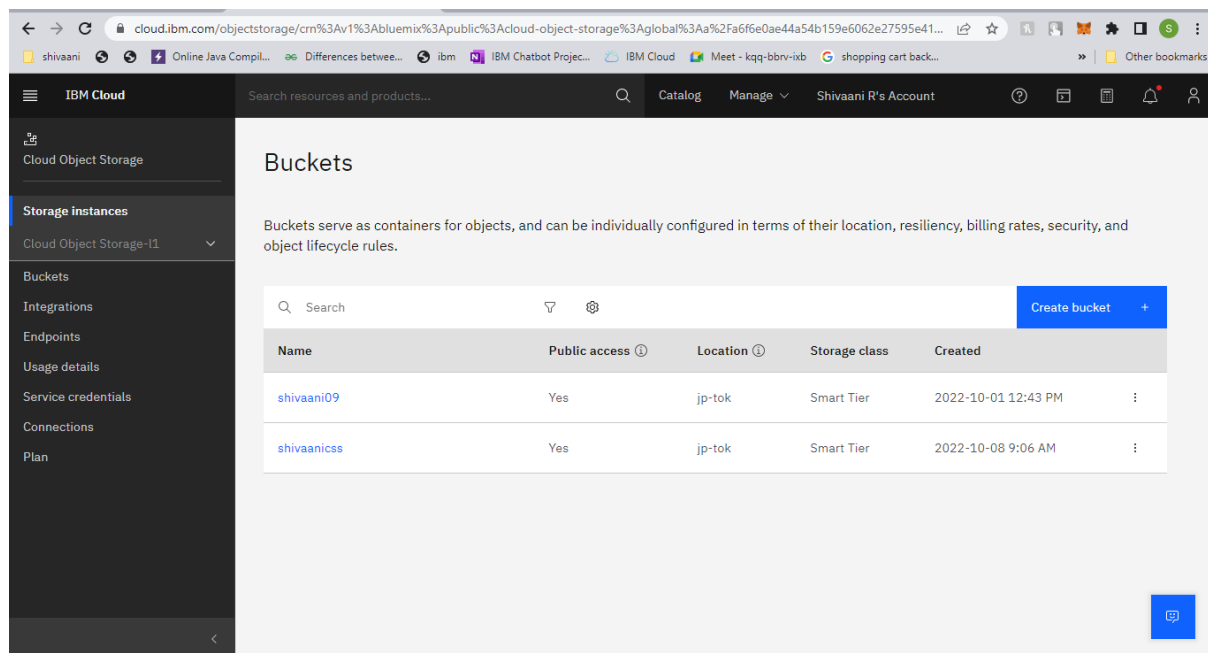


Assignment-3

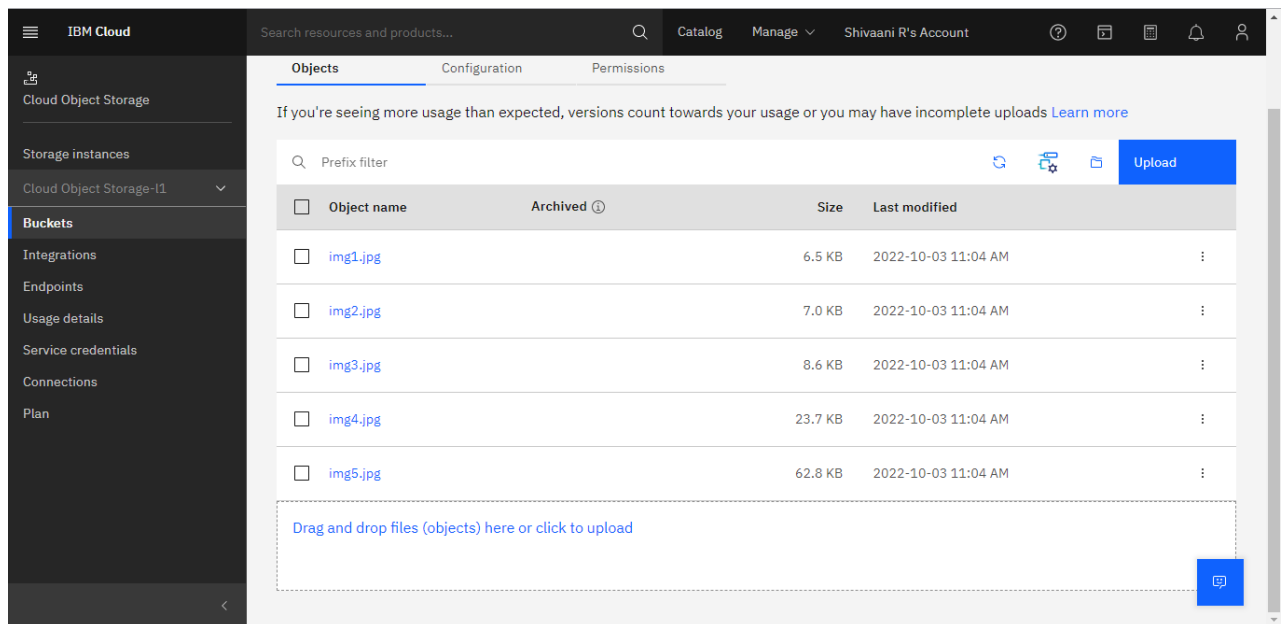
Assignment Date	26 September 2022
Student Name	Shivaani .R
Student Roll Number	813819104092
Maximum Marks	2 Marks

1.Create a Bucket in IBM object storage.



