

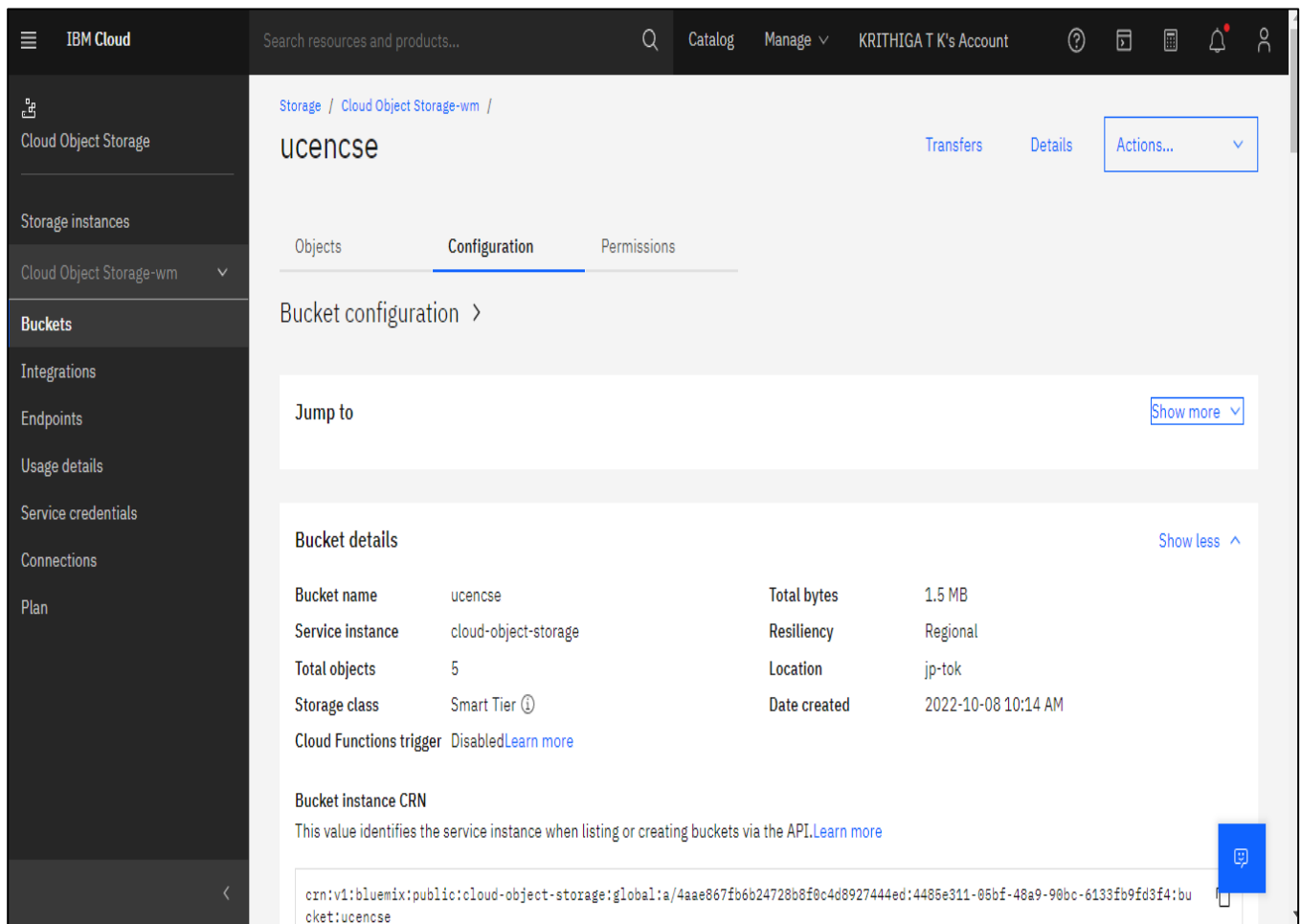
ASSIGNMENT 3

PROJECT TITLE : SKILL AND JOB RECOMMENDER

TEAM ID : PNT2022TMID34851

OBJECT STORAGE AND WATSON ASSISTANT IN IBM CLOUD

1. Create a Bucket in IBM object storage.



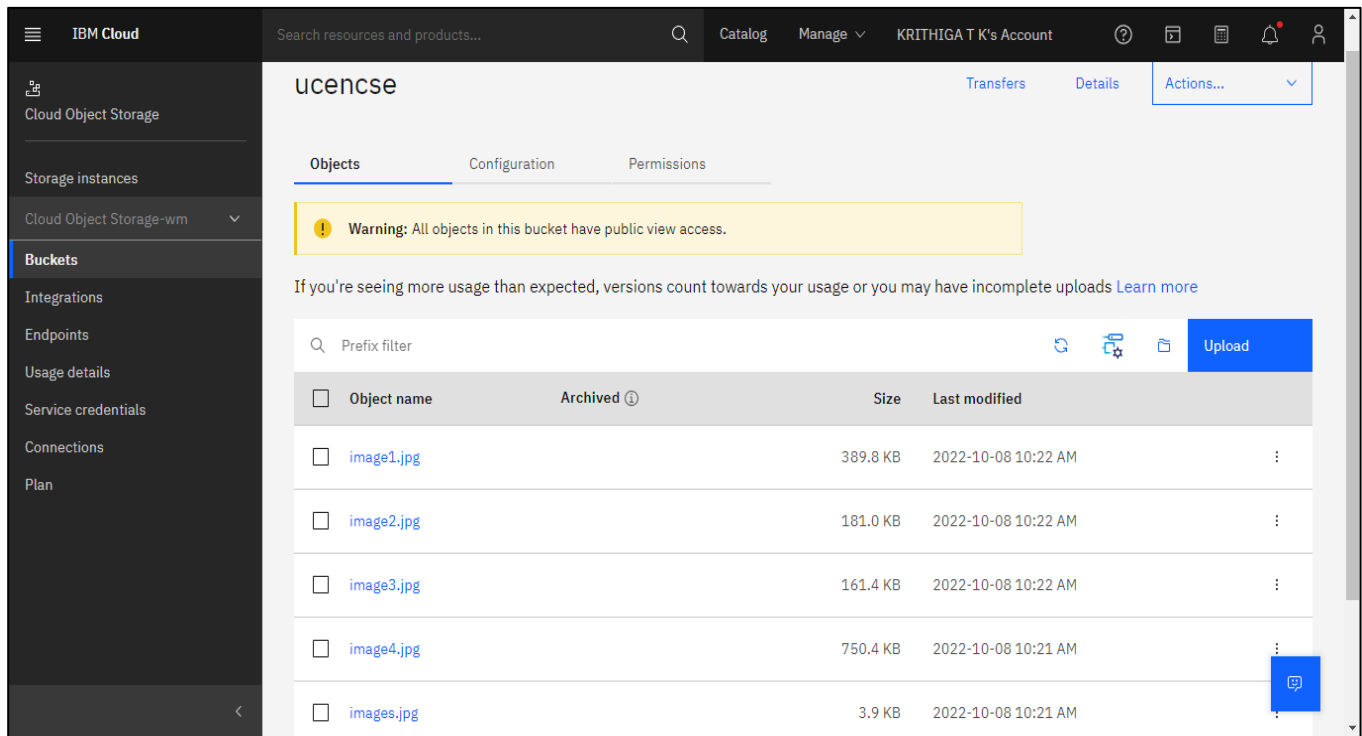
The screenshot displays the IBM Cloud console interface for managing Cloud Object Storage. The left sidebar shows the navigation menu with 'Buckets' selected. The main content area shows the configuration for a bucket named 'ucencse'. The 'Configuration' tab is active, displaying the 'Bucket configuration' section. Below this, there is a 'Jump to' section with a 'Show more' dropdown. The 'Bucket details' section provides the following information:

