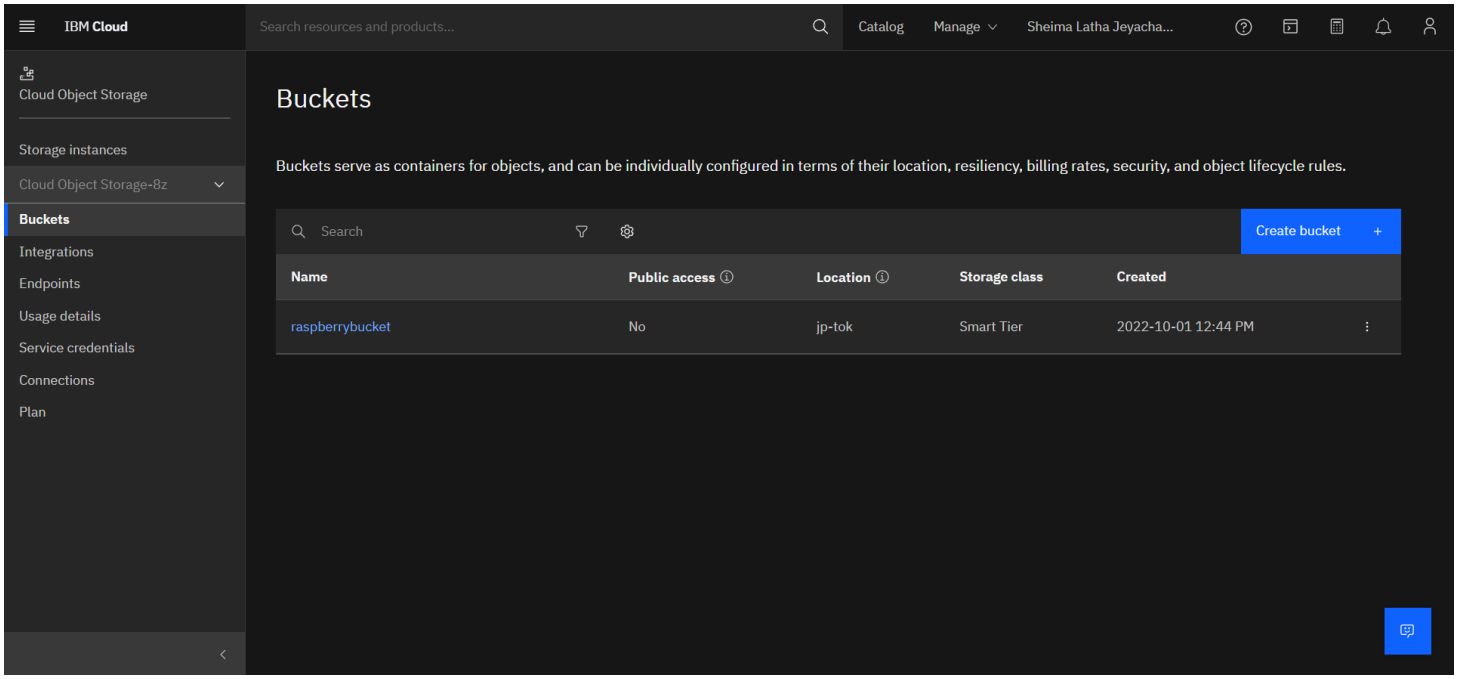
ASSIGNMENT – 3

CLOUD APPLICATION DEVELOPMENT

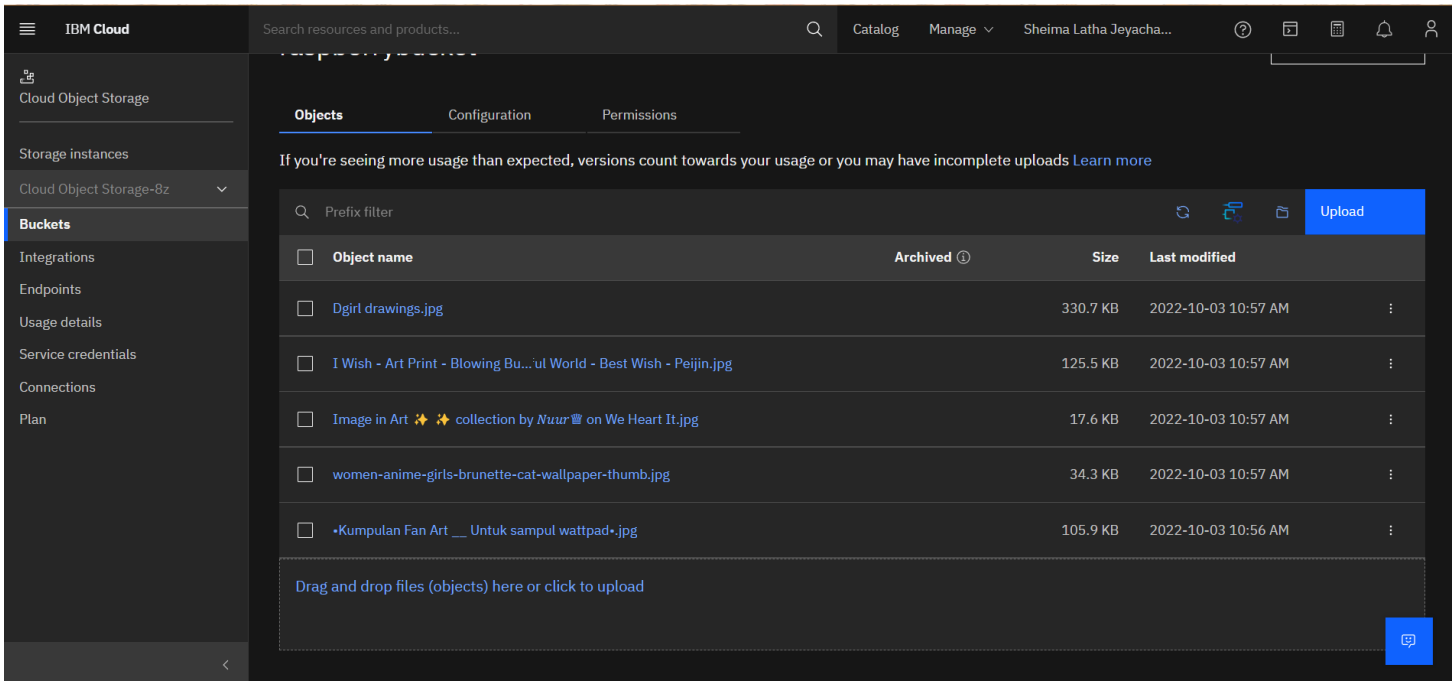
Date	26 September 2022
Student Name	Sheima Latha. J
Student Roll Number	813819104090
Maximum Marks	2 Marks

1. **Create a bucket in IBM object storage.**



Thus, a bucket named “raspberrybucket” has been created.