Thus a bucket is created with unique name.

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



main.py

```
from flask import Flask, render_template, request, redirect, url_for
from connect import *
from objstorage import *
app = Flask(__name__)
```

```
@app.route('/redirect_to')
def redirect_to():
    link = request.args.get('link', '/')
    return redirect(link), 301
```

```
@app.route("/assignment-3")
def objstorage():
    return render_template('assignment-3.html')
```

assignment-3.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
```

```

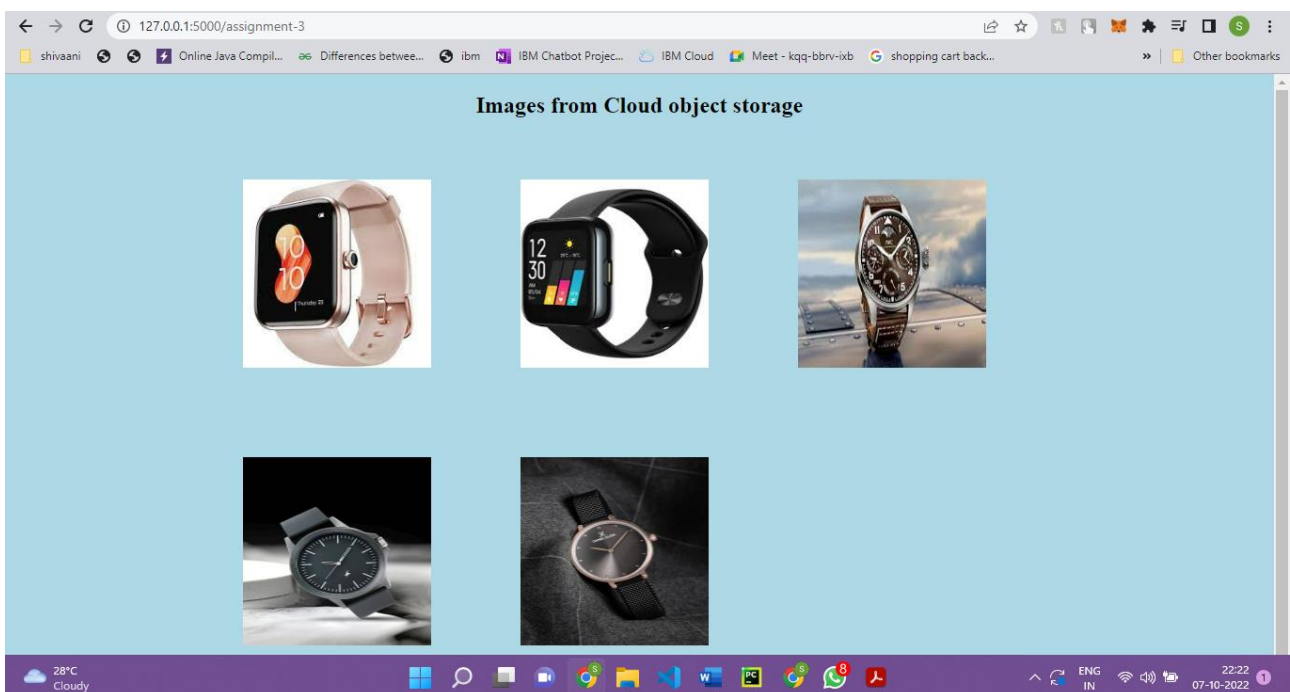
<title>Assignment-3</title>