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		

Below the details, the 'Bucket instance CRN' is shown: 'crn:v1:bluemix:public:cloud-object-storage:global:a/4aae867fb6b24728b8f0c4d8927444ed:4485e311-06bf-48a9-90bc-6133fb9fd3f4:bucket:ucencse'. A 'Show less' link is available to the right of the details section.

Bucket was created

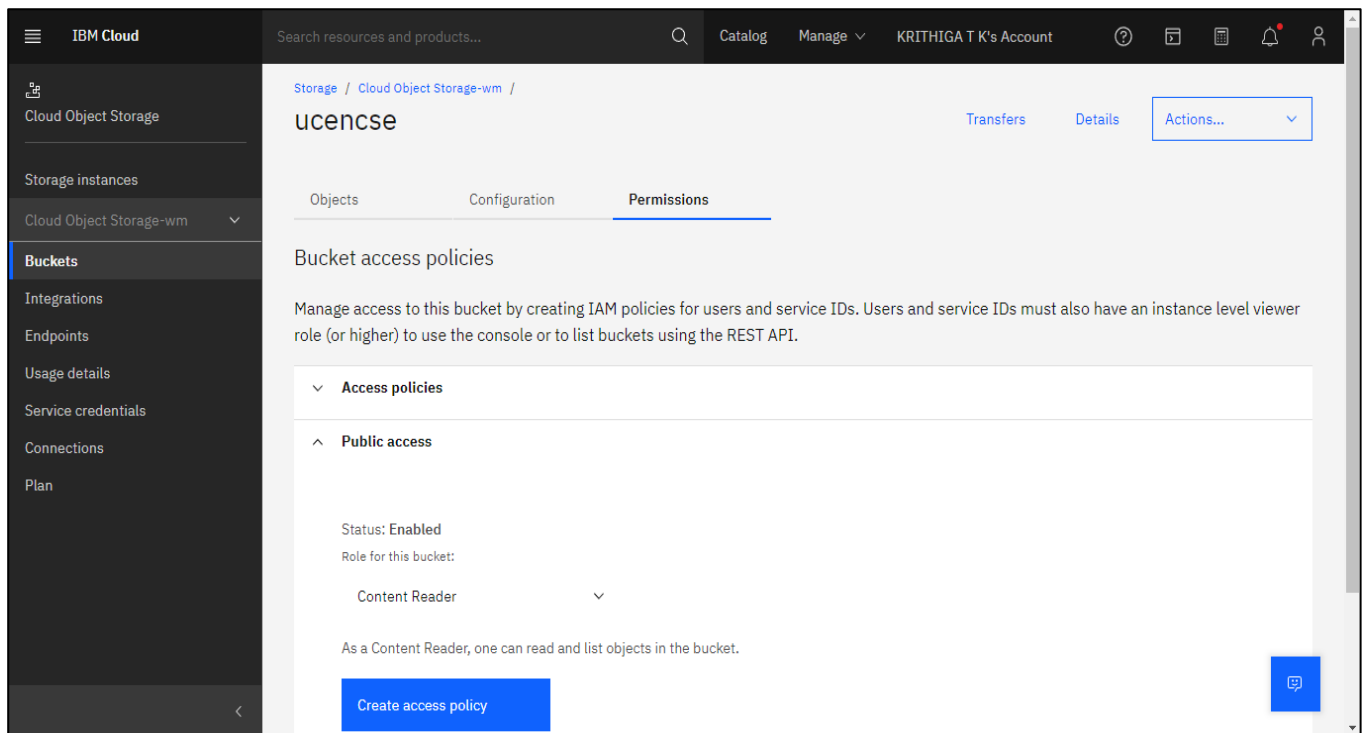
2. Upload 5 images to IBM object storage and make it public. Write html code to displaying all the 5 images.



The screenshot shows the IBM Cloud Object Storage console. The left sidebar contains navigation links: IBM Cloud, Cloud Object Storage, Storage instances, Cloud Object Storage-wm, Buckets, Integrations, Endpoints, Usage details, Service credentials, Connections, and Plan. The main content area is for the bucket 'ucencse'. It has tabs for Objects, Configuration, and Permissions. A yellow warning box states: 'Warning: All objects in this bucket have public view access.' Below this, a message says: 'If you're seeing more usage than expected, versions count towards your usage or you may have incomplete uploads [Learn more](#)'. There is a search bar with 'Prefix filter' and an 'Upload' button. A table lists the objects:

<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



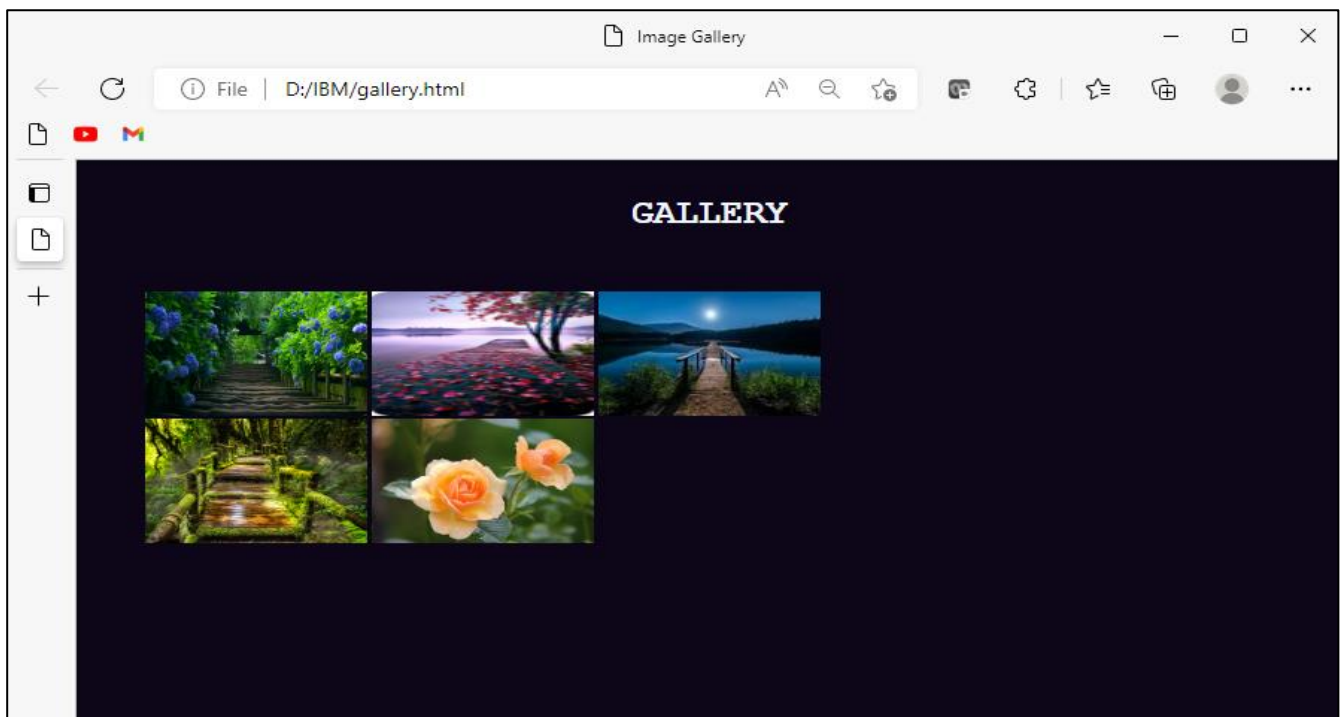
The screenshot shows the IBM Cloud Object Storage console for the bucket 'ucencse'. The left sidebar is the same as the previous screenshot. The main content area has tabs for Objects, Configuration, and Permissions. The 'Permissions' tab is selected, showing 'Bucket access policies'. A message states: '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.' Below this, there is a section for 'Access policies' with a sub-section for 'Public access'. The 'Public access' section shows 'Status: Enabled' and 'Role for this bucket: Content Reader'. A description says: 'As a Content Reader, one can read and list objects in the bucket.' There is a 'Create access policy' button.

