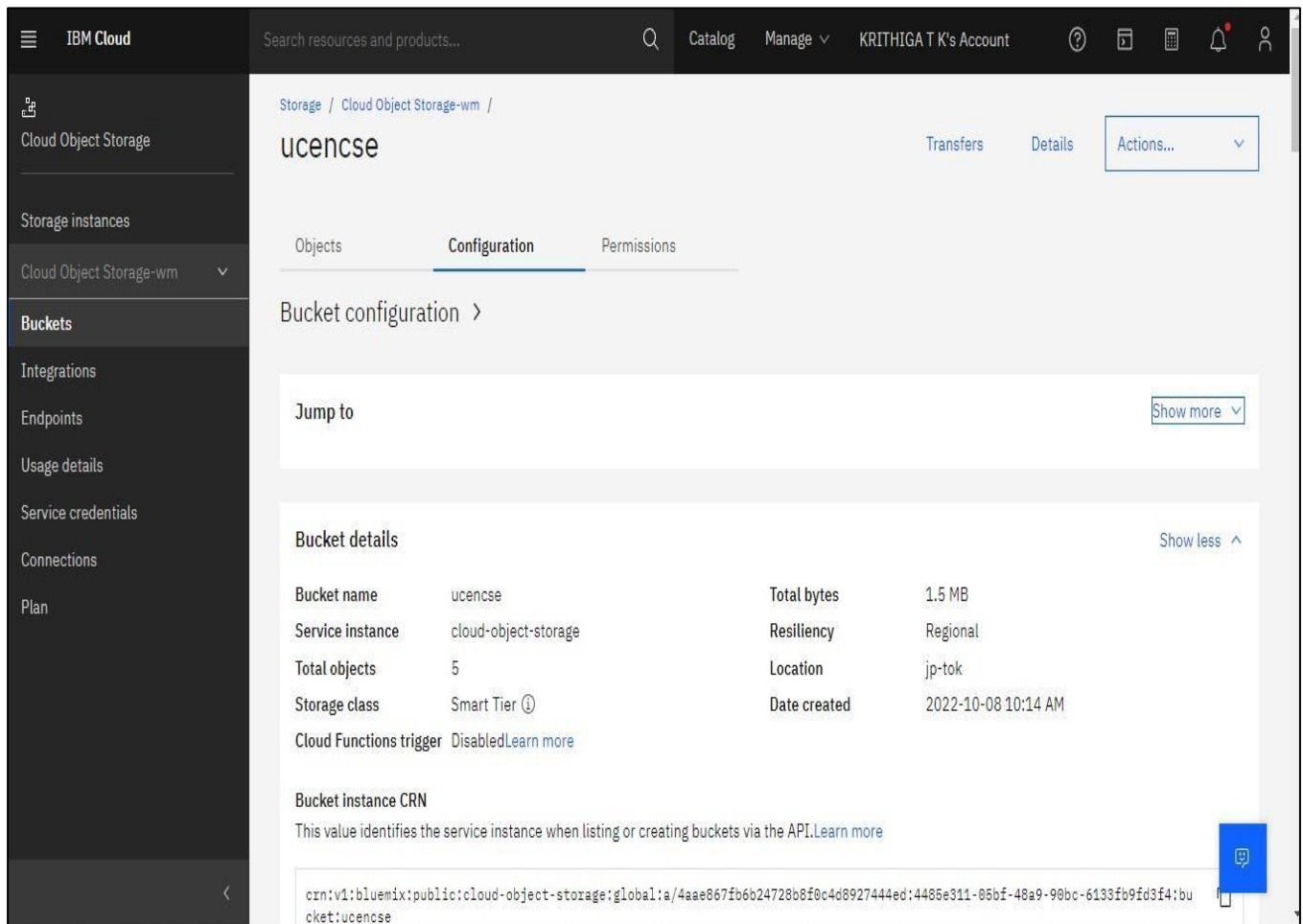
ASSIGNMENT 3

PROJECT TITLE:NEWS TRACKER APPLICATION

TEAM ID : PNT2022TMID23559

OBJECT STORAGE AND WATSON ASSISTANT IN IBM CLOUD

1. Create a Bucket in IBM object storage.



The screenshot displays the IBM Cloud console interface for managing Cloud Object Storage. The left sidebar shows the navigation menu with options like Cloud Object Storage, Storage instances, Buckets, Integrations, Endpoints, Usage details, Service credentials, Connections, and Plan. The main content area is titled 'ucencse' and shows the 'Configuration' tab. The 'Bucket configuration' section includes a 'Jump to' dropdown and a 'Show more' button. Below this, the 'Bucket details' section provides information about the bucket, including its name, service instance, total objects, storage class, and date created. The 'Bucket instance CRN' is also displayed, along with a 'Learn more' link.