<link rel="stylesheet" href="{{url_for('redirect_to',link='https://s3.jp-tok.cloud-
object-storage.appdomain.cloud/shivaanicss/assignment-3.css')}}"
type="text/css">
</head>
<body>
<h2> Images from Cloud object storage</h2>
<div class="imgd">



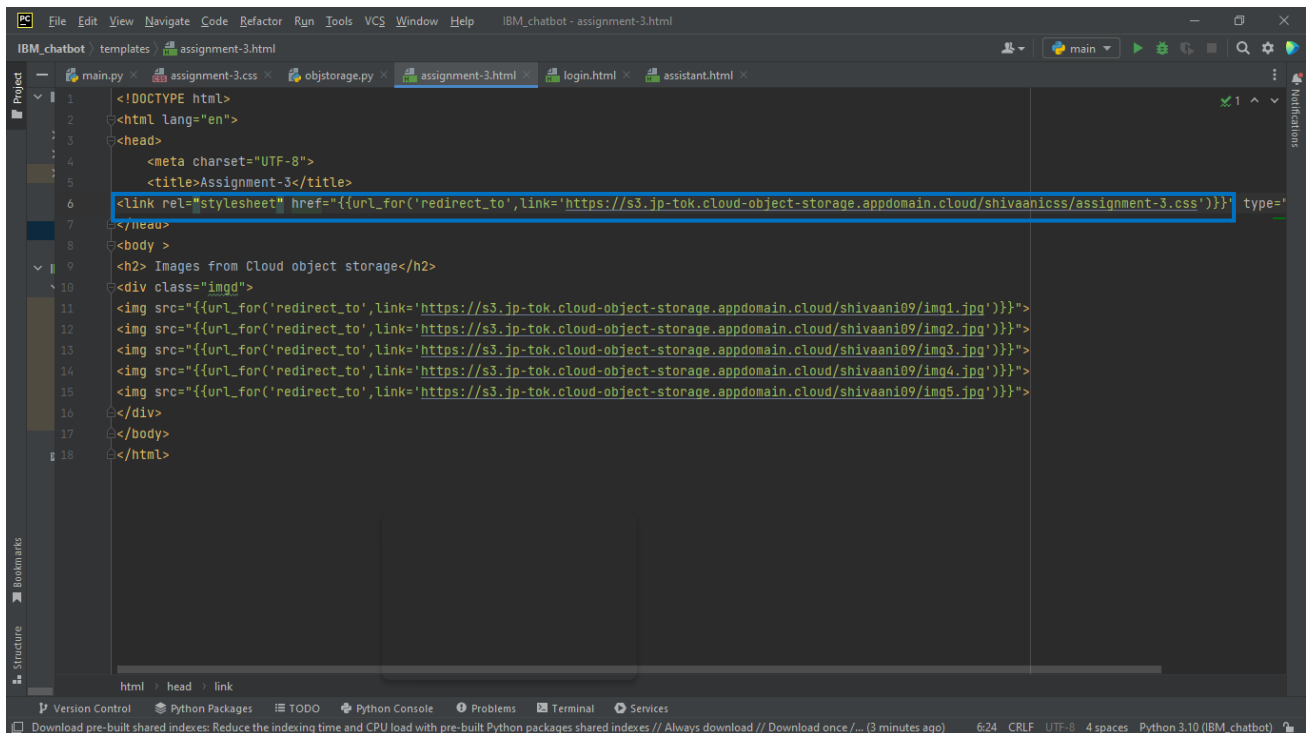