Public access is enabled

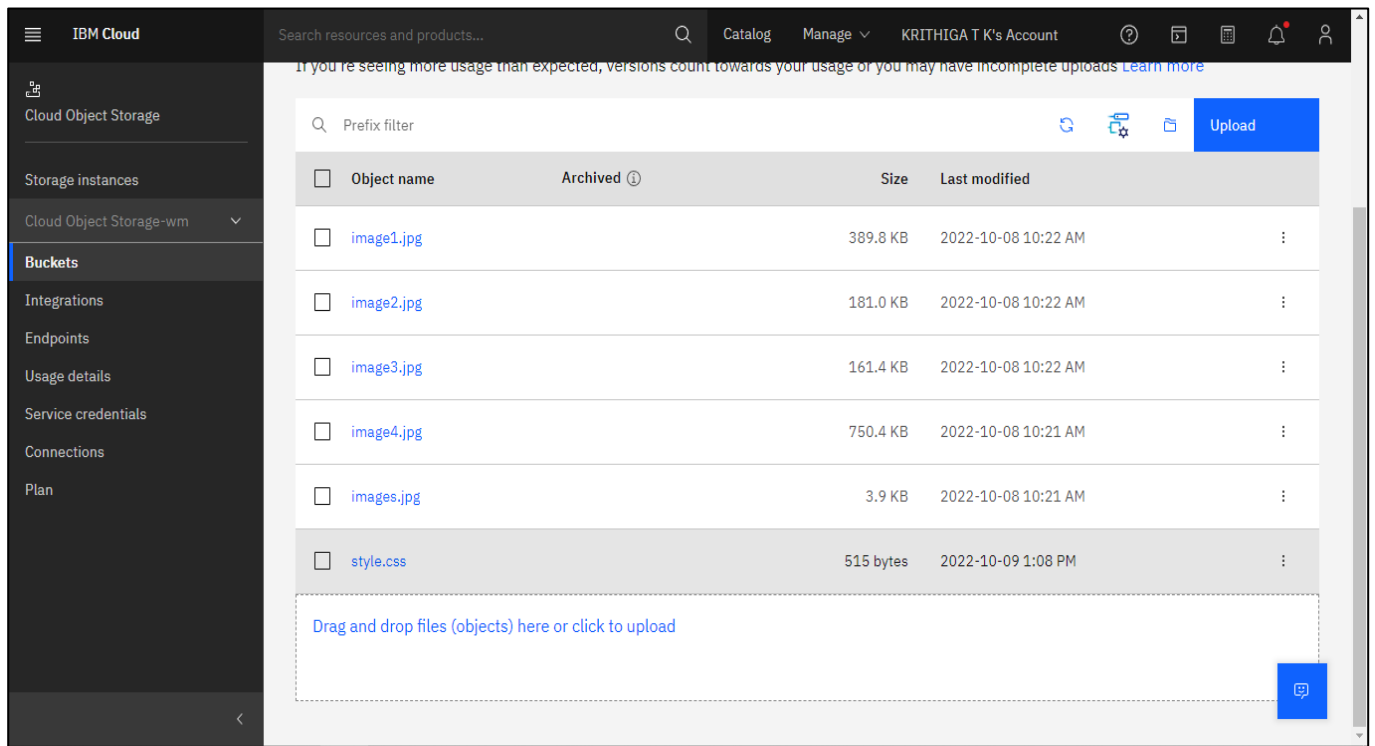
Gallery.html

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

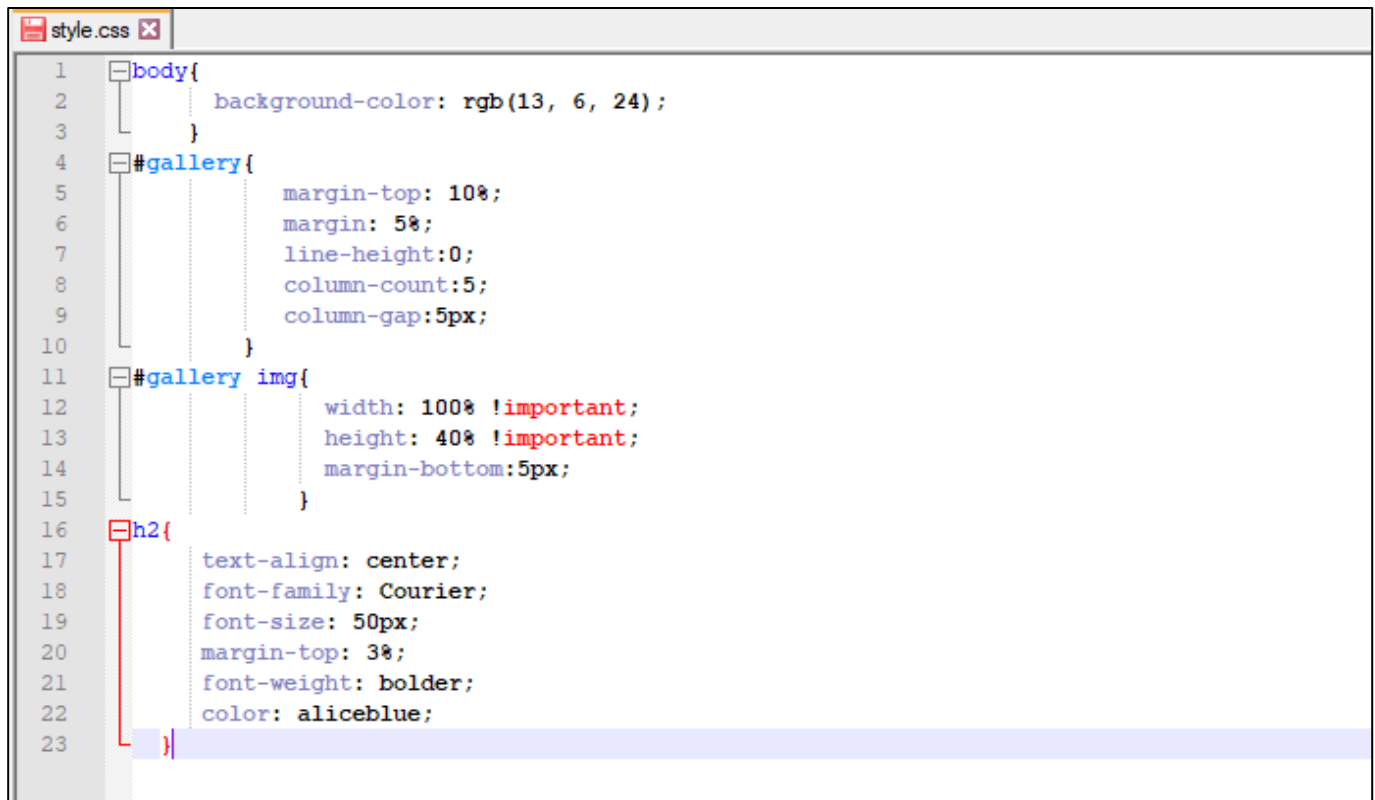
Output



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



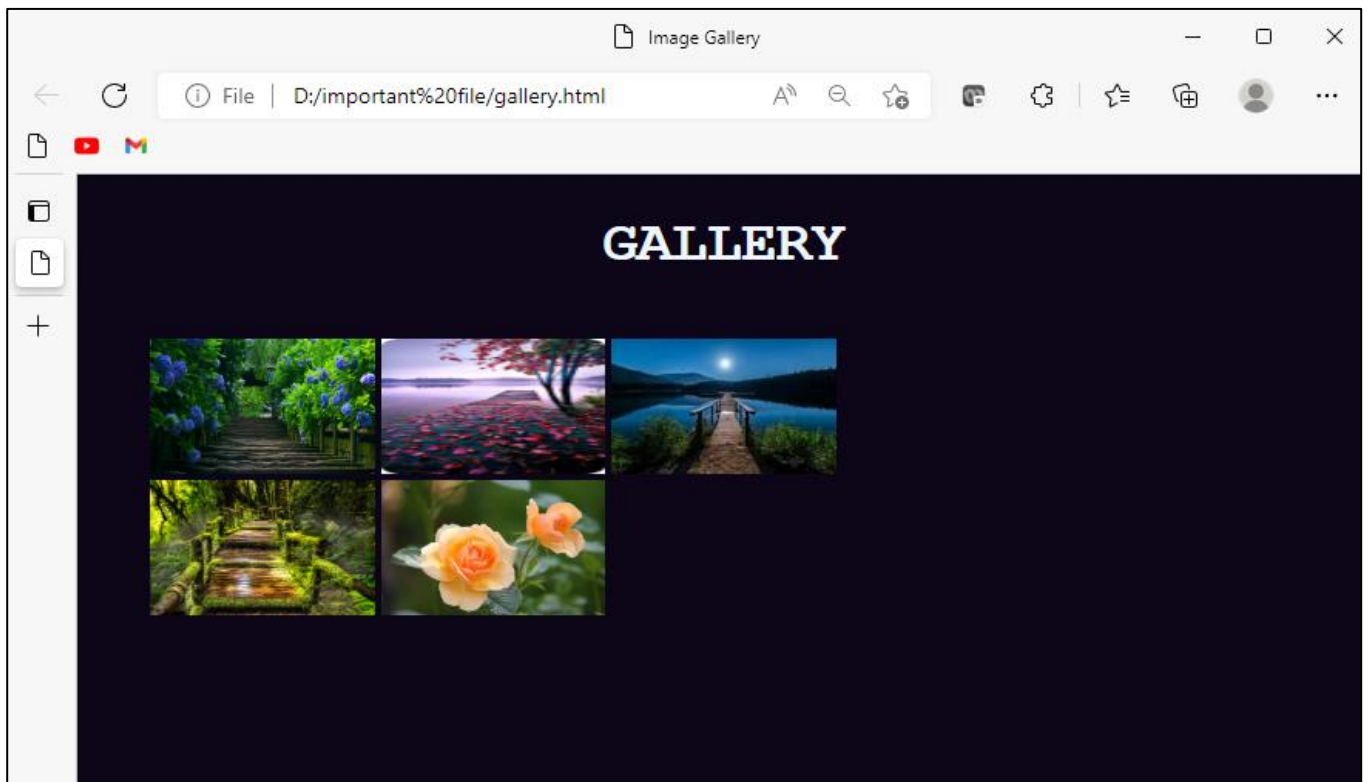
Style.css



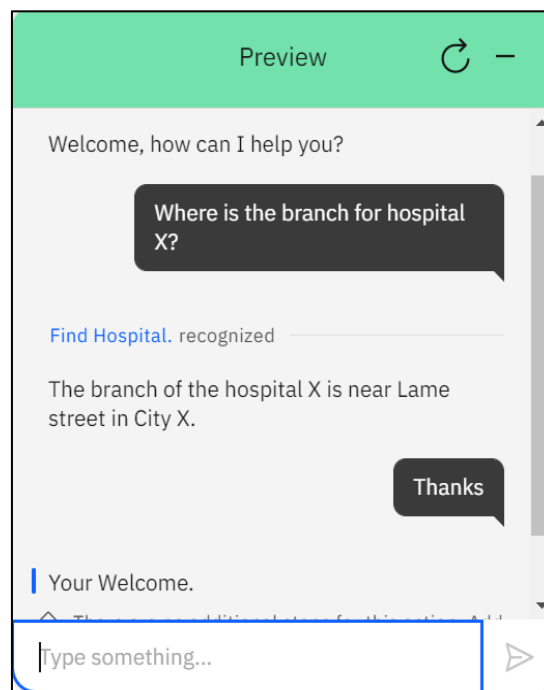
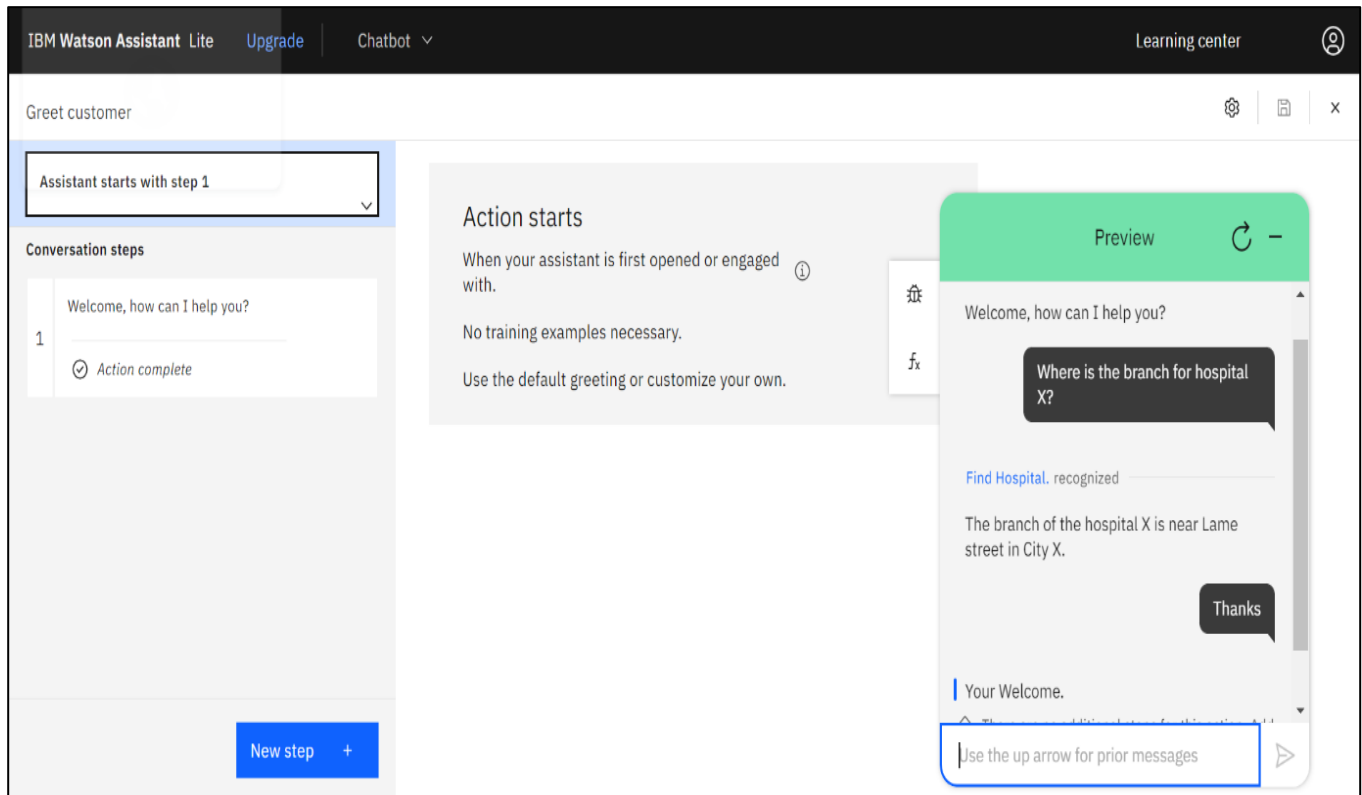