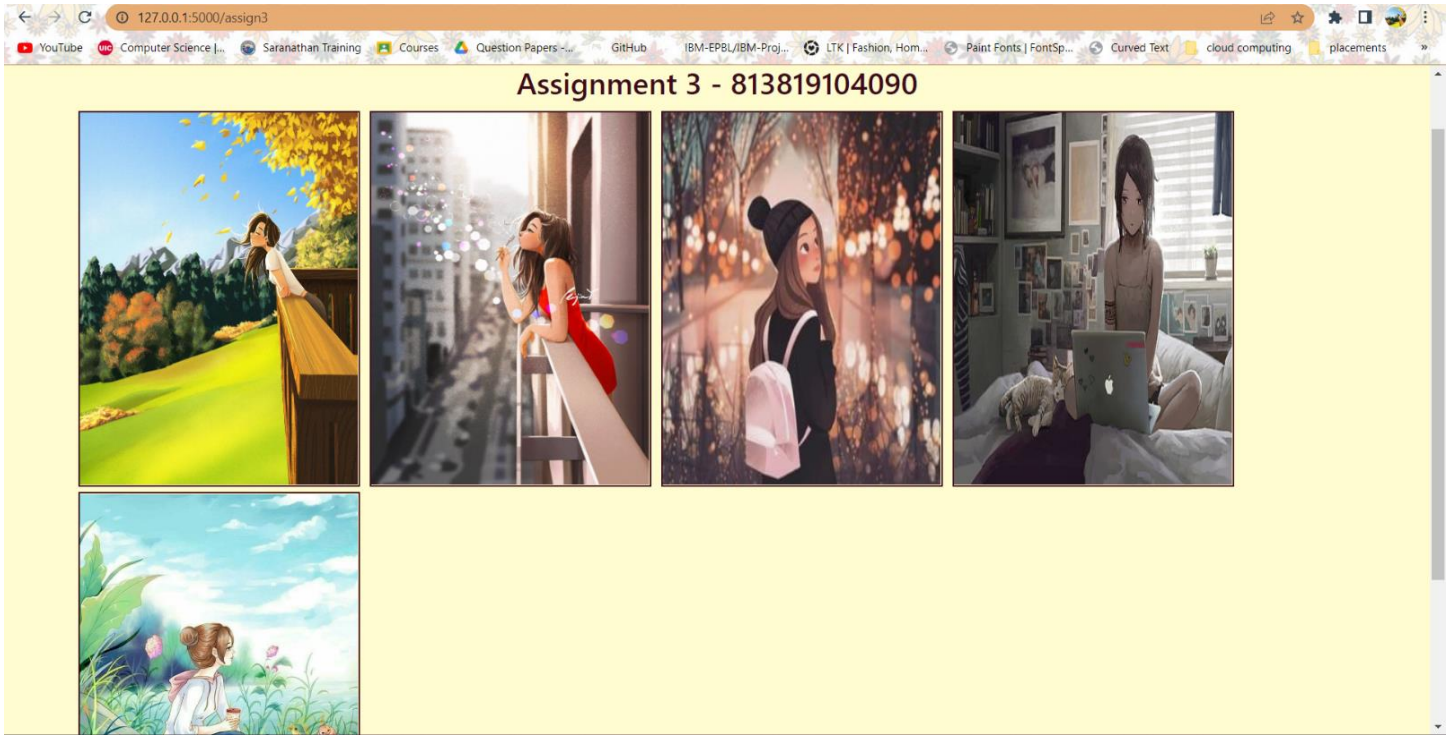
2. **Upload 5 images to IBM Object Storage and make it public. Write an HTML code to display all the 5 images.**



HTML Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Assignment 3</title>
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css" integrity="sha384-9aIt2nRpC12Uk9gS9baDl4
    <link rel="stylesheet" type="text/css" href="{{url_for('static',filename='assign3styles.css')}}"/>
</head>
<body>
  <h2>Assignment 3 - 813819104090</h2>
  <img class="imgdiv" src="{{url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/raspberrybucket/Dgirl drawings.jpg'
  <img class="imgdiv" src="{{url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/raspberrybucket/I Wish - Art Print
  <img class="imgdiv" src="{{url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/raspberrybucket/Image in Art
  <img class="imgdiv" src="{{url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/raspberrybucket/women-anime-girls-b
  <img class="imgdiv" src="{{url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/raspberrybucket/·Kumpulan Fan Art _
</body>
</html>
```

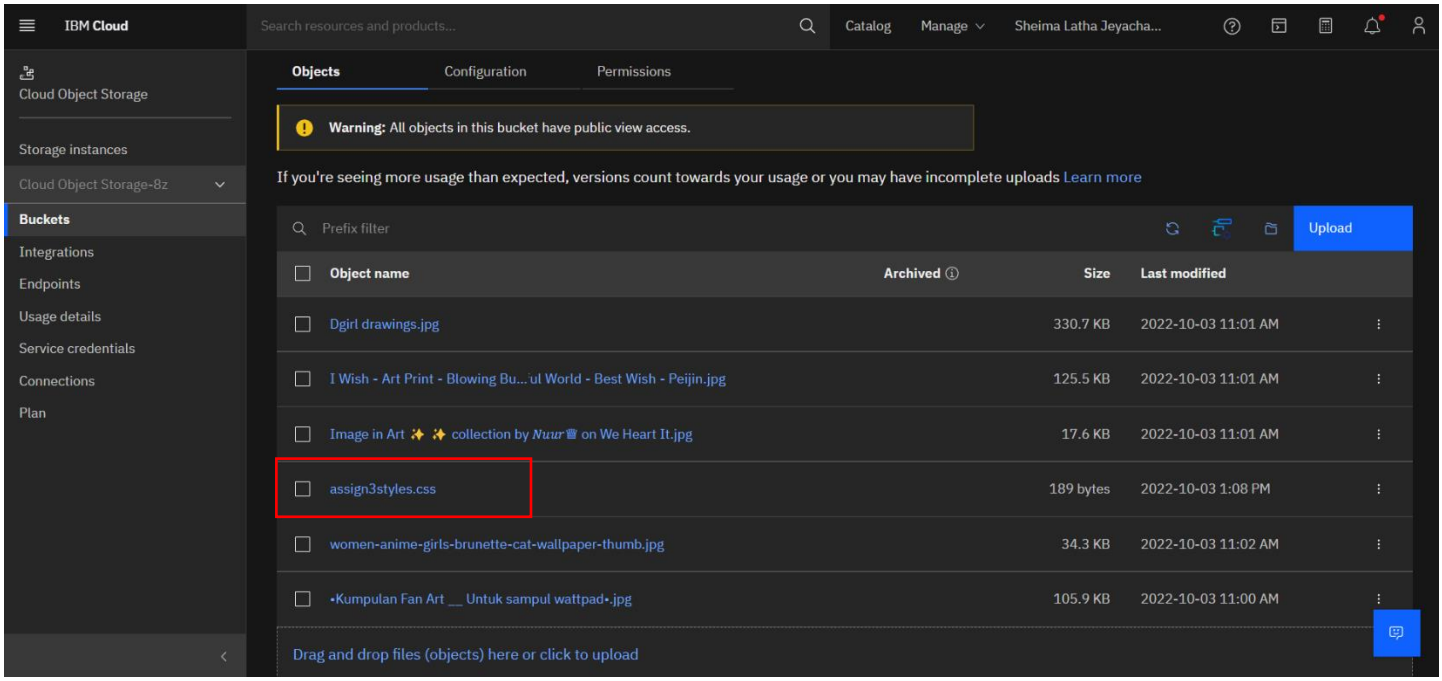
Output:



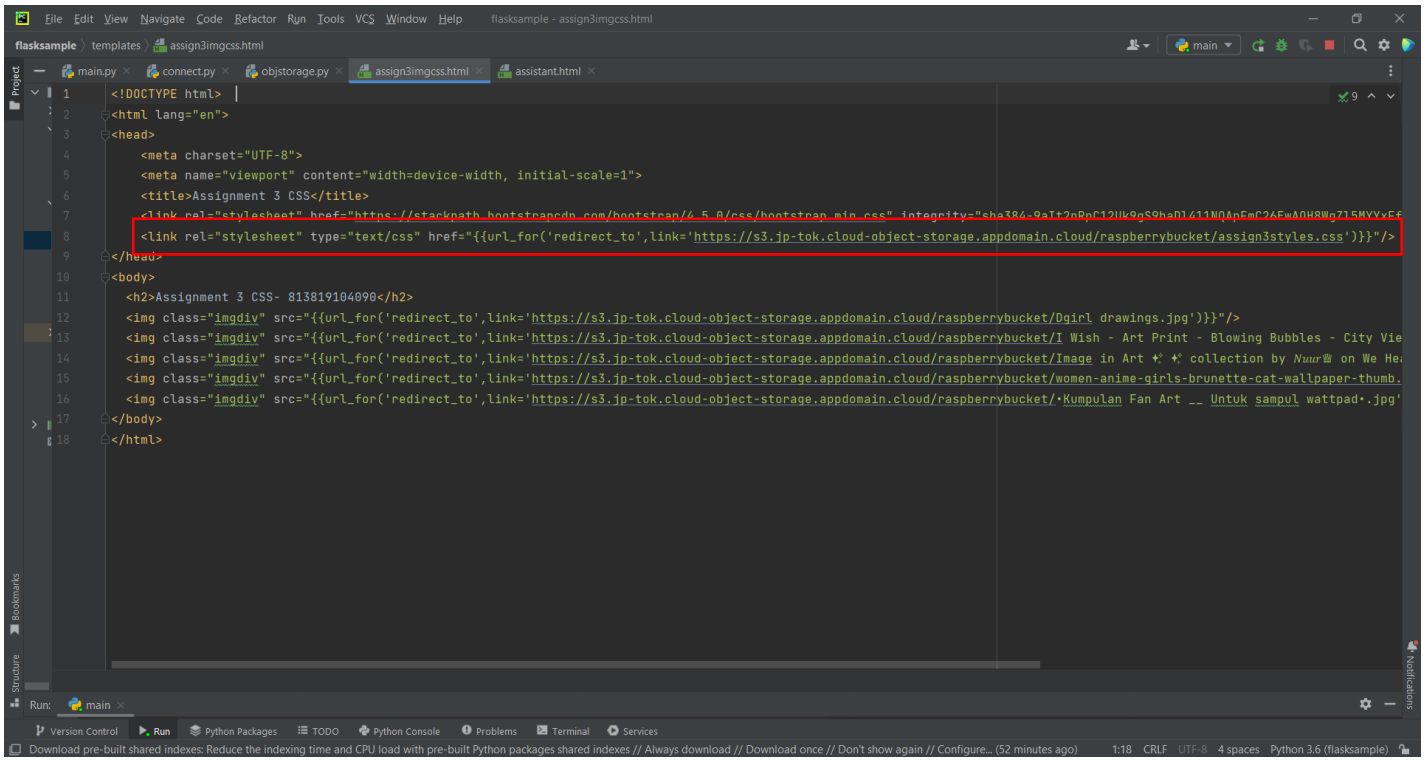
File name: assign3img.html, main.py, objstorage.py

The above mentioned files have been uploaded in GitHub.