Bucket details			
Bucket name	ucencse	Total bytes	1.5 MB
Service instance	cloud-object-storage	Resiliency	Regional
Total objects	5	Location	jp-tok
Storage class	Smart Tier ⓘ	Date created	2022-10-08 10:14 AM
Cloud Functions trigger	Disabled Learn more		

Bucket instance CRN
This value identifies the service instance when listing or creating buckets via the API. [Learn more](#)

```
crn:v1:bluemix:public:cloud-object-storage:global:a/4aae867fb6b24728b8f0c4d8927444ed:4485e311-05bf-48a9-90bc-6133fb9fd3f4:bucket:ucencse
```

Bucket was created

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

IBM Cloud

Cloud Object Storage

Storage instances

Cloud Object Storage-wm

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

Search resources and products...

Q

Catalog

Manage

KRITHIGA T K's Account

?

ucencse

Transfers

Details

Actions...

Objects

Configuration

Permissions

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"/>	image1.jpg		389.8 KB	2022-10-08 10:22 AM	⋮
<input type="checkbox"/>	image2.jpg		181.0 KB	2022-10-08 10:22 AM	⋮
<input type="checkbox"/>	image3.jpg		161.4 KB	2022-10-08 10:22 AM	⋮
<input type="checkbox"/>	image4.jpg		750.4 KB	2022-10-08 10:21 AM	⋮
<input type="checkbox"/>	images.jpg		3.9 KB	2022-10-08 10:21 AM	⋮

5 images was uploaded in the cloud object storage

IBM Cloud

Cloud Object Storage

Storage instances

Cloud Object Storage-wm

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

Search resources and products...

Q

Catalog

Manage

KRITHIGA T K's Account

?

Storage / Cloud Object Storage-wm /

ucencse

Transfers

Details

Actions...

Objects

Configuration

Permissions

Bucket access policies

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

Status: Enabled

Role for this bucket:

Content Reader

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

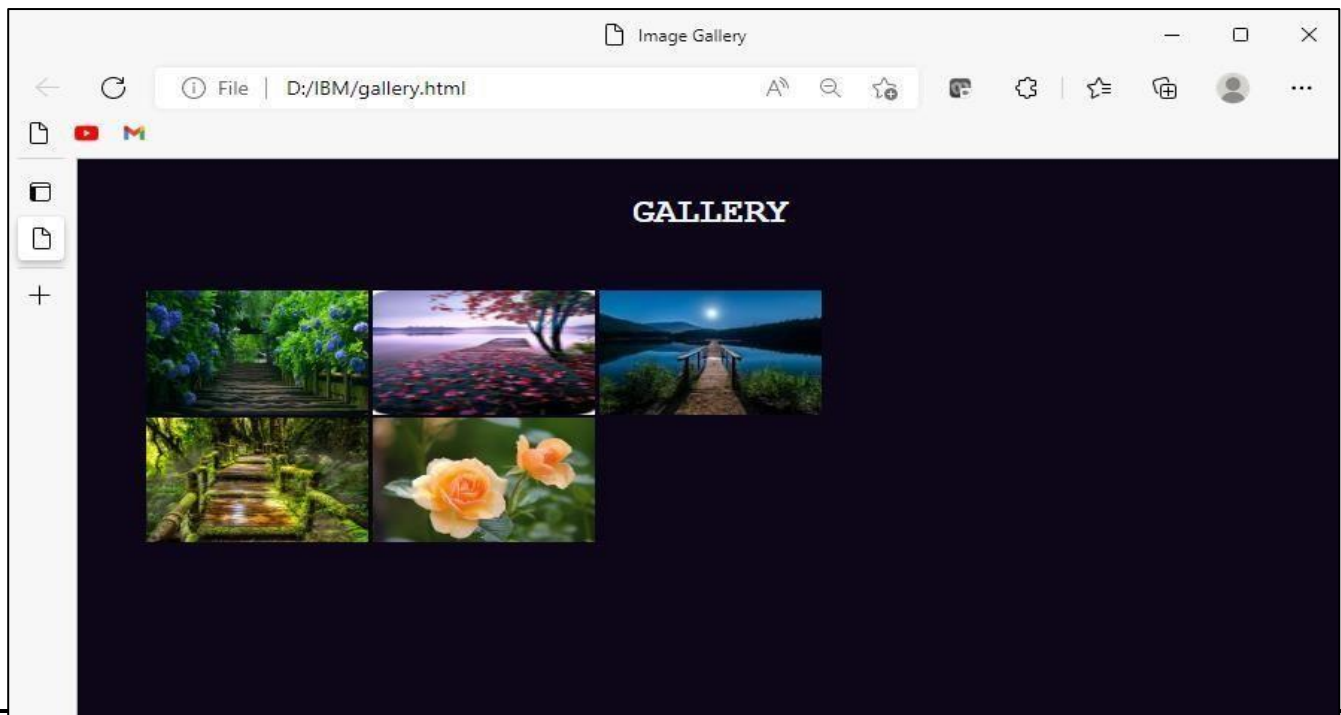
Create access policy

Public access is enabled

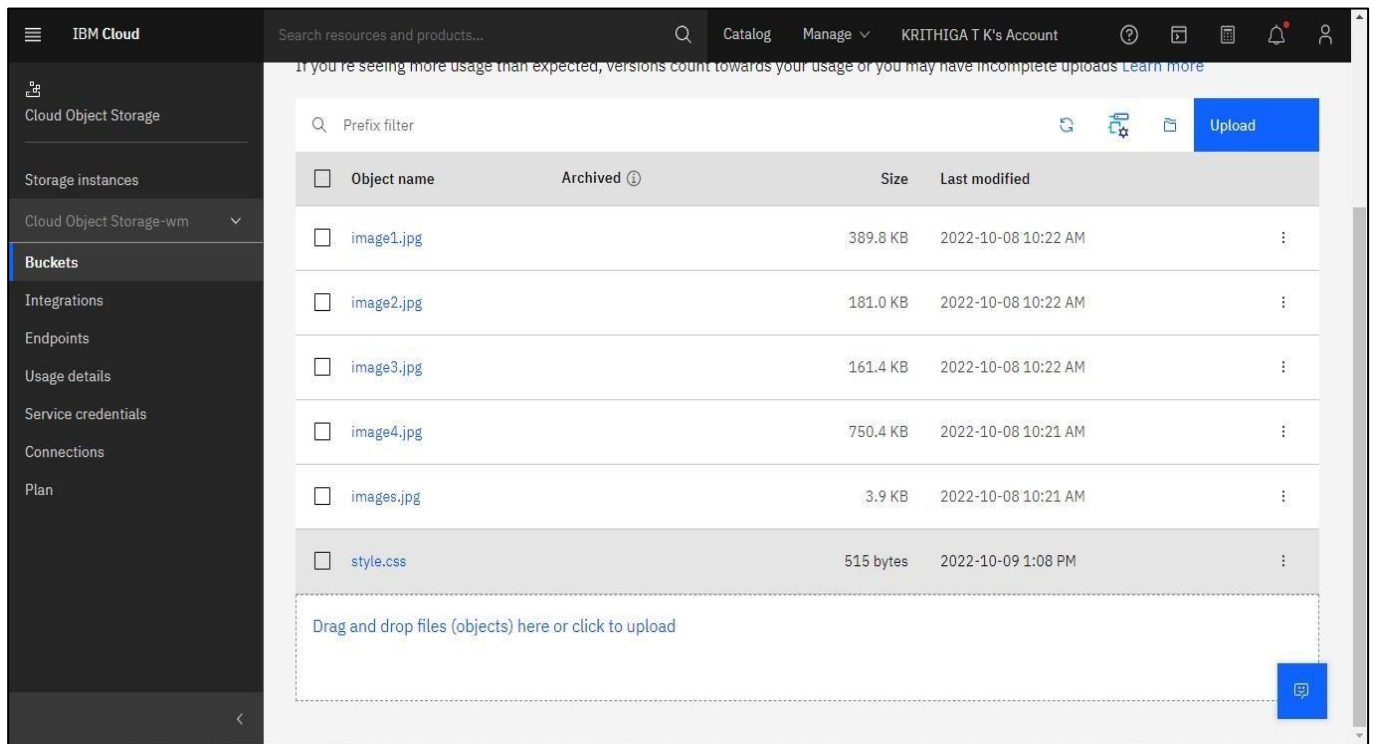
Gallery.html

```
1  gallery.html x
2  <html>
3  <head>
4    <title>Image Gallery</title>
5    <style type="text/css">
6      body{
7        background-color: rgb(13, 6, 24);
8      }
9      #gallery{
10       margin-top: 10%;
11       margin: 5%;
12       line-height:0;
13       column-count:5;
14       column-gap:5px;
15     }
16     #gallery img{
17       width: 100% !important;
18       height: 40% !important;
19       margin-bottom:5px;
20     }
21     h2{
22       text-align: center;
23       font-family: Courier;
24       font-size: 50px;
25       margin-top: 3%;
26       font-weight: bolder;
27       color: aliceblue;
28     }
29   </style>
30 </head>
31 <body>
32   <h2>GALLERY</h2>
33   <div id="gallery">
34     
35     
36     
37     
38     
39   </div>
40 </body>
41 </html>
```