Gallery.html

```
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



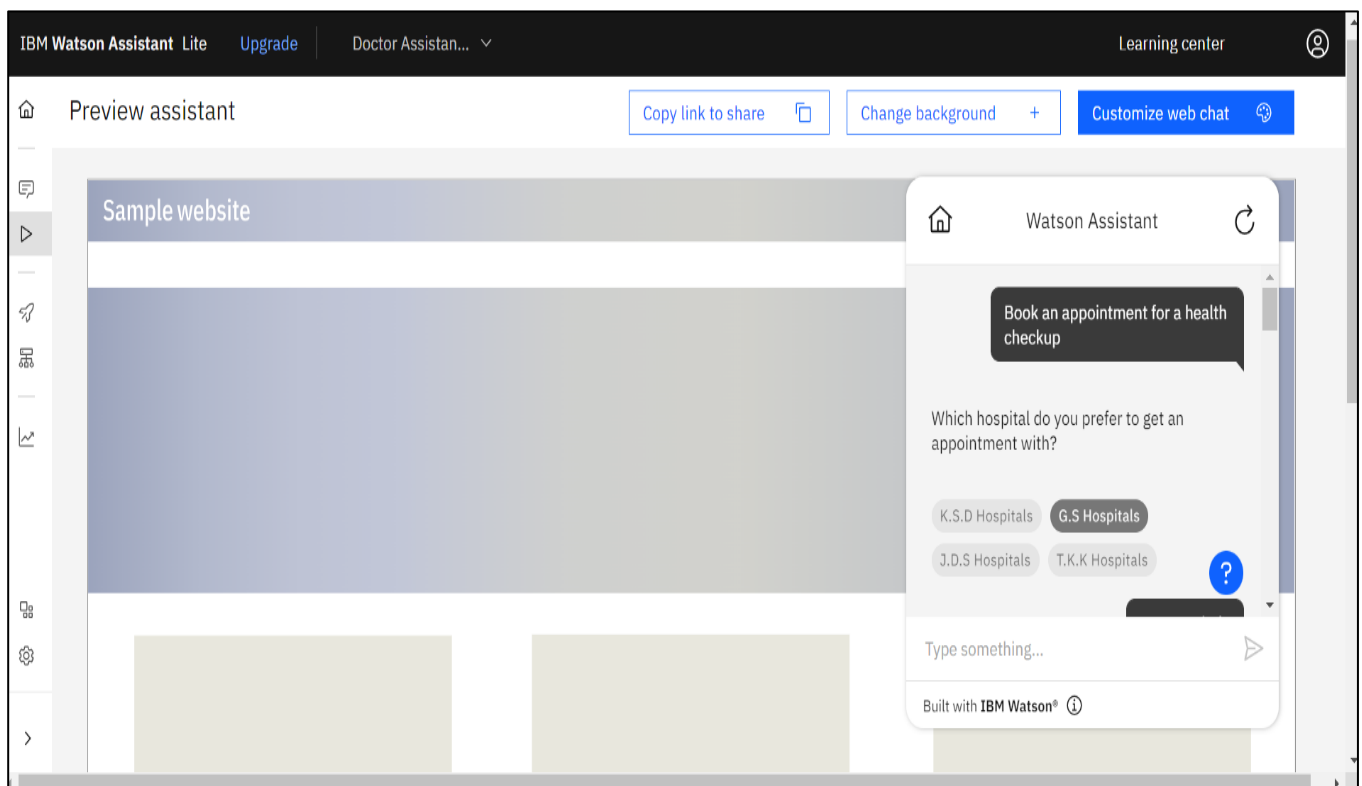
4. Design a chat bot using IBM Watson assistant for a hospital. Example the user comes with a query to know the branches of that hospital in your city. Submit the web URL of that chat bot as an assignment.

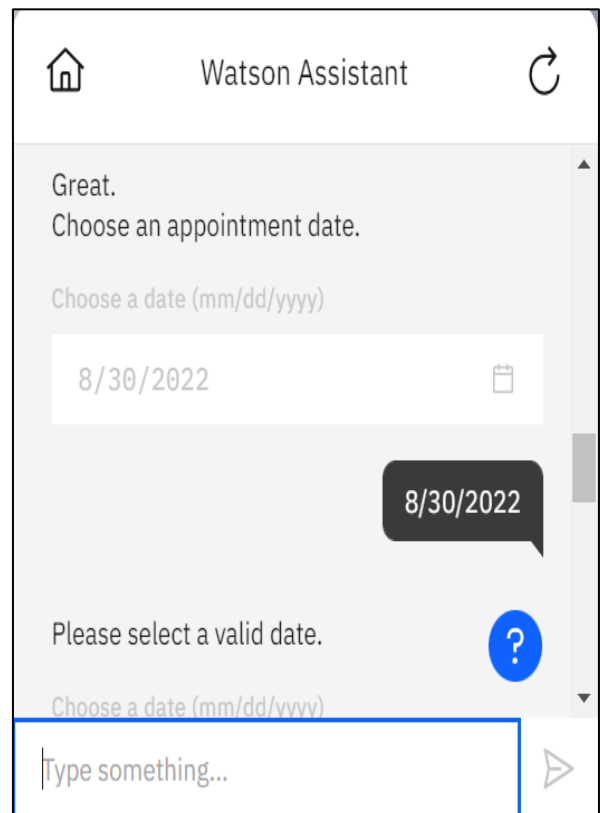