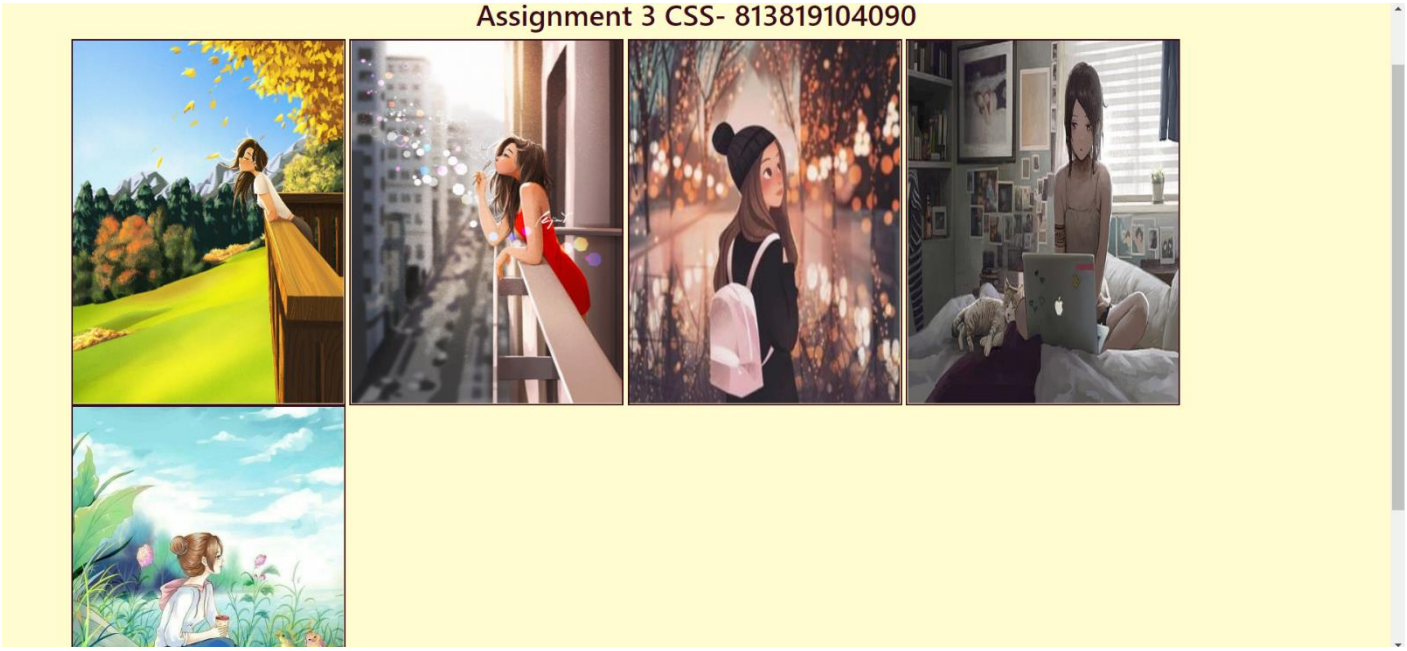
3. **Upload a CSS page to the object storage and use the same page in your HTML code.**



HTML Code:



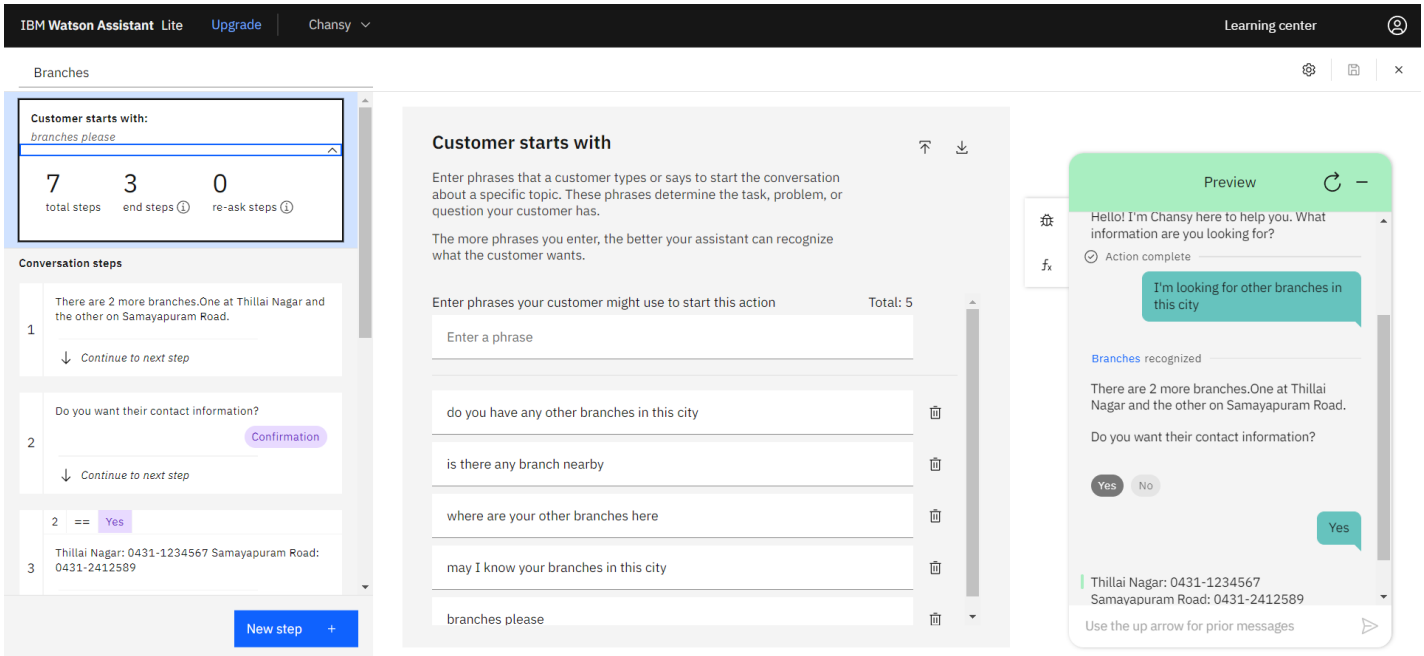
Output:

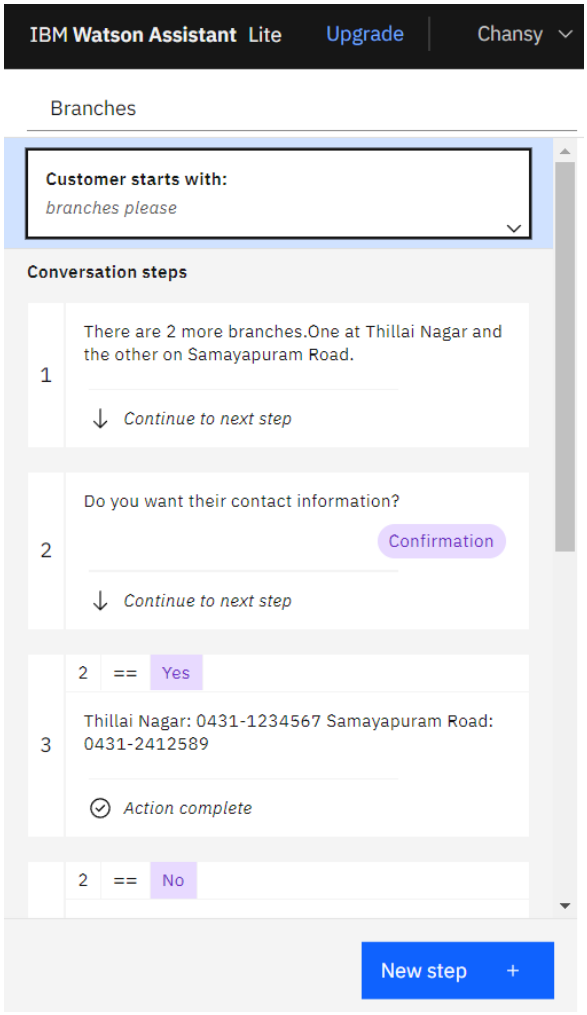


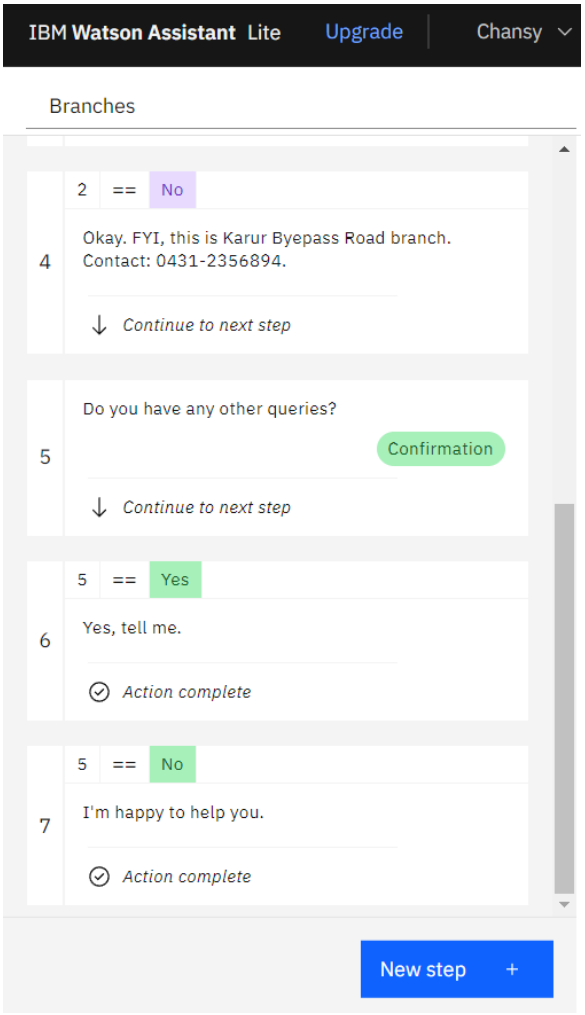
File name: assign3imgcss.html , assign3styles.css, main.py, objstorage.py

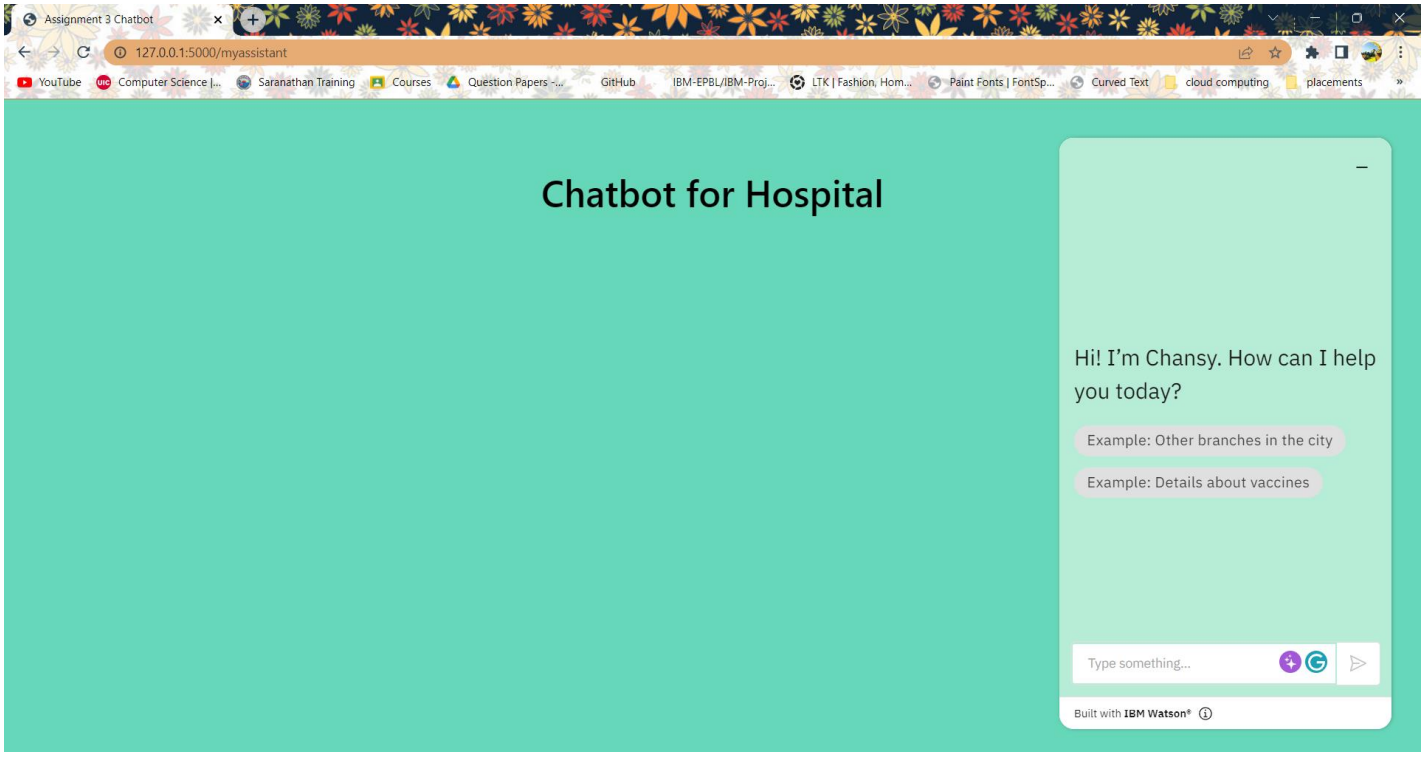
The files have been uploaded in GitHub.

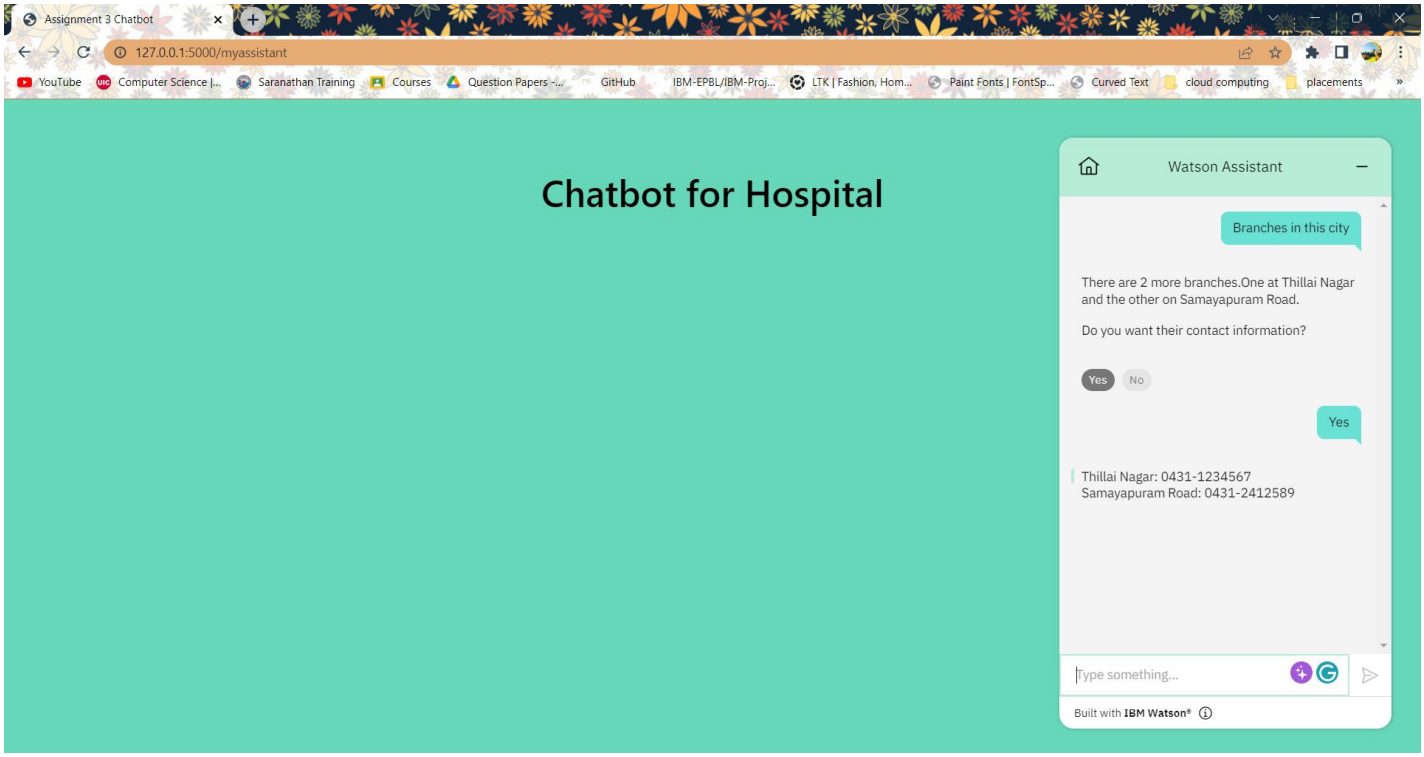
4. **Design a chatbot using IBM Watson assistant for hospital. For e.g., user comes with query to know the branches for that hospital in your city. Submit the web URL of that chatbot as an assignment.**




IBM Watson Assistant Lite
Upgrade
Chansy
Branches
Customer starts with:
branches please
Conversation steps
1
There are 2 more branches.One at Thillai Nagar and the other on Samayapuram Road.
Continue to next step
2
Do you want their contact information?