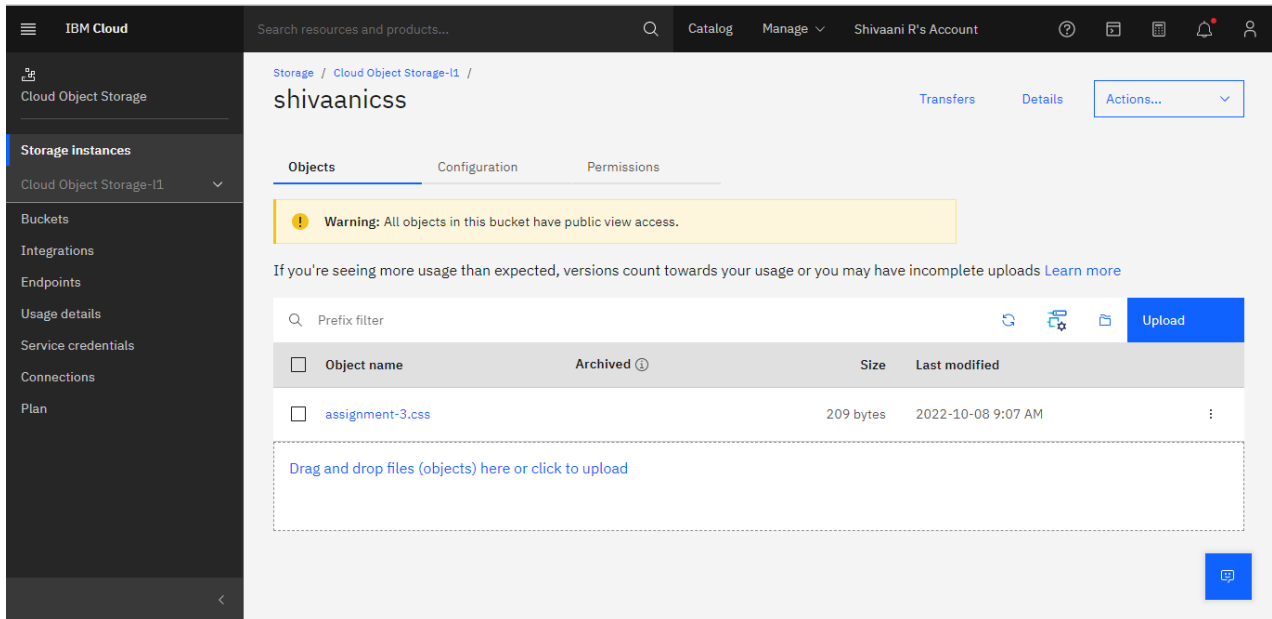


</div>
</body>
</html>

```



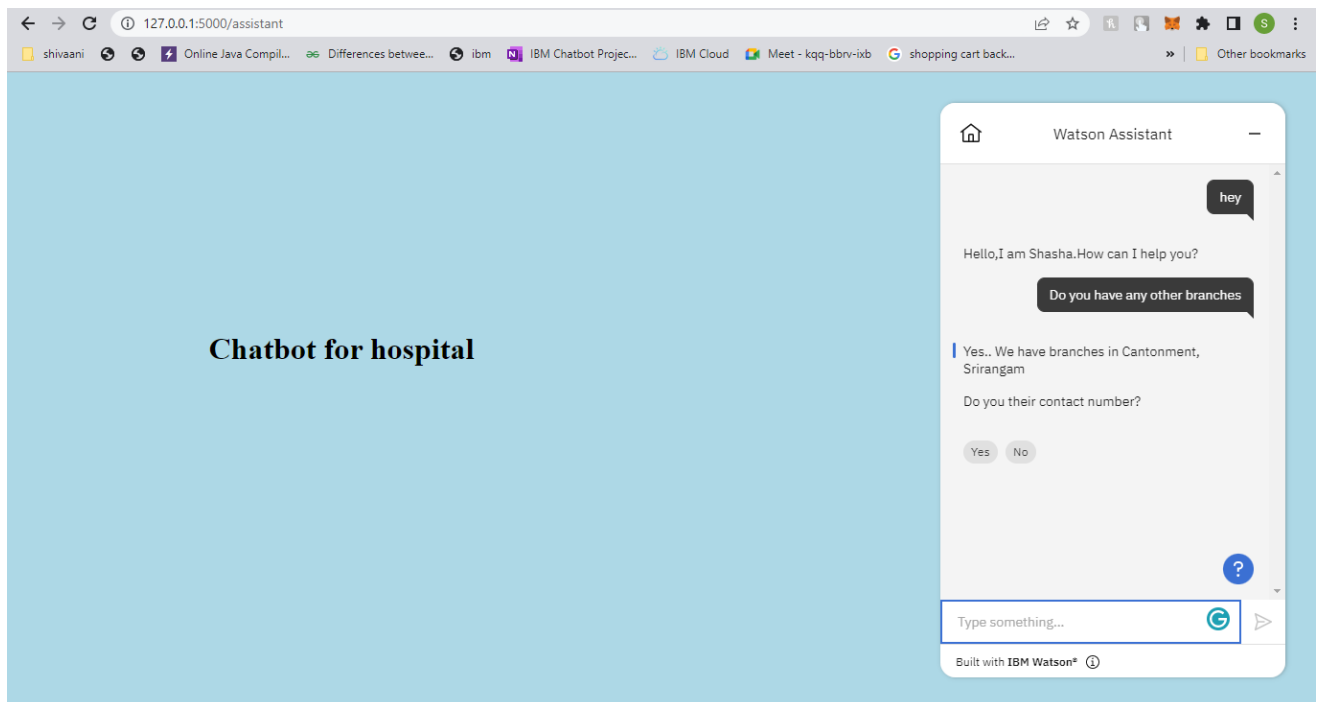
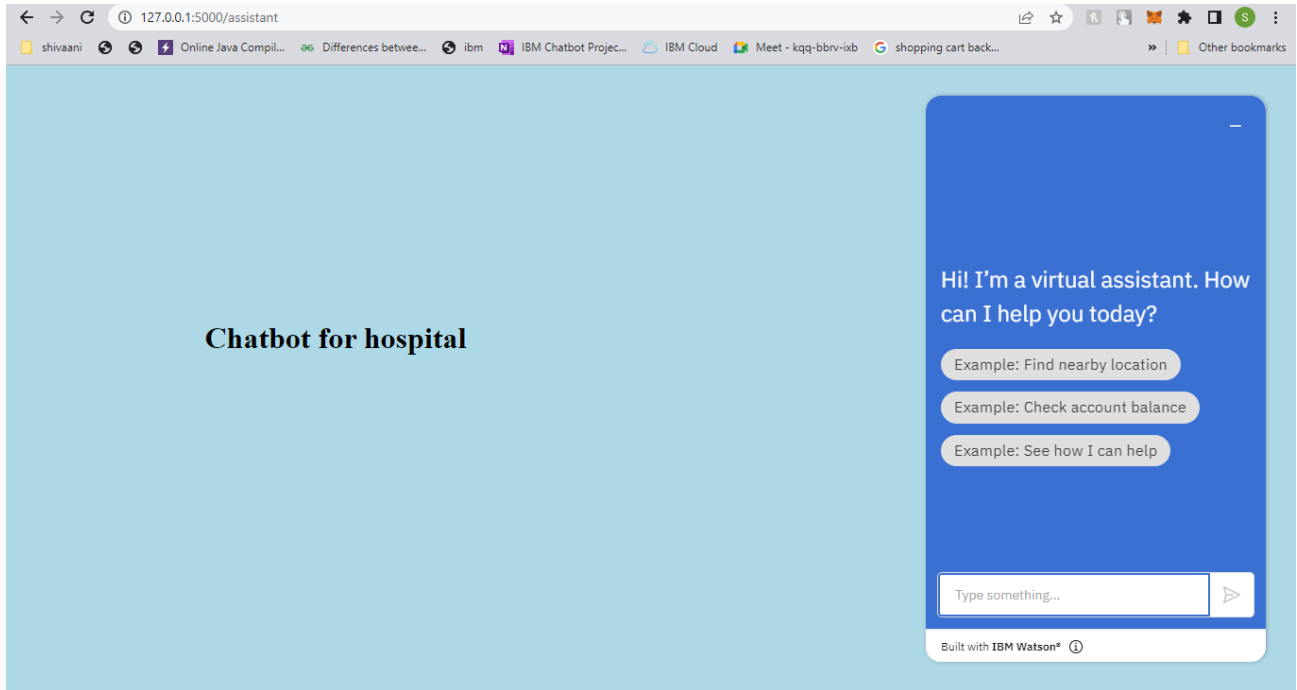
3.Upload a css page to the object storage and use the same page in your HTML code.

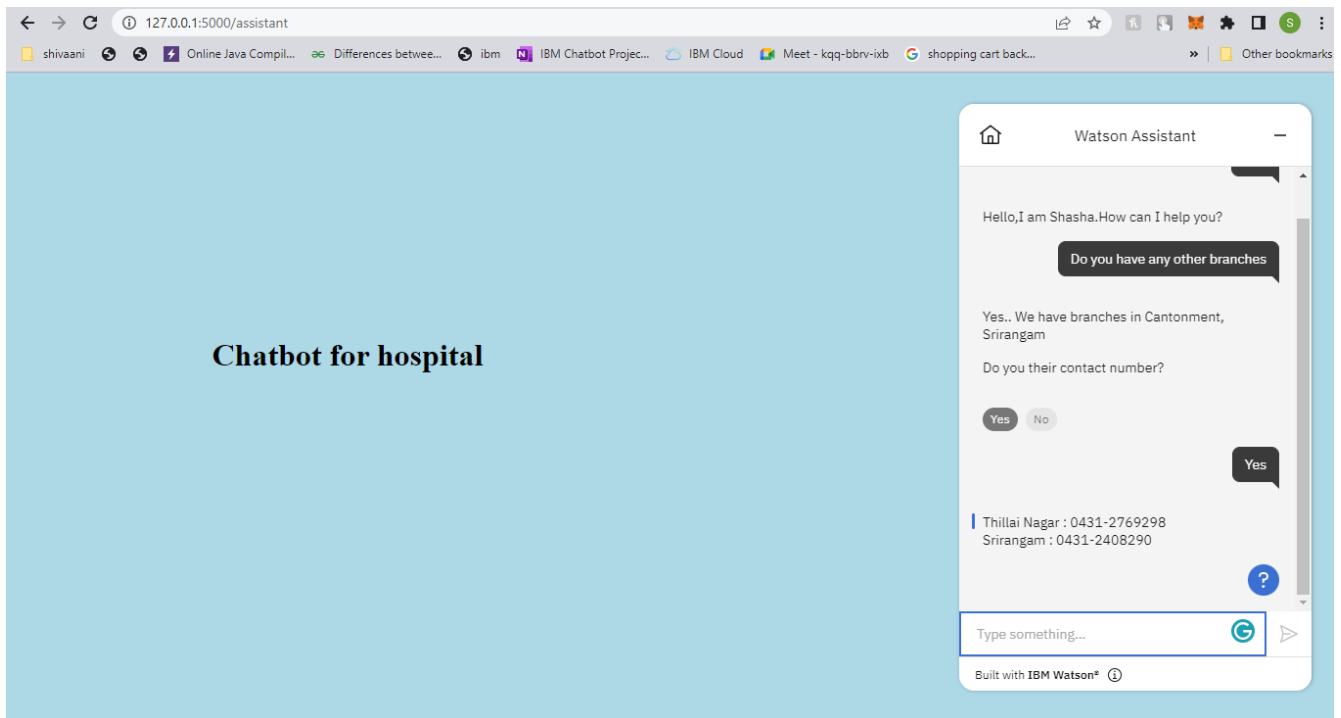


The same css file is used in html code.

