

Assignment-3

Date	10 October 2022
Team ID	PNT2022TMID30734
Project Name	Plasma Donor Application

1. CREATE A BUCKET IN IBM OBJECT STORAGE.

The screenshot displays the IBM Cloud Object Storage interface. The left sidebar shows the navigation menu with 'Buckets' selected. The main content area is titled 'Buckets' and includes a search bar, a 'Create bucket' button, and a table listing existing buckets.

Name	Public access	Location	Storage class	Created
plasmaadkn	Yes	jp-tok	Smart Tier	2022-10-25 9:59 PM

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

IBM Cloud

Search resources and products...

Cloud Object Storage

Storage instances

Cloud Object Storage-qg

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

If you're seeing more usage than expected, versions count towards your usage or you may have incomplete uploads [Learn more](#)

Prefix filter

Upload

<input type="checkbox"/>	Object name	Archived ⓘ	Size	Last modified	
<input type="checkbox"/>	Screen...2).png		118.4 KB	2022-10-25 10:01 PM	⋮
<input type="checkbox"/>	Screen...9).png		455.9 KB	2022-10-26 10:11 AM	⋮
<input type="checkbox"/>	Scree... (4).png		122.7 KB	2022-10-26 10:12 AM	⋮
<input type="checkbox"/>	Scree... (5).png		117.6 KB	2022-10-26 10:12 AM	⋮
<input type="checkbox"/>	swe...2.png		292.4 KB	2022-10-26 10:12 AM	⋮

Drag and drop files (objects) here or click to upload

plasma.avif

plasma donor ap...html

Show all

Task View

29°C

10:13 AM
10/26/2022

IBM Cloud

Search resources and products...

Cloud Object Storage

Storage instances

Cloud Object Storage-qg

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

Access policies

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 [Learn more](#)

Access policy update

Access group policy created

A new access policy for this bucket was created for the group:
Public Access
To delete/edit go to the [IAM console](#).

Create access policy

Context-based restrictions

Firewall (legacy)