Confirmation
Continue to next step
3
2 == Yes
Thillai Nagar: 0431-1234567 Samayapuram Road: 0431-2412589
Action complete
2 == No
New step +


IBM Watson Assistant Lite
Upgrade
Chansy
Branches
4
2 == No
Okay. FYI, this is Karur Byepass Road branch.
Contact: 0431-2356894.
Continue to next step
5
Do you have any other queries?
Confirmation
Continue to next step
6
5 == Yes
Yes, tell me.
Action complete
7
5 == No
I'm happy to help you.
Action complete
New step +


Assignment 3 Chatbot
127.0.0.1:5000/myassistant
Chatbot for Hospital
Hi! I'm Chansy. How can I help you today?
Example: Other branches in the city
Example: Details about vaccines
Type something...
Built with IBM Watson®



File name: assistant.html, main.py

The files have been uploaded in GitHub.

Web URL:

```
<script>
    window.watsonAssistantChatOptions = {
    integrationID: "148c8b05-d7de-489c-a970-517c3676838f", // The ID of this
integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "efccfecd-0afd-4fd7-8f9e-8a9e819f7075", // 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>
```

main.py:

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

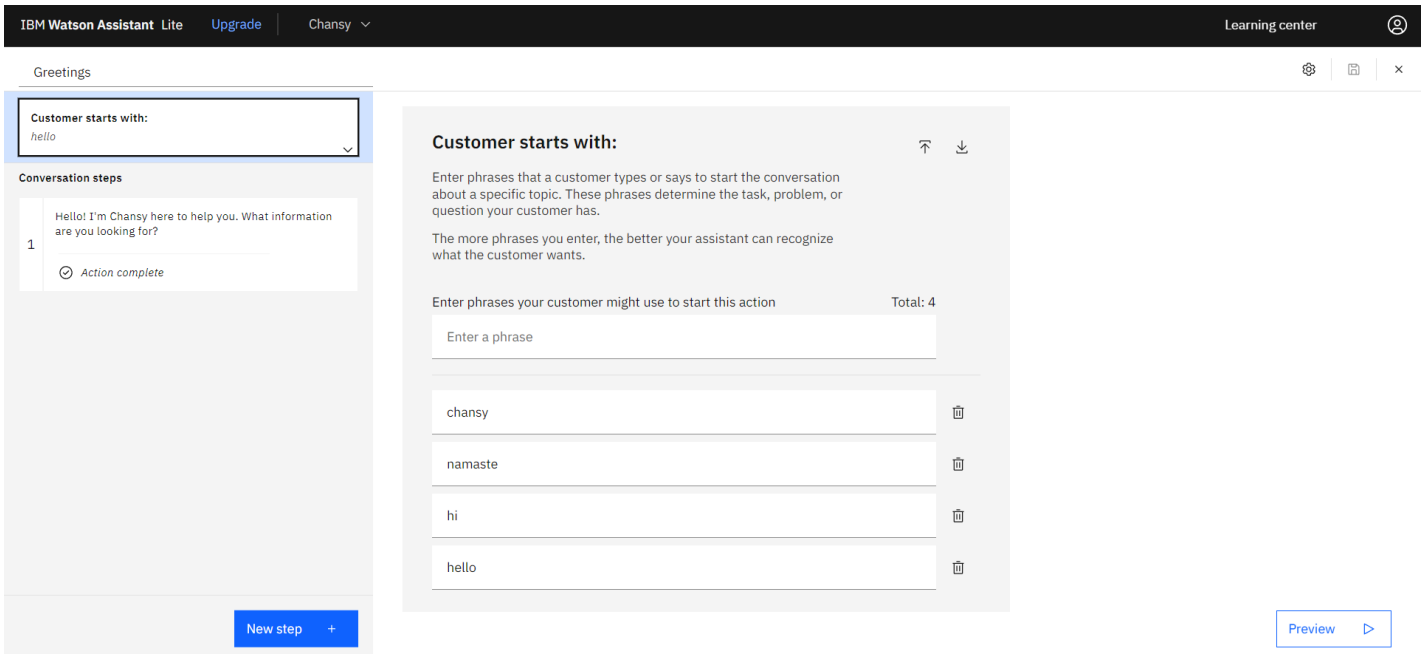
```
@app.route('/myassistant')
def assistant_page():
    return render_template('assistant.html')

if(__name__=='__main__'):
    app.run()
```

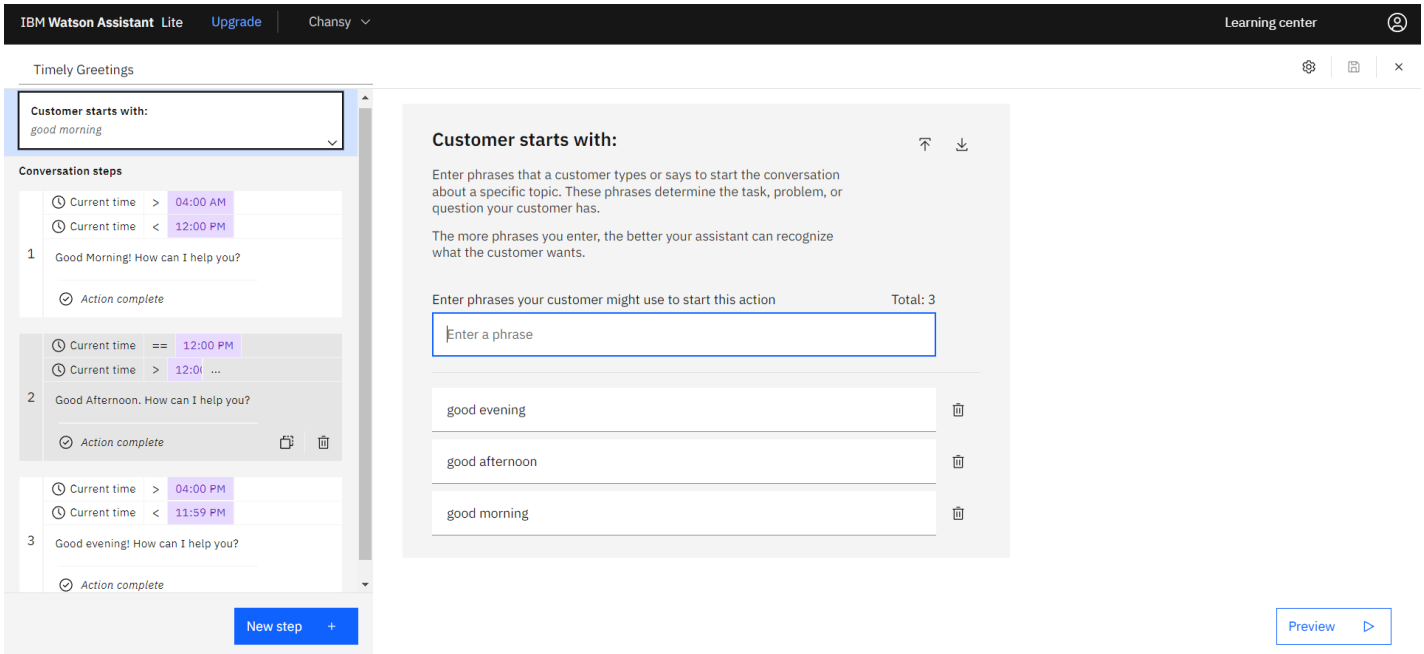
5. **Create Watson assistant service with 10 steps and use 3 conditions in it. Load that script in HTML page.**

I'm creating a chatbot for hospital.