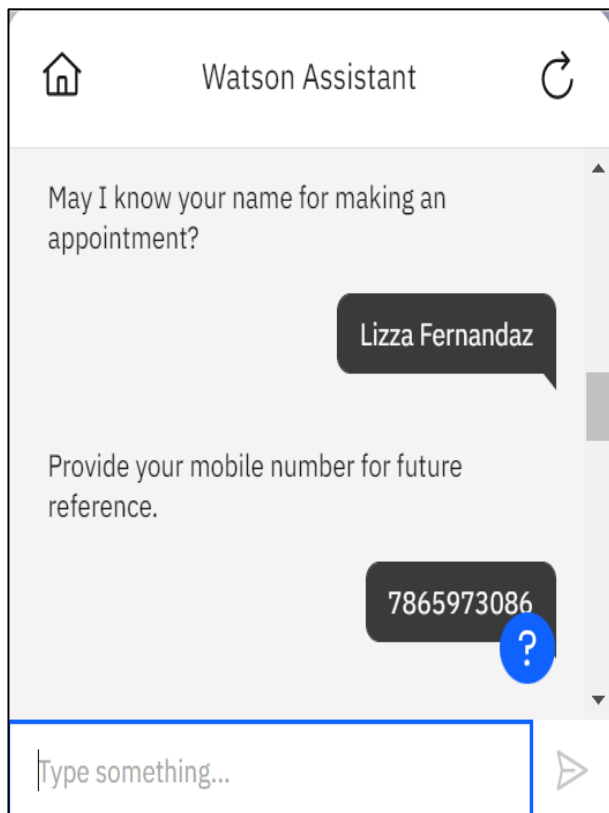
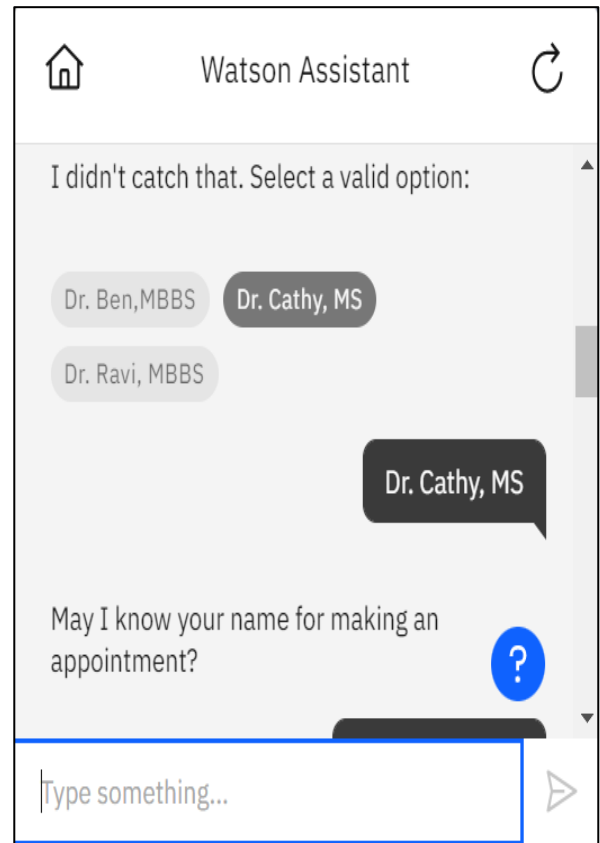
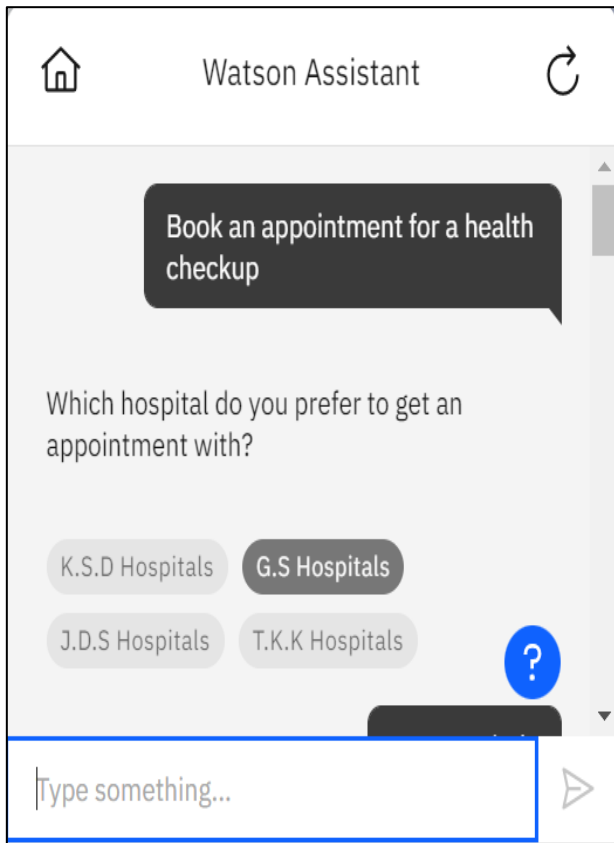


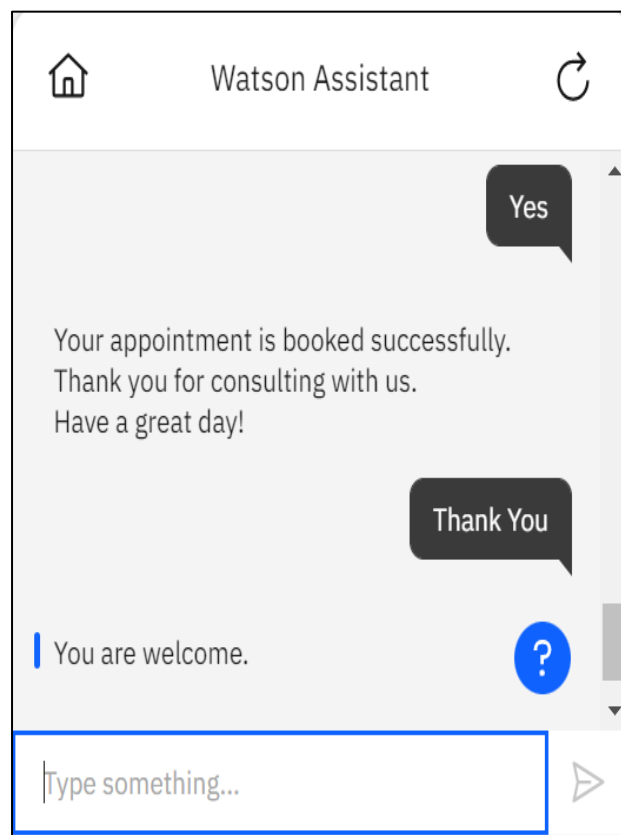
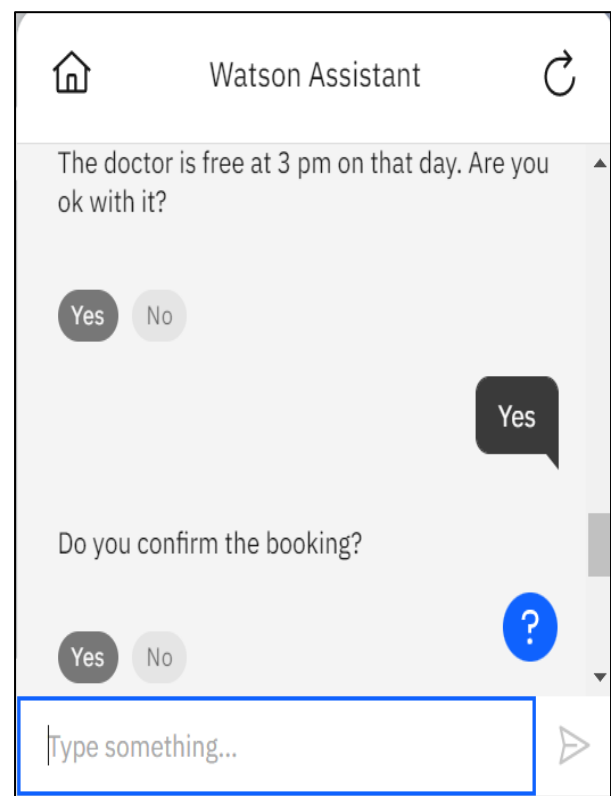
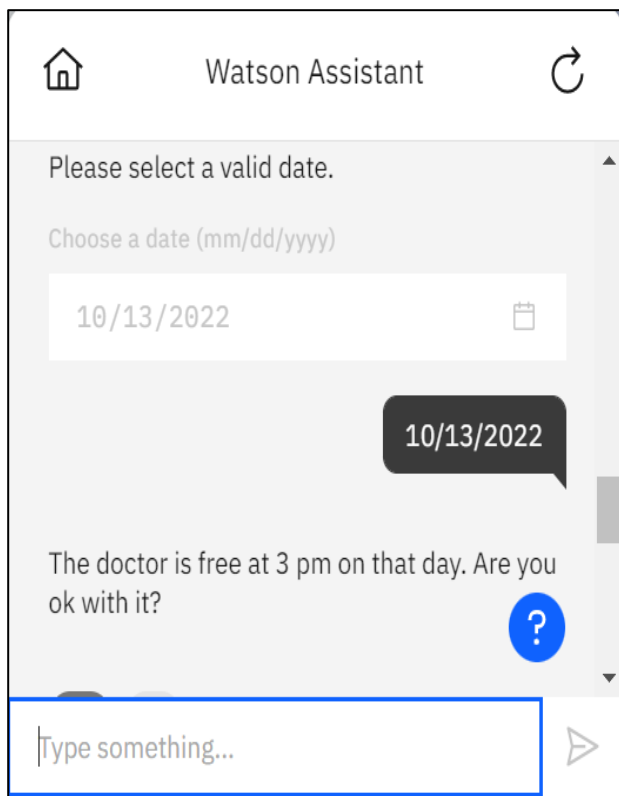
Chatbot Access URL

<https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Fjp-tok.