plasma.avif

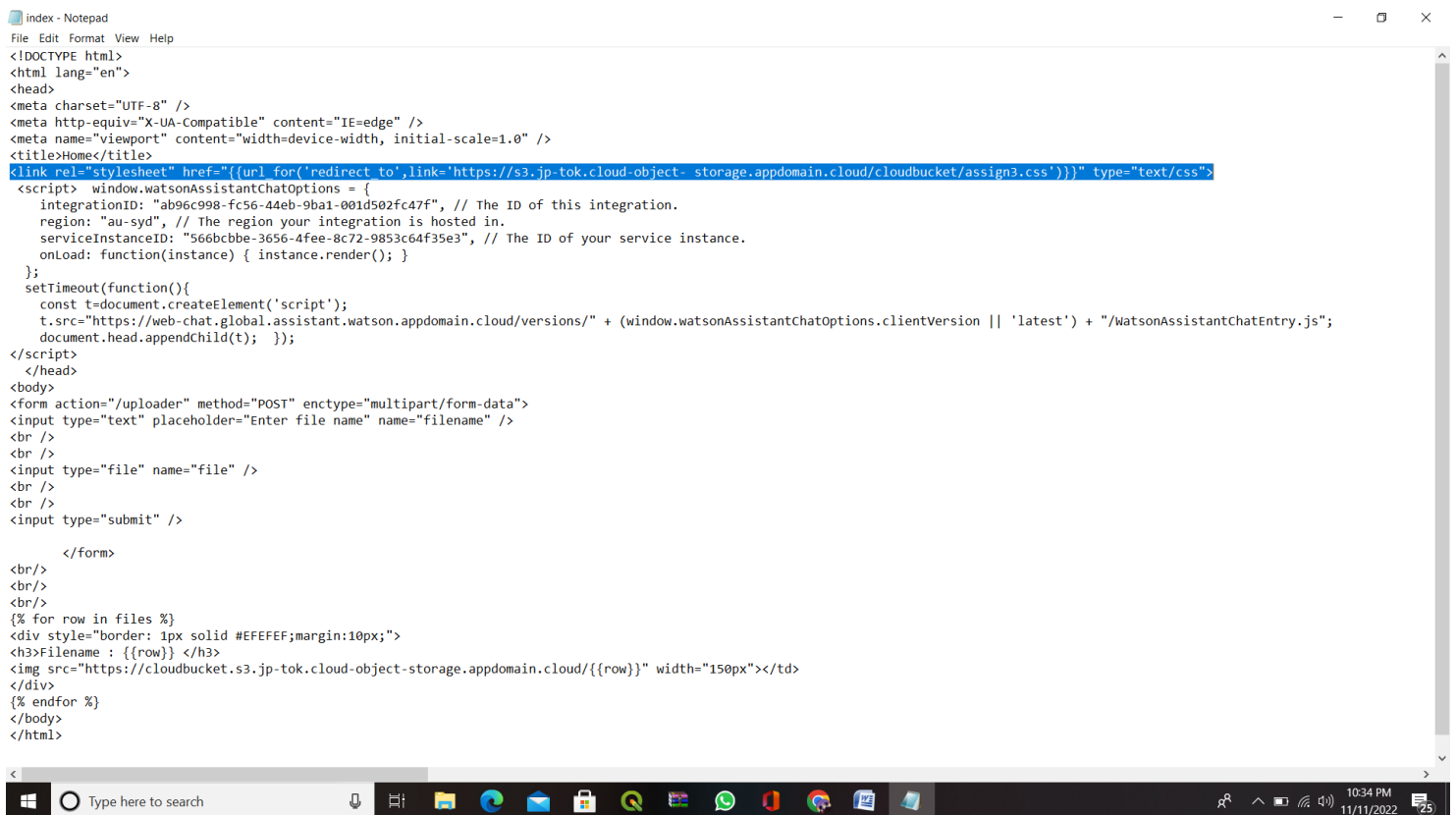