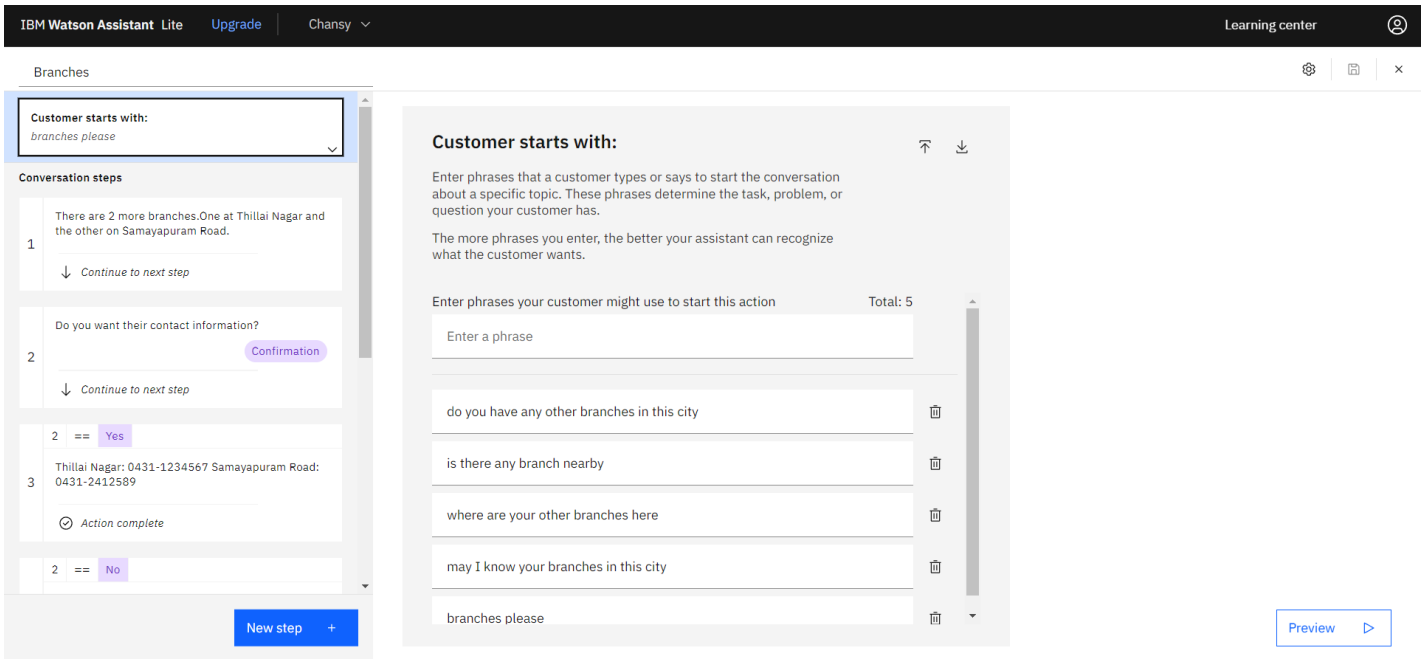
Actions: Greetings

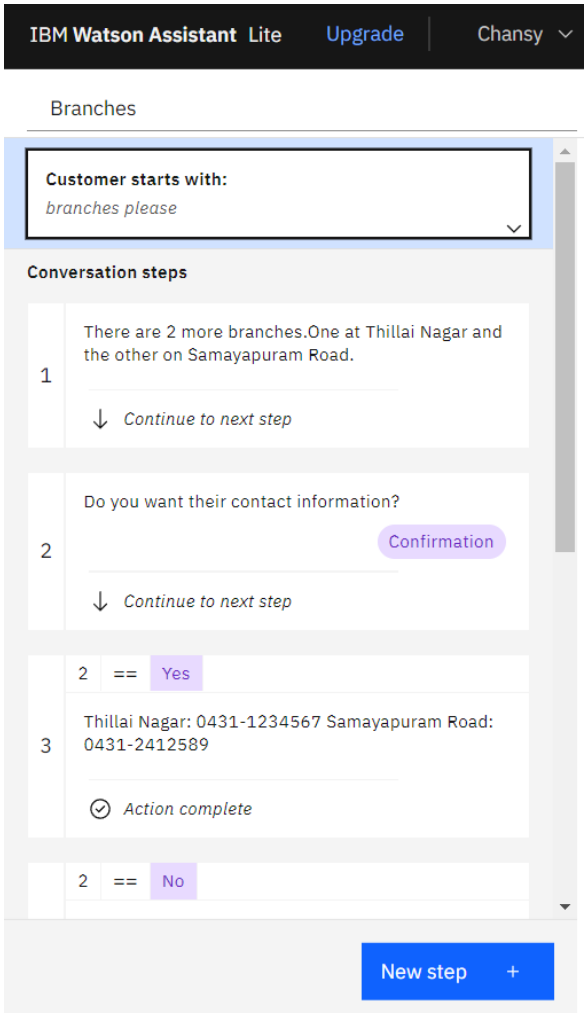


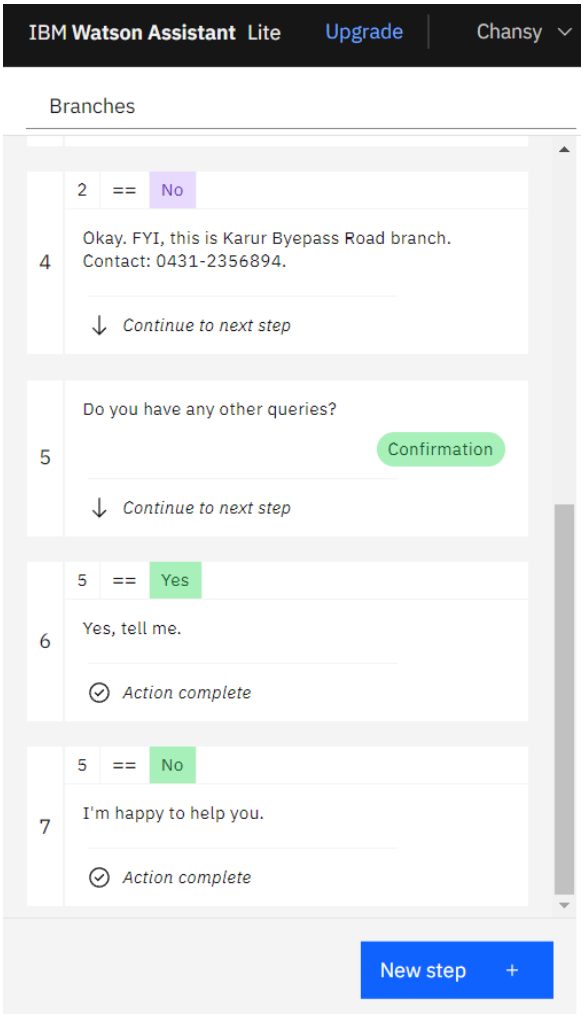
Actions: Timely Greetings



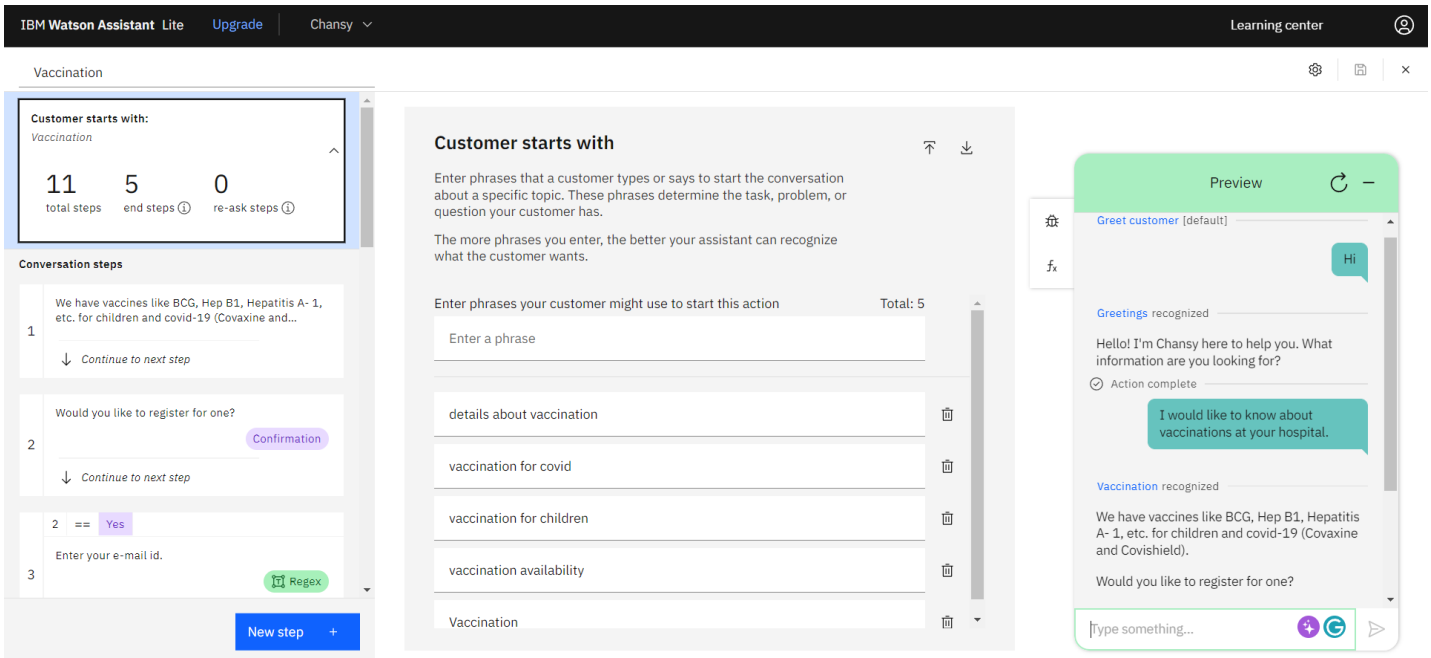
Actions: Branches

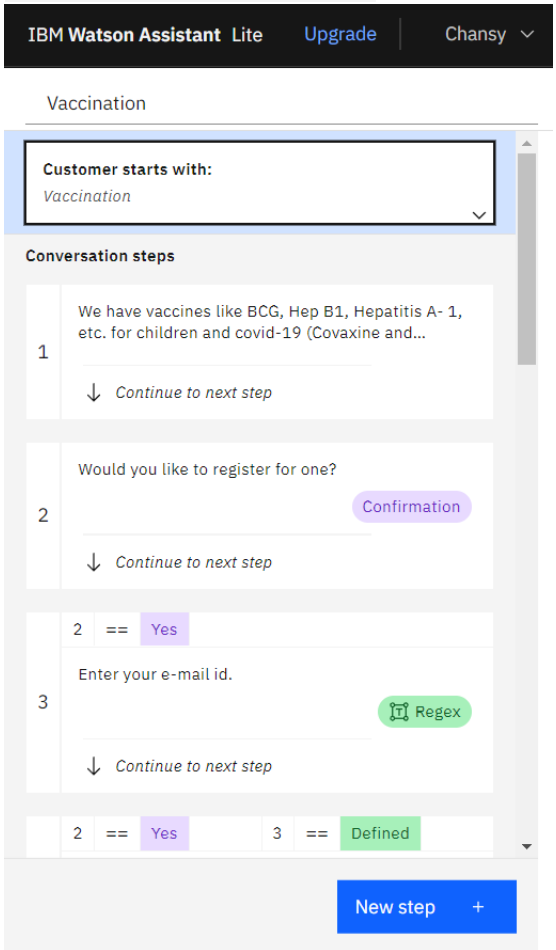


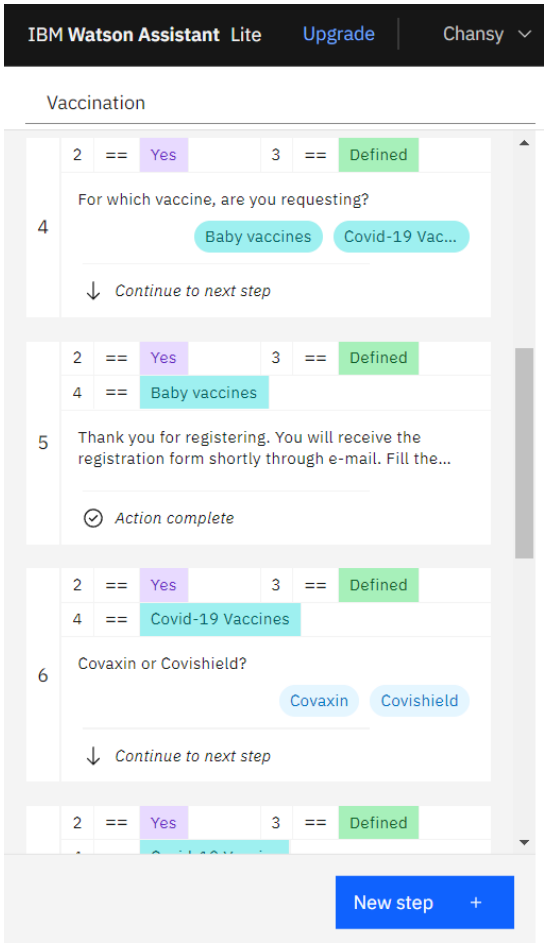


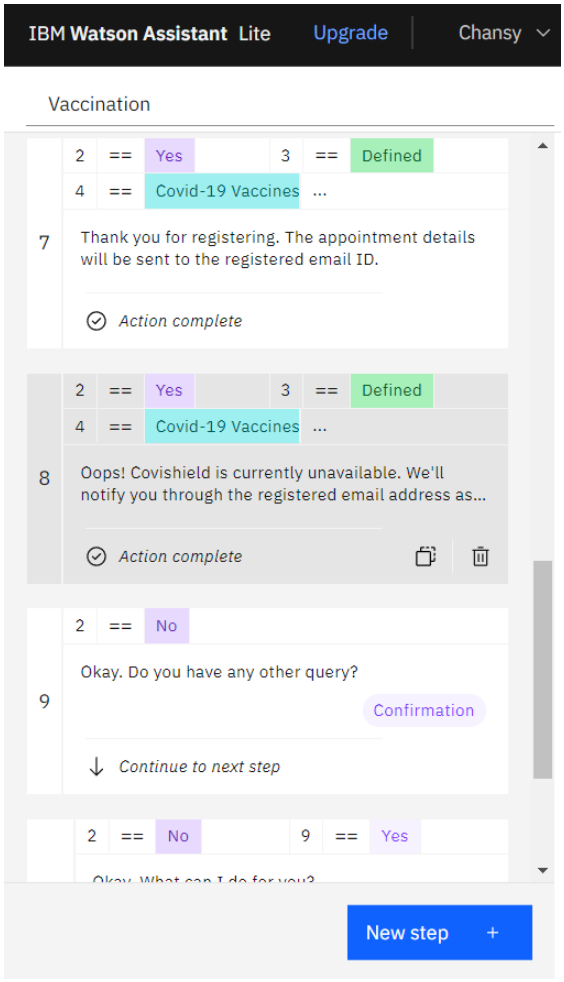


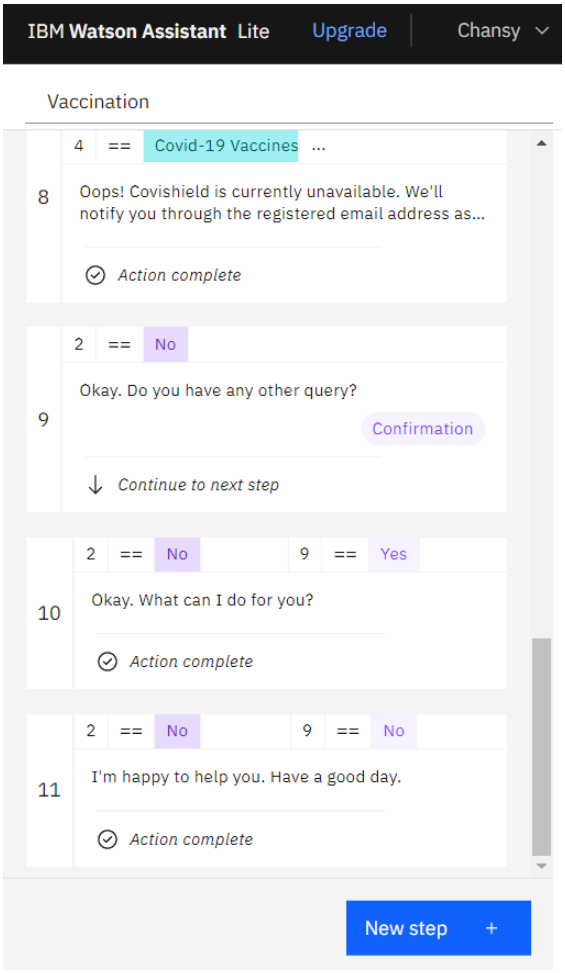
Action: Vaccination





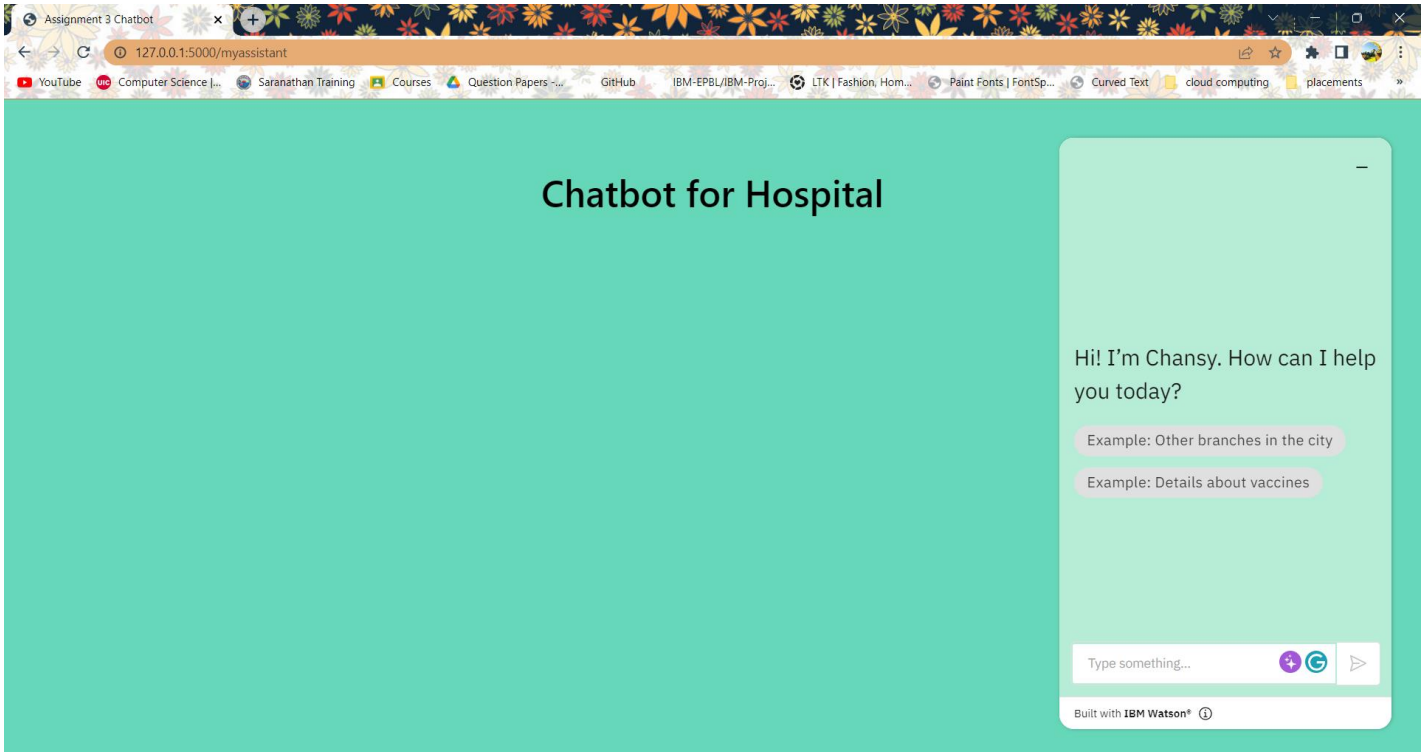


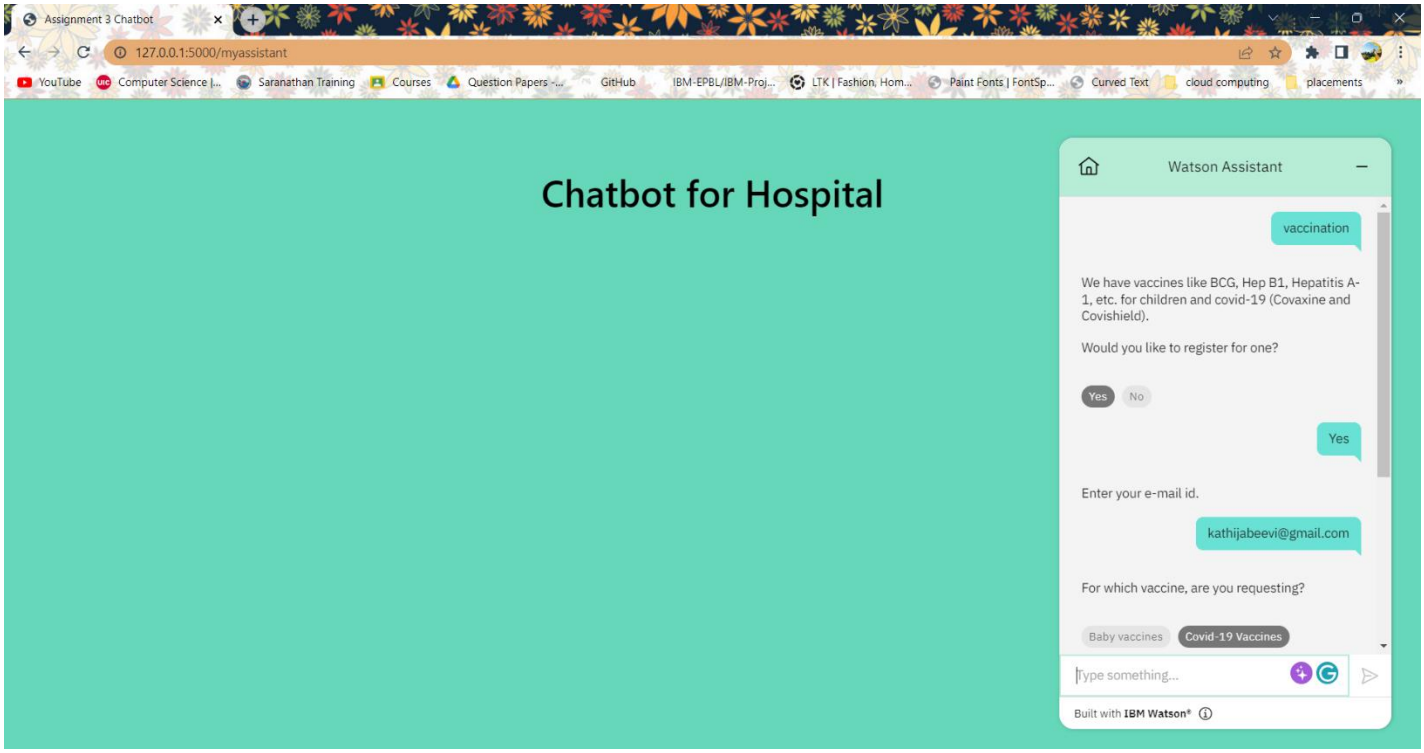


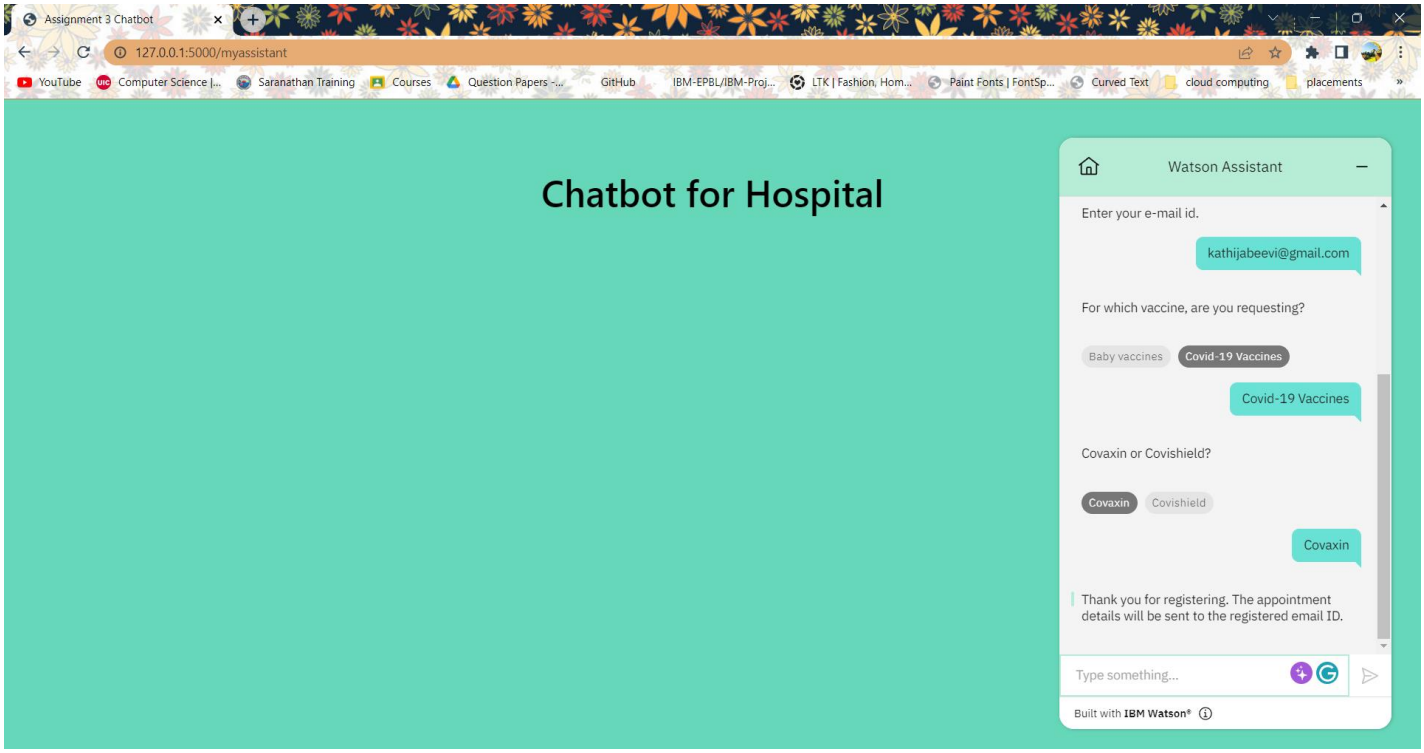


Sample Output:

Scenario: The user wants to know vaccination details at a hospital and register for it.







File name: assistant.html, main.py

The files have been uploaded in GitHub.

Script in assistant.html file:

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

```
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Assignment 3 Chatbot</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk"
crossorigin="anonymous">
    <style>
        body{
            background-color: #66d7b9;
            color: black;
            margin: 5%;
        }
    </style>
</head>
<body>
    <h1 style="text-align: center">Chatbot for Hospital</h1>
    <script>
    window.watsonAssistantChatOptions = {
    integrationID: "148c8b05-d7de-489c-a970-517c3676838f", // The ID of this integration.
    region: "au-syd", // The region your integration is hosted in.
    serviceInstanceID: "efccfecd-0afd-4fd7-8f9e-8a9e819f7075", // 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>
</body>
</html>
```

main.py:

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

@app.route('/myassistant')
def assistant_page():
    return render_template('assistant.html')

if(__name__=='__main__'):
    app.run()
```