Output



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



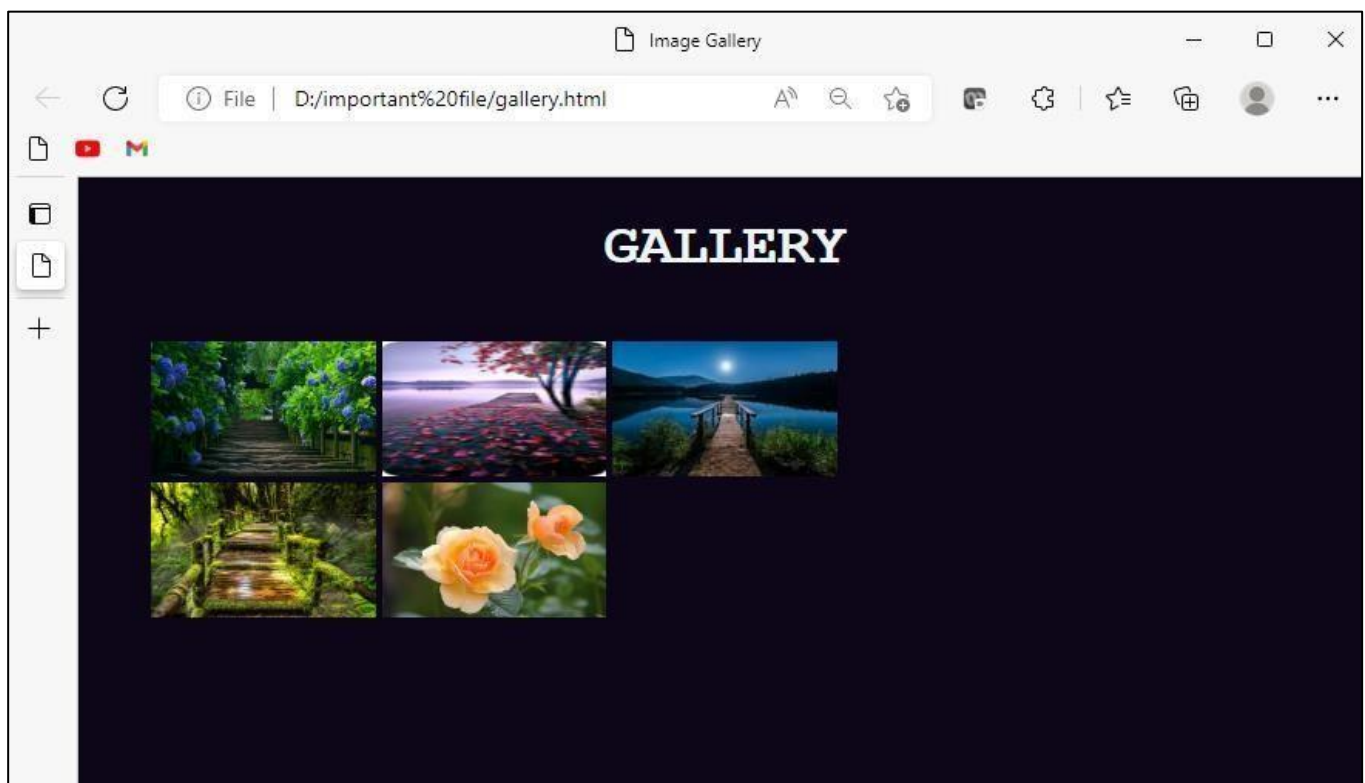
Style.css



Gallery.html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Image Gallery</title>
5     <link rel="stylesheet" href="https://ucencse.s3.jp-tok.cloud-object-storage.appdomain.cloud/style.css">
6   </head>
7   <body>
8     <h2>GALLERY</h2>
9     <div id="gallery">
10      
11      
12      
13      
14      
15    </div>
16  </body>
17 </html>
```

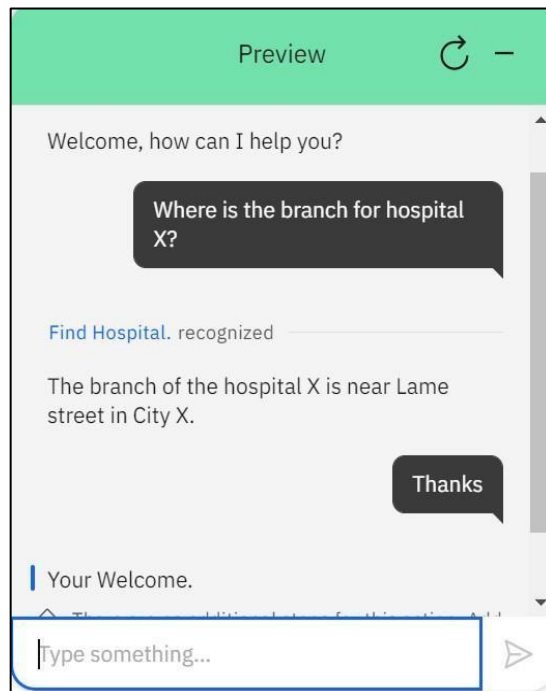
Output



4. Design a chat bot using IBM Watson assistant for a hospital. Example the user comes with a query to know the branches of that hospital in your city. Submit the web URL of that chat bot as an assignment.