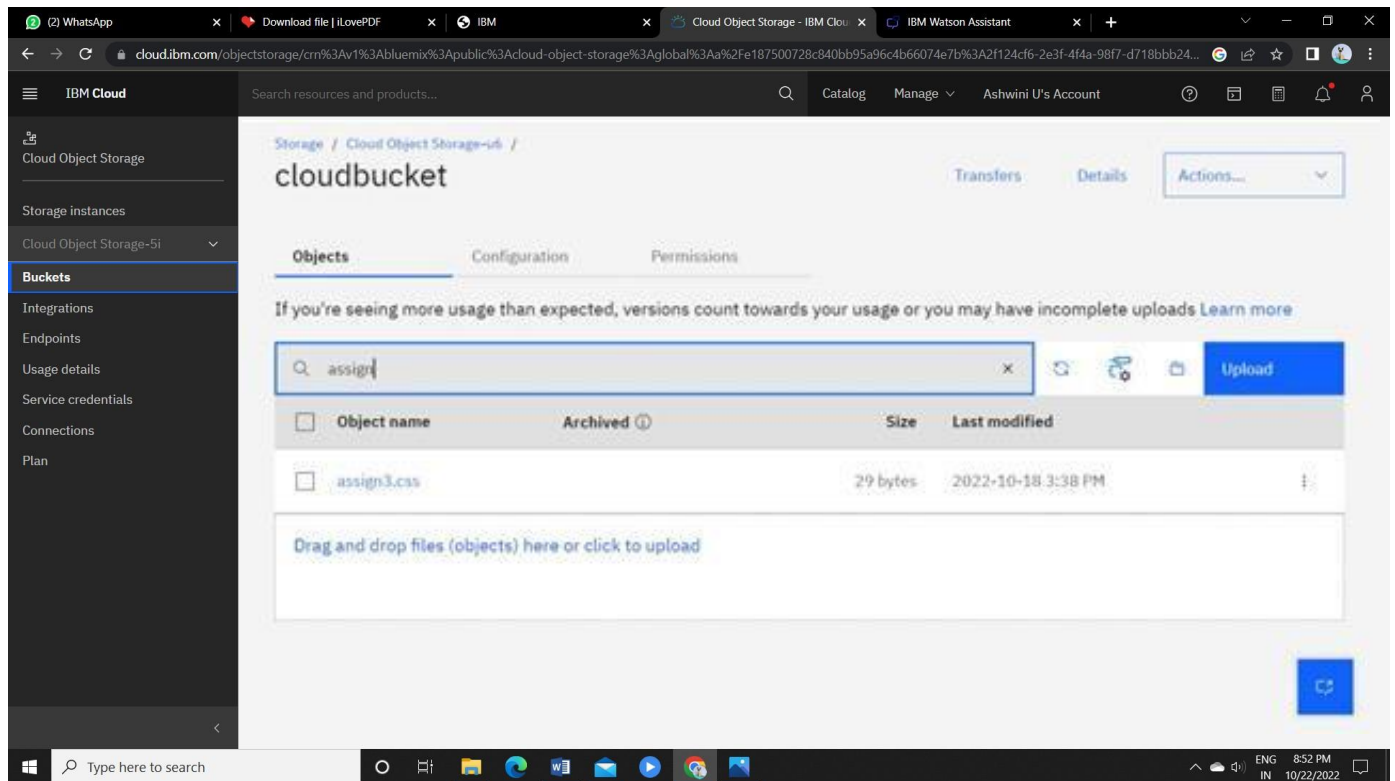
plasma donor ap...html