4.Design a chatbot using IBM Watson assistant for hospital.

User comes with query to know the branches for that hospital in your city



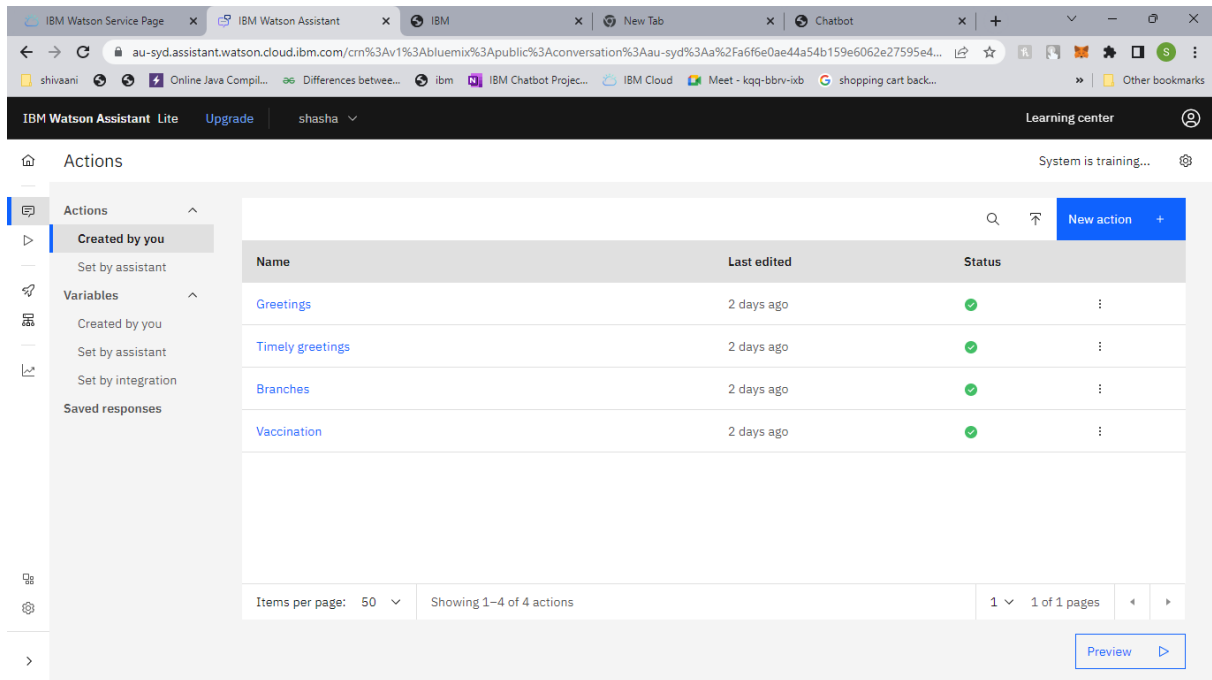


Web URL of assistant

```
window.watsonAssistantChatOptions = {  
  integrationID: "975ba6f0-d5c0-4f04-b1f4-5870b7074a7e", // The ID of this  
  integration.  
  region: "au-syd", // The region your integration is hosted in.  
  serviceInstanceID: "7c599db5-022d-4610-afe5-57274f0e14ac", // 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);  
});
```

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

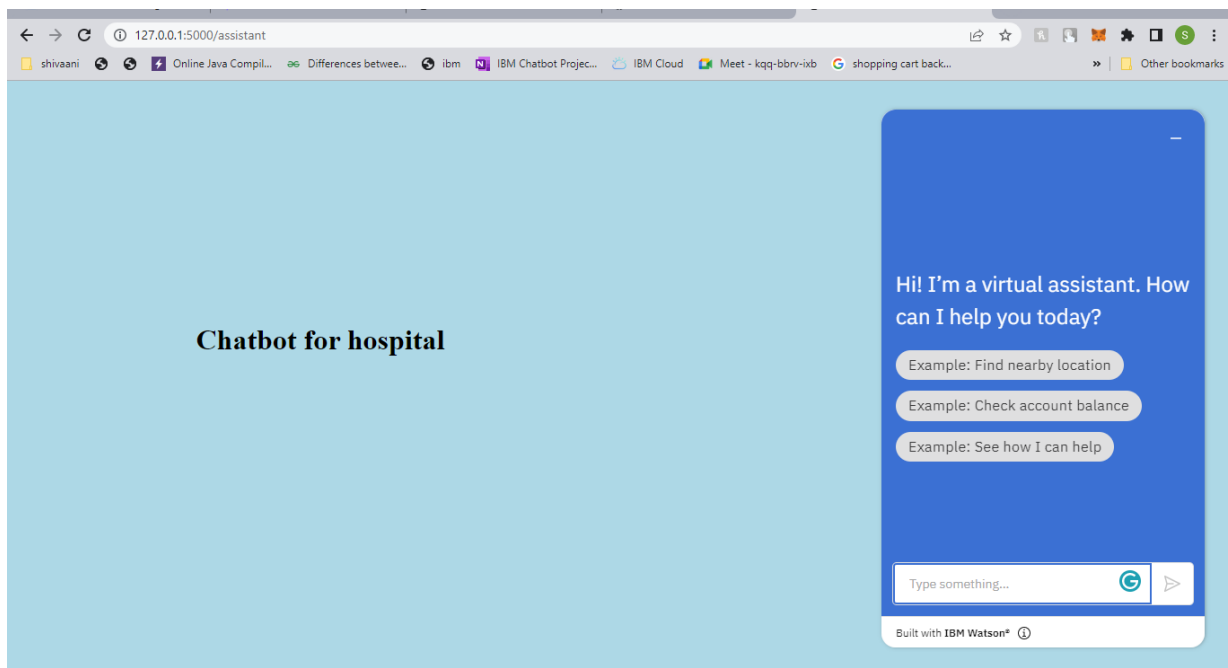
Scenario: User inquires about vaccination details.