The screenshot displays the IBM Watson Assistant Lite interface. At the top, there's a navigation bar with "IBM Watson Assistant Lite", "Upgrade", "Chatbot", and "Learning center". Below this, the main workspace is divided into several sections:

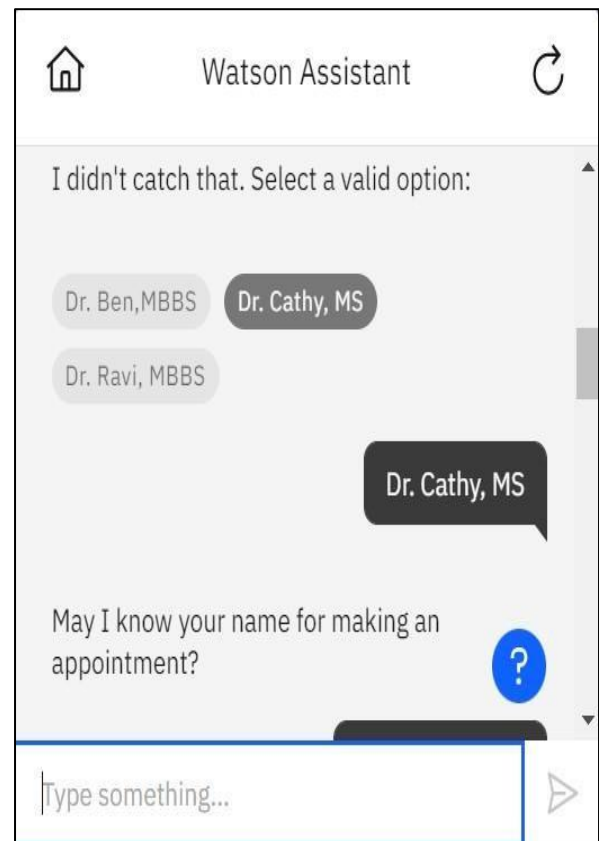
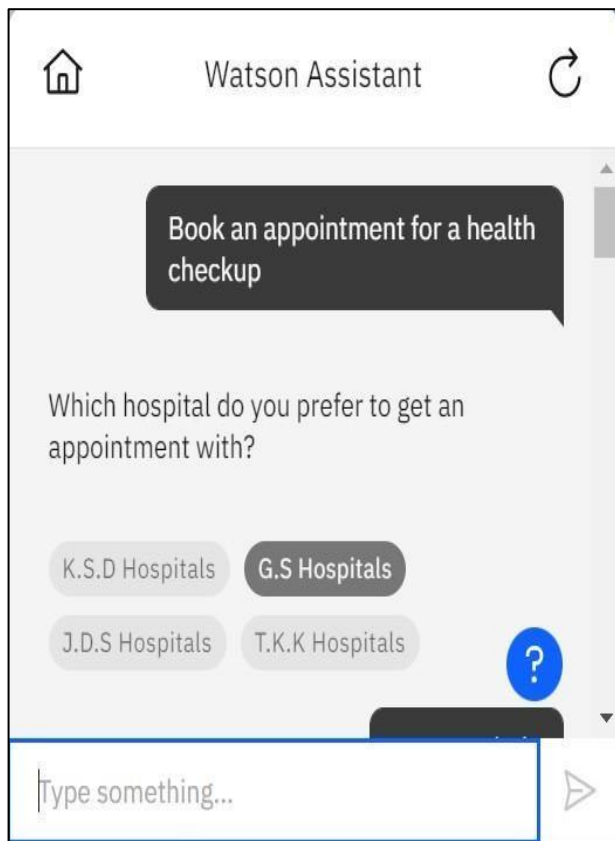
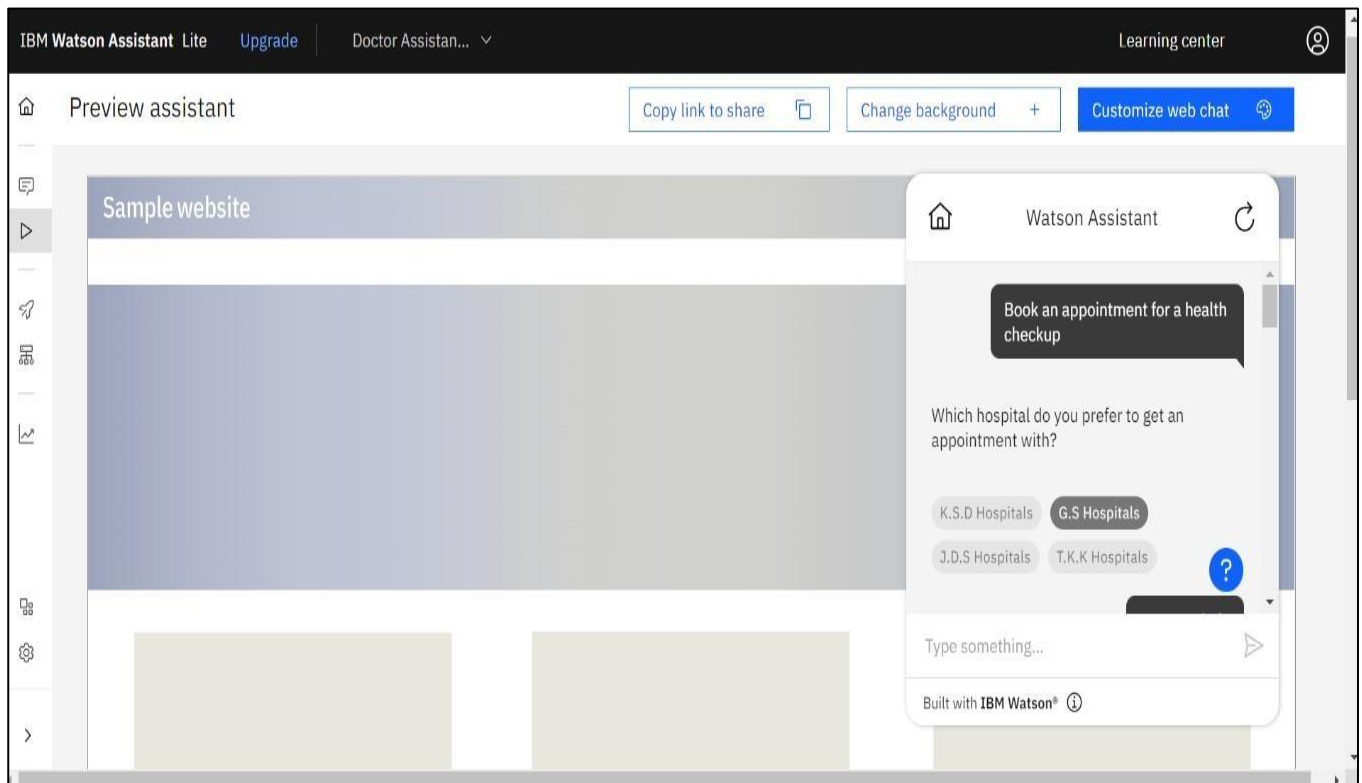
- Greet customer:** A dropdown menu showing "Assistant starts with step 1".
- Conversation steps:** A list of steps. Step 1 is "Welcome, how can I help you?" with a status of "Action complete". A "New step +" button is at the bottom.
- Action starts:** A section explaining that the action starts when the assistant is first opened or engaged. It notes "No training examples necessary" and "Use the default greeting or customize your own."