Show all

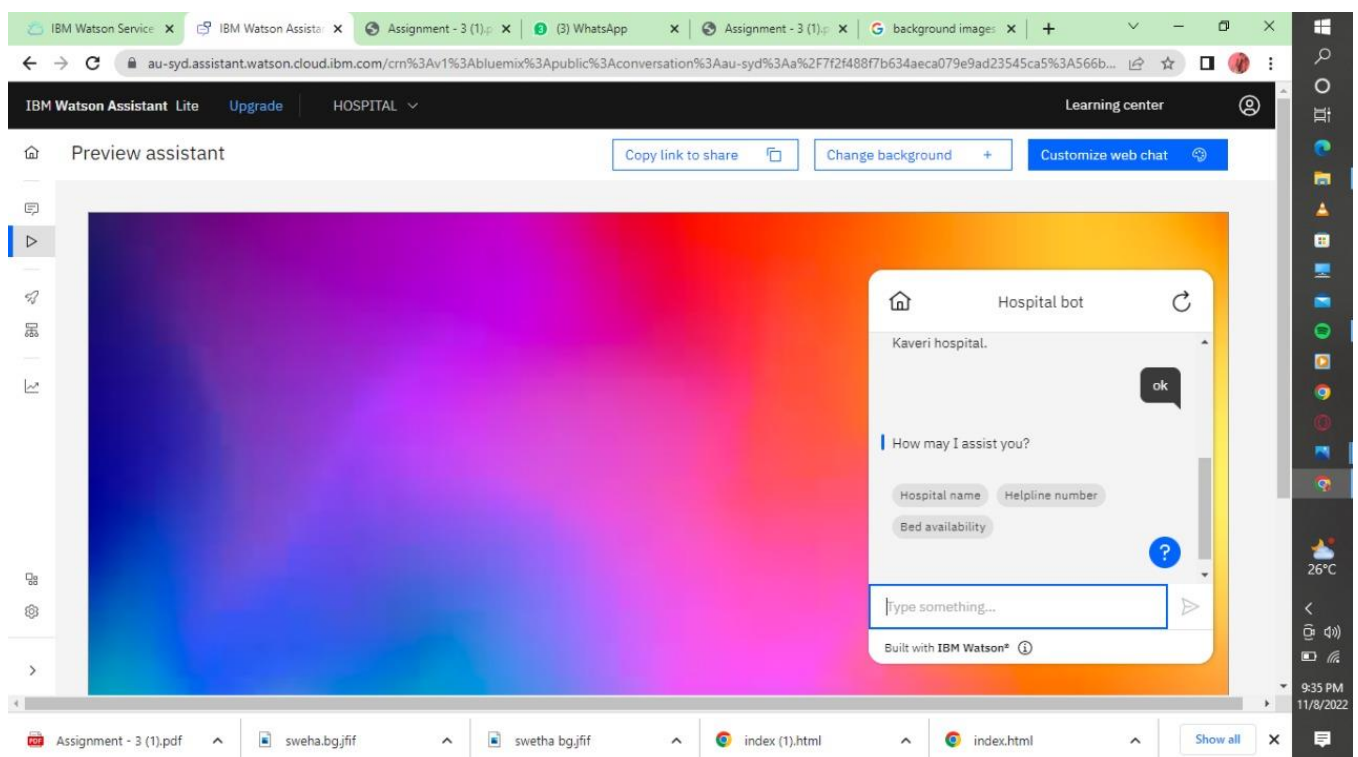
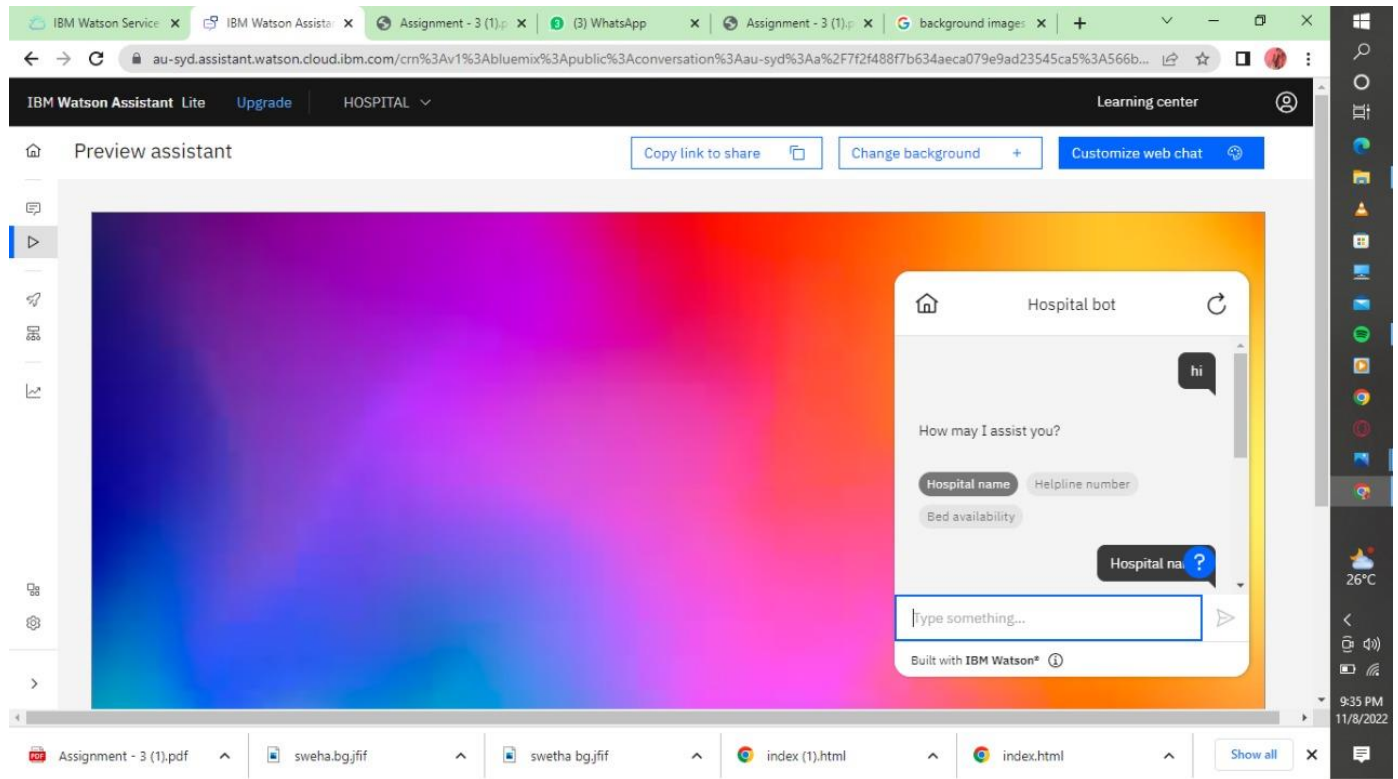
Satisf...