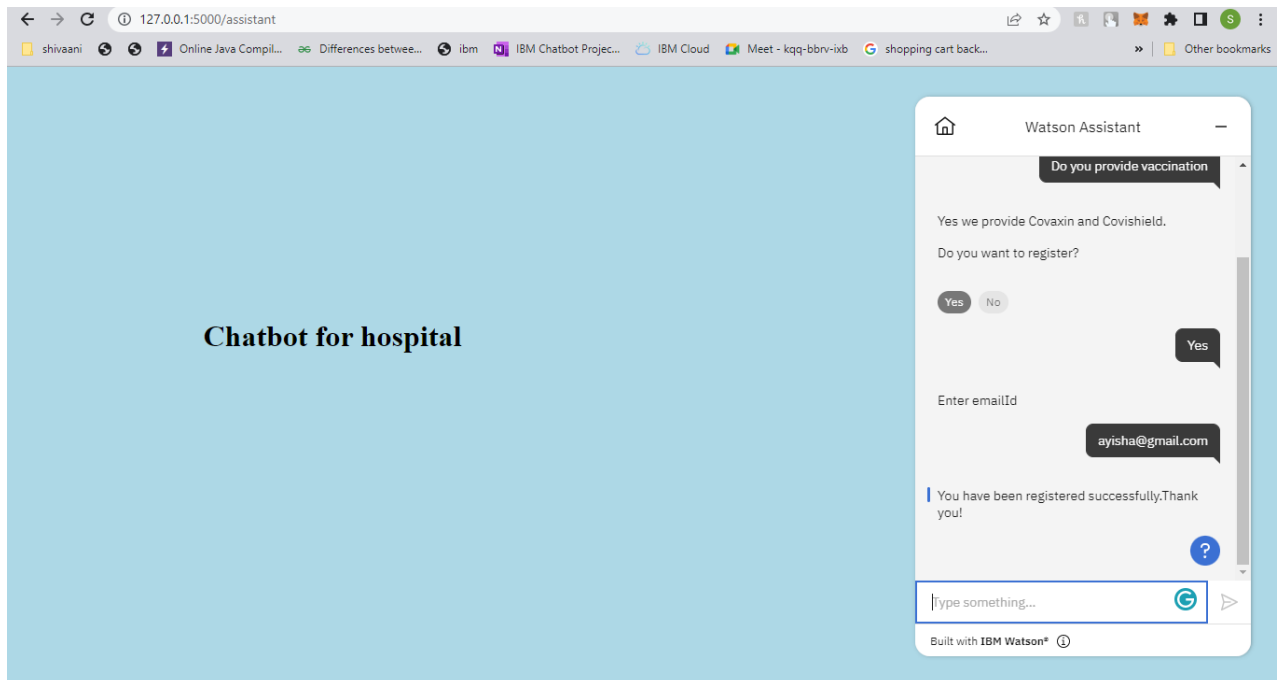
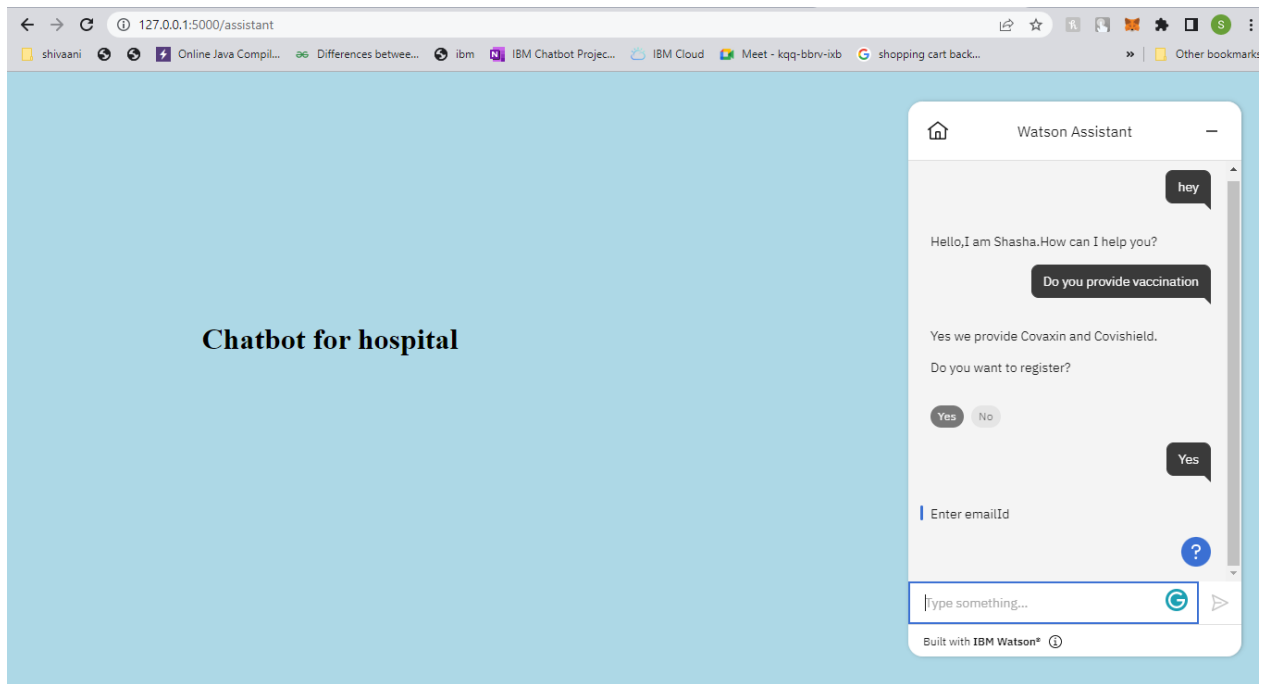
The screenshot shows the IBM Watson Assistant 'Actions' page. The left sidebar contains a menu with 'Actions' selected, which is further divided into 'Created by you', 'Set by assistant', 'Variables', 'Set by integration', and 'Saved responses'. The main area displays a table of actions created by the user.

Name	Last edited	Status
Greetings	2 days ago	✓
Timely greetings	2 days ago	✓
Branches	2 days ago	✓
Vaccination	2 days ago	✓

At the bottom of the table, it says 'Items per page: 50' and 'Showing 1–4 of 4 actions'. A 'Preview' button is located at the bottom right of the table area.



The screenshot shows a chatbot interface titled 'Chatbot for hospital'. The chatbot's message is: 'Hi! I'm a virtual assistant. How can I help you today?'. Below the message are three example prompts: 'Example: Find nearby location', 'Example: Check account balance', and 'Example: See how I can help'. At the bottom, there is a text input field with the placeholder 'Type something...' and a green circular button with a white 'G' icon. The footer of the chatbot interface says 'Built with IBM Watson®'.



Watson assistant is given with 3 conditions .