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-c1c094d9-c048-4877-b09e-202b2a623fda%3A%3A28be94d0-4a06-4bb7-a0af-358479d049ca&integrationID=640b8a9b-167a-4301-8cf9-f956c3444e5f®ion=jp-tok&serviceInstanceID=c1c094d9-c048-4877-b09e-202b2a623fda>

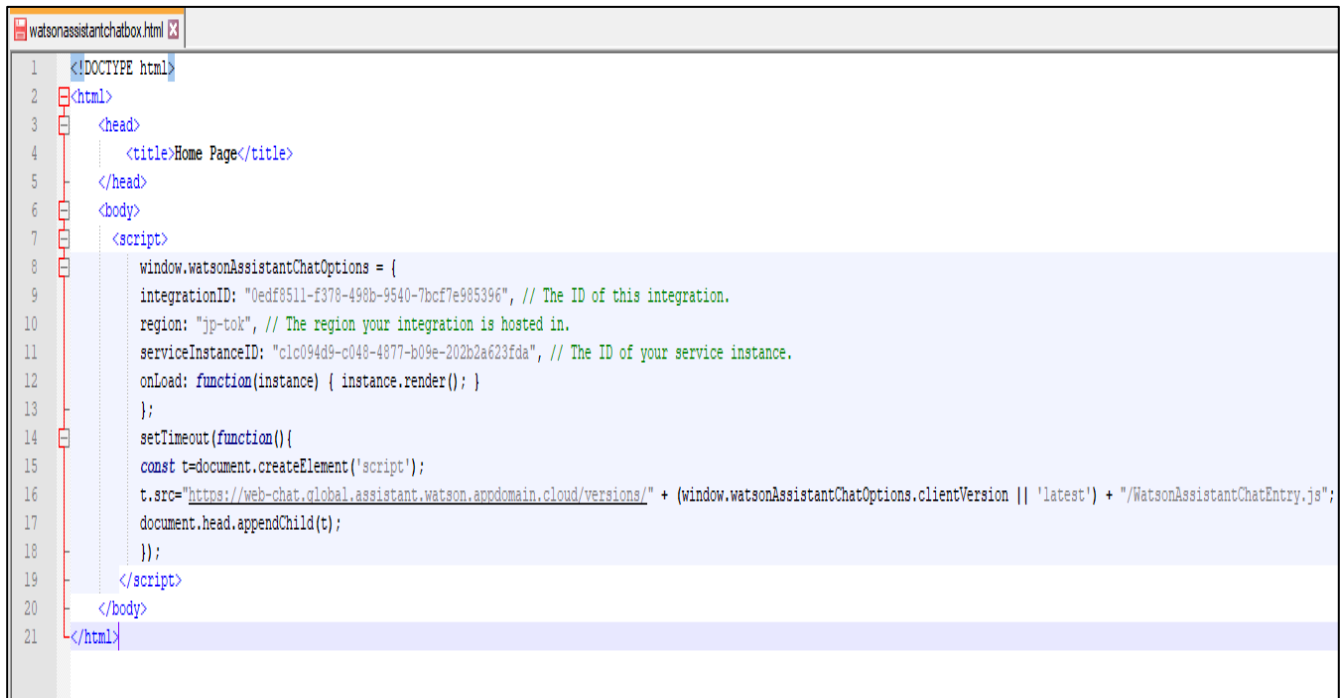
5. Create a Watson assistant service with ten steps and use three conditions in it. Load that script in the html page.







Watsonassistantchatbot.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://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Fjp-tok.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-c1c094d9-c048-4877-b09e-202b2a623fda%3A%3A970fcc61-3fe8-4aa3-962b-7c9d376ceb12&integrationID=0edf8511-f378-498b-9540-7bcf7e985396®ion=jp-tok&serviceInstanceID=c1c094d9-c048-4877-b09e-202b2a623fda>