10:16 AM
10/26/2022

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



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



Web URL for Assistant:

<https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Fau-syd.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-566bcbbe-3656-4fee-8c72-9853c64f35e3%3A%3A101effc6-ad22-489d-bc5e-19c352ec9c23&integrationID=ab96c998-fc56-44eb-9ba1-001d502fc47f®ion=au-syd&serviceInstanceID=566bcbbe-3656-4fee-8c72-9853c64f35e3>

4. Create Watson assistant service with 10 steps and use 3 conditions in it. Load thatscript in HTML page.

The screenshot displays the IBM Watson Assistant Lite web interface. The top navigation bar includes 'IBM Watson Assistant Lite', 'Upgrade', 'HOSPITAL', and a 'Learning center' link. The main workspace is divided into three sections:

- Left Sidebar:** A vertical list of steps (1-10). Step 10 is highlighted, indicating it is the current step being edited.
- Center Editor:** The 'Step 10' configuration panel. It shows the step is 'without conditions'. The 'Assistant says' section contains a text box with the response: 'Thank you. your booking details will send through sms.' Below this, there are options for 'Edit response' and 'Edit validation'. The 'And then' section shows 'End the action' as the next step.
- Right Panel:** A 'Preview' button and a 'User enters free text' input field.

The bottom of the screen shows a file explorer with files like 'Assignment - 3 (1).pdf', 'swetha.bgjiff', and 'index.html'. The Windows taskbar on the right shows the date and time as 9:36 PM on 11/8/2022.

Included 3 conditions in steps:

The screenshot displays the IBM Watson Assistant configuration interface. On the left, the 'Conversation steps' panel shows a sequence of steps. Step 2 is highlighted, showing a 'Free text' input field and a 'Re-ask previous step(s)' button. The main panel shows the configuration for 'Step 2 is taken'. It is set to 'with conditions' and has one condition defined: 'If All of this is true: 1. How may ... is Hospital name'. The 'Assistant says' panel shows the response 'Kaveri hospital.'.

Customer starts with: hi

Conversation steps

1 How may I assist you?

1 Hospital name Helpline num... +1

Continue to next step

2 1 is Hospital name

Kaveri hospital.

Free text

Re-ask previous step(s)

3 1 is Helpline number

6637723759

Free text

New step +

Step 2 is taken with conditions fx

Conditions 1 condition

If All of this is true:

1. How may ... is Hospital name

and Add condition +

New condition group +

Assistant says

B I % % % % % % % %

Kaveri hospital.

Preview

The screenshot displays the IBM Watson Assistant configuration interface. On the left, the 'Conversation steps' panel shows a sequence of steps. Step 8 is highlighted, showing a 'Free text' input field and a 'Continue to next step' button. The main panel shows the configuration for 'Step 8 is taken'. It is set to 'with conditions' and has one condition defined: 'If All of this is true: 7. On which ... is on Tuesday, Thursday'. The 'Assistant says' panel shows the response 'For example: What type of transfer would you like to make?'.

Customer starts with: hi

Conversation steps

5 is Yes

On which date would you like for booking?

7 Date

Continue to next step

7 is any of Tuesday, Thursday

This step has no content

Continue to next step

9 Doctor is available at 2pm.

Free text

Continue to next step

Thank you, your booking details will send

New step +

Step 8 is taken with conditions fx

Conditions 1 condition

If All of this is true:

7. On which ... is on Tuesday, Thursday

and Add condition +

New condition group +

Assistant says

B I % % % % % % % %

For example: What type of transfer would you like to make?