- Preview:** A floating window showing a simulated chat conversation:
 - User: "Where is the branch for hospital X?"
 - Assistant: "Find Hospital. recognized"
 - Assistant: "The branch of the hospital X is near Lane street in City X."
 - User: "Thanks"
 - Assistant: "Your Welcome."

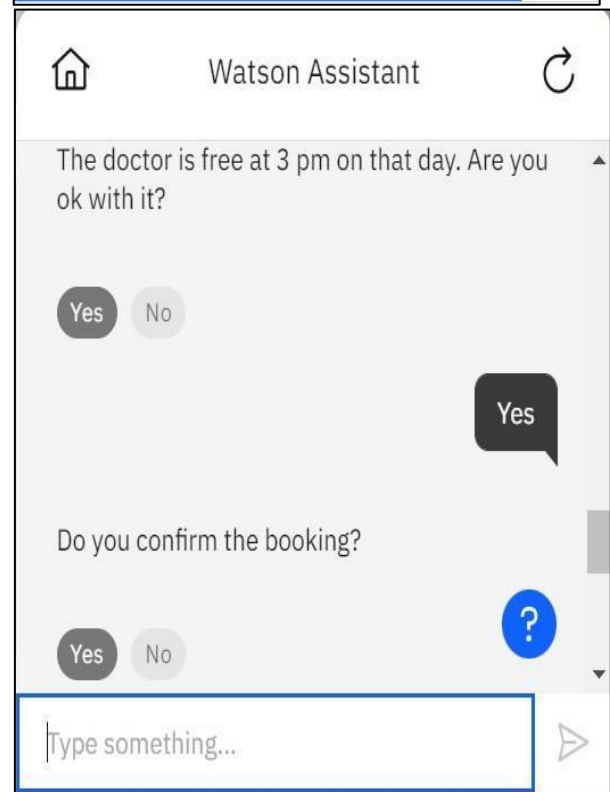
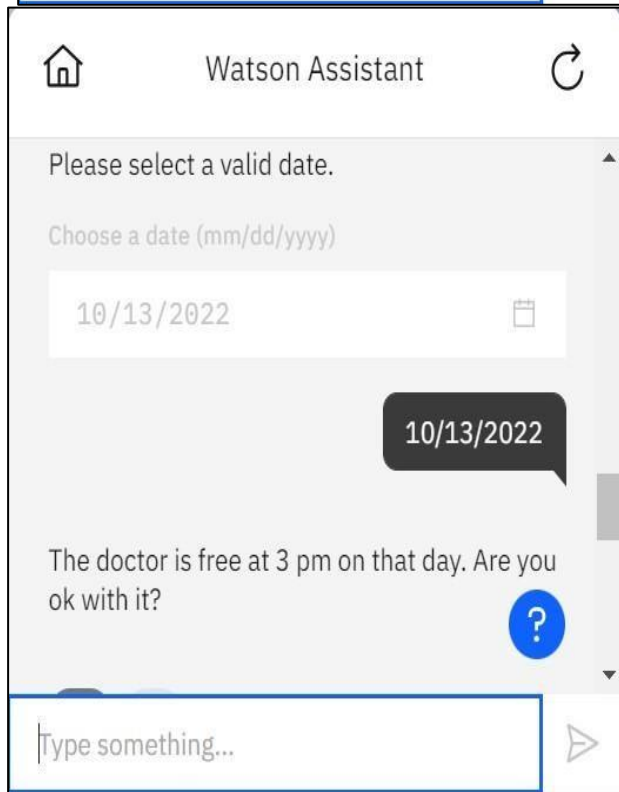
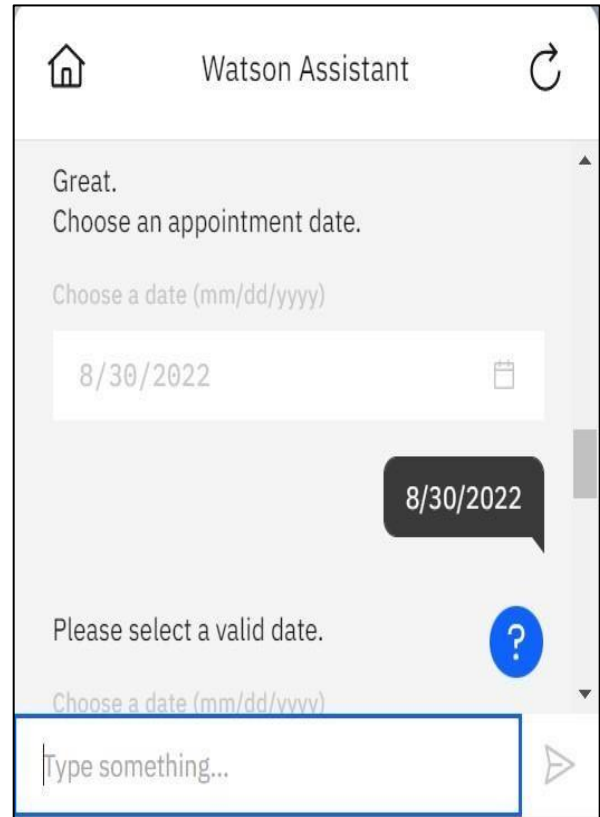
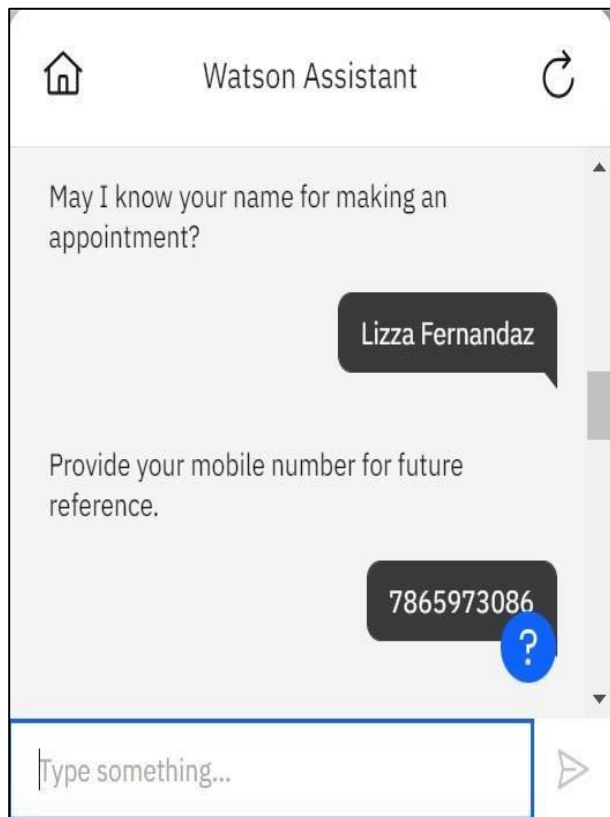


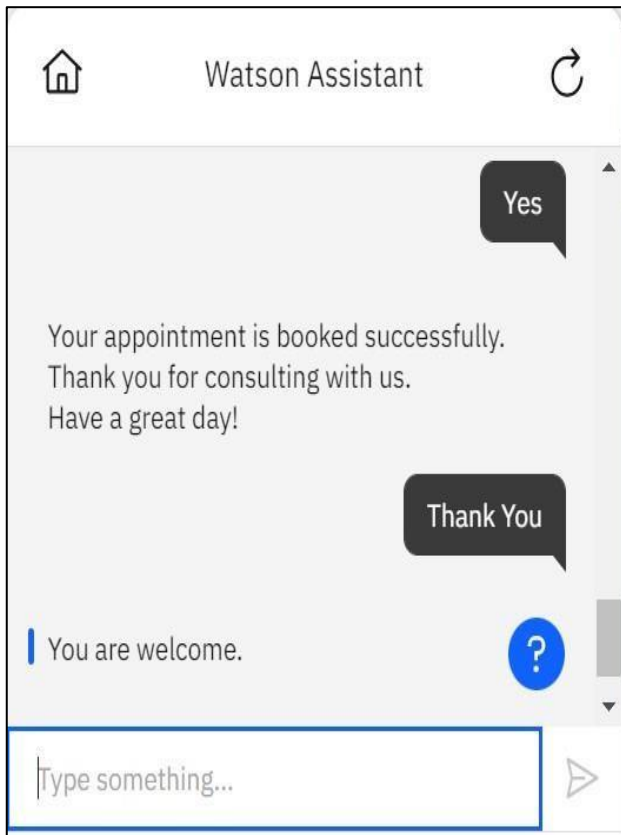
Chatbot Access URL

<https://webchat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Fjp-tok.