The screenshot displays the IBM Watson Assistant interface in a web browser. The browser's address bar shows the URL: `au-syd.assistant.watson.cloud.ibm.com/cm%3Av1%3Abluemix%3Apublic%3Aconversation%3Aau-syd%3Aa%2Fa6f6e0ae44a54b159e6062e27595e4...`. The browser tabs include 'IBM Watson Service Page', 'IBM Watson Assistant', 'IBM', 'New Tab', and 'Chatbot'. The page header features 'IBM Watson Assistant Lite', an 'Upgrade' button, a user profile 'shasha', and a 'Learning center' link.

The main content area is titled 'Timely greetings'. It is divided into two panels. The left panel, 'Conversation steps', shows a sequence of steps for a conversation. Step 1 is 'Good morning, How can I help you?' with a 'Continue to next step' button. Step 2 is 'Good afternoon, How can I help?' with a 'Continue to next step' button. A 'New step +' button is at the bottom. The right panel, 'Customer starts with:', provides instructions on how to define starting phrases. It includes a text input field 'Enter a phrase' and a list of example phrases: 'good evening', 'good afternoon', and 'good morning'. A 'Preview' button is located at the bottom right of the right panel.

Conversation steps

- 1. Good morning, How can I help you?
Continue to next step
- 2. Good afternoon, How can I help?
Continue to next step

Customer starts with:

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 4

Enter a phrase

- good evening
- good afternoon
- good morning

Preview