Preview

IBM Watson Assistant LiteUpgradeHOSPITALLearning center

hi

1 is Helpline number

6637723759

3

Free text

Re-ask previous step(s)

1 is Bed availability

200 beds.

4

Free text

Continue to next step

Are you looking for an appointment?

Confirmation

5

Continue to next step

5 is Yes

New step +

Step 4 is taken with conditions

Conditions1 condition

If All of this is true:

1. How may ... is Bed availability

and Add condition +

New condition group +

Assistant says

200 beds.

Preview

Assignment - 3 (1).pdf

swaha.bgjfff

swetha bgjfff

index (1).html

index.html

Show all

26°C

9:36 PM

11/8/2022

Index.html

```
<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Home</title>

    <link rel="stylesheet" href="{ {url_for('redirect_to',link='https://s3.jp-tok.cloud-object-storage.appdomain.cloud/cloudbucket/assign3.css')} }" type="text/css">

  <script>

    window.watsonAssistantChatOptions = {

      integrationID: "ab96c998-fc56-44eb-9ba1-001d502fc47f", // The ID of this
      integration.

      region: "au-syd", // The region your integration is hosted in.

      serviceInstanceID: "566bcbbe-3656-4fee-8c72-9853c64f35e3", // 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>
</head>
<body>
  <form action="/uploader" method="POST" enctype="multipart/form-data">
    <input type="text" placeholder="Enter file name" name="filename" />
    <br />
    <br />
    <input type="file" name="file" />
    <br />
    <br />
    <input type="submit" />
```

```

</form>

<br/>

<br/>

<br/>

{ % for row in files % }

    <div style="border: 1px solid #EFEFEF;margin:10px;">
        <h3>Filename : {{row}} </h3>

        </td>
    </div>

{ % endfor % }

</body>

</html>

```

App.py

```

import io

from flask import Flask,redirect,url_for,render_template,request

import ibm_boto3

from ibm_botocore.client import Config, ClientError

COS_ENDPOINT="https://s3.jp-tok.cloud-object-storage.appdomain.cloud"

COS_API_KEY_ID=""

COS_INSTANCE_CRN=""


cos = ibm_boto3.resource("s3",
    ibm_api_key_id=COS_API_KEY_ID,
    ibm_service_instance_id=COS_INSTANCE_CRN,
    config=Config(signature_version="oauth"),
    endpoint_url=COS_ENDPOINT
)

```

```
app=Flask(__name__)
```

```
@app.route('/')
```

```
def index():
```

```
try:
```

```
    files = cos.Bucket('cloudbucket').objects.all()
```

```
    files_names = []
```

```
    for file in files:
```

```
        files_names.append(file.key)
```

```
        print(file)
```

```
        print("Item: {0} ({1} bytes)".format(file.key, file.size))
```

```
    return render_template('index.html',files=files_names)
```

```
except ClientError as be:
```

```
    print("CLIENT ERROR: {0}\n".format(be))
```

```
    return render_template('index.html')
```

```
except Exception as e:
```

```
    print("Unable to retrieve bucket contents: {0}".format(e))
```

```
    return render_template('index.html')
```

```
@app.route('/uploader',methods=['POST'])
```

```
def upload():
```

```
name_file=request.form['filename']
```

```
f = request.files['file']
```

```
try:
```

```
    part_size = 1024 * 1024 * 5
```

```
    file_threshold = 1024 * 1024 * 15
```

```
transfer_config = ibm_boto3.s3.transfer.TransferConfig(
```

```
    multipart_threshold=file_threshold,
```

```
        multipart_chunksize=part_size
    )

    content = f.read()
    cos.Object('cloudbucket', name_file).upload_fileobj(
        Fileobj=io.BytesIO(content),
        Config=transfer_config
    )
    return redirect(url_for('index'))
```

```
except ClientError as be:
    print("CLIENT ERROR: {0}\n".format(be))
    return redirect(url_for('index'))
```

```
except Exception as e:
    print("Unable to complete multi-part upload: {0}".format(e))
    return redirect(url_for('index'))
```

```
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080, debug=True)
```