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupxc1c094d9-c048-4877-b09e-202b2a623fda%3A%3A28be94d0-4a06-4bb7a0af358479d049ca&integrationID=640b8a9b-167a-4301-8cf9-f956c3444e5f®ion=jptok&serviceInstanceID=c1c094d9-c048-4877-b09e202b2a623fda>

5. Create a Watson assistant service with ten steps and use three conditions in it. Load that script in the html page.







Watsonassistantchatbot.html

```
watsonassistantchatbox.html
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Home Page</title>
5  </head>
6  <body>
7    <script>
8      window.watsonAssistantChatOptions = {
9        integrationID: "0edf8511-f378-498b-9540-7bcf7e985396", // The ID of this integration.
10       region: "jp-tok", // The region your integration is hosted in.
11       serviceInstanceID: "c1c094d9-c048-4877-b09e-202b2a623fda", // The ID of your service instance.
12       onLoad: function(instance) { instance.render(); }
13     };
14     setTimeout(function(){
15       const t=document.createElement('script');
16       t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
17       document.head.appendChild(t);
18     });
19   </script>
20 </body>
21 </html>
```

Chatbot access URL

<https://webchat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageUrl=https%3A%2F%2Fjp-tok.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupxc1c094d9-c0484877-b09e-202b2a623fda%3A%3A970fcc61-3fe8-4aa3962b7c9d376ceb12&integrationID=0edf8511-f378-498b-9540-7bcf7e985396®ion=jptok&serviceInstanceID=c1c094d9-c048-4877-b09e202b2a623fda>