

Assignment -3



Assignment Date	07 October 2022
Student Name	Kaavya S
Student Roll Number	111519205017
Maximum Marks	2 Marks

Question 1:

Create a bucket in IBM object Storage.


Buckets

Buckets serve as containers for objects, and can be individually configured in terms of their location, resiliency, object lifecycle rules.

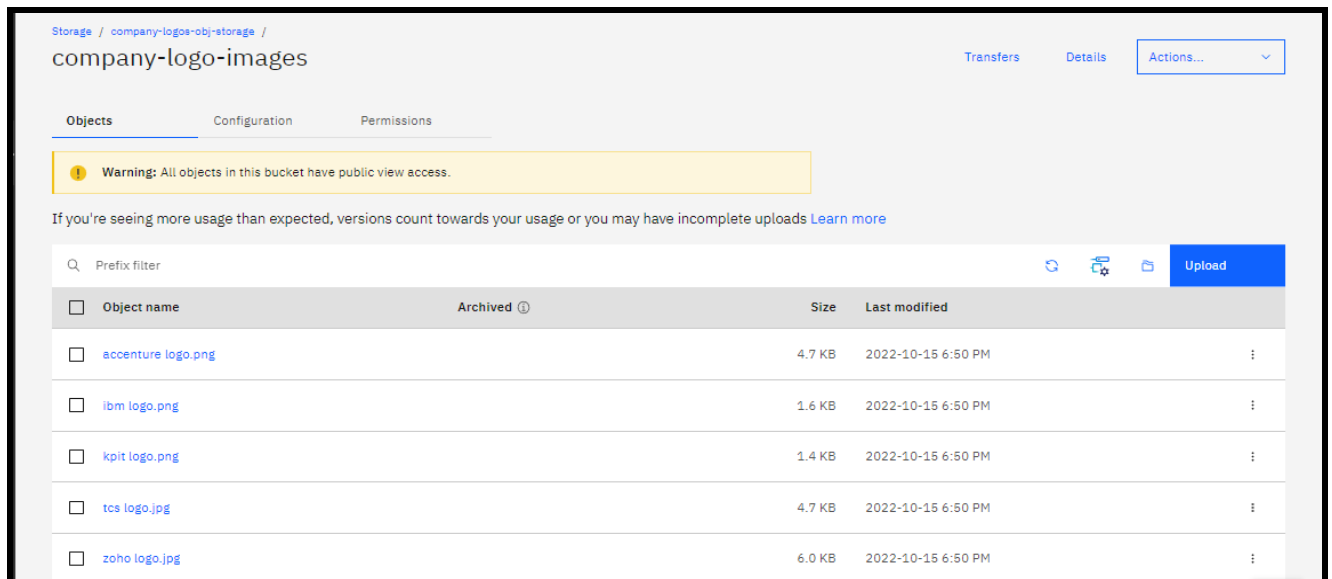
Create bucket +

Name	Public access ⓘ	Location ⓘ	Storage class	Created	
company-logo-images	Yes	us-south	Smart Tier	2022-10-15 6:40 PM	⋮

 A bucket created successfully! logo-images has been created and is now available to add objects.

Question 2:

Upload any 5 images to IBM object storage and make it public. Write HTML code to display all the 5 images.



HTML code for displaying images in cloud2 :-

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
<head>
  <meta charset="utf-8">
  <title>company logos</title>
</head>
<style>
  body{
    text-align: center;
    background-color:rgb(253, 51, 206);

  }
  h2{
    color: black;
    font-size: 50px;
```

```
}
h1{
    color:black;
    text-align: center;
    font-size: 75px;

}
img{
    height : 230px;
    width: 430px;
}
</style>
<body>
    <h1>Popular company logos</h1>
    <h2>Ibm</h2>
    
    <h2>Accenture</h2>
    
    <h2>Zoho</h2>
    
    <h2>Kpit</h2>
    
    <h2>Tcs</h2>
    
</body>
</html>
```

HTML Output for displaying images from IBM cloud storage: 4-



Question 3:


Upload a CSS page to the object storage and use the same page in your HTML code. HTML code with CSS file present in IBM cloud storage.


Style.css code:

```
body{
  background-color: aliceblue;
}
h1{
  text-align: center;
  color: brown;
}
p{
  color: whitesmoke;
  text-align: center;
}
```

style.css

x

[Download object](#) 


[Delete object](#) 

Overview

Lifecycle

Object details

Last modified	Object size	Storage class	Tags
2022-10-15 7:30 PM	152 bytes	Smart Tier	0 Tags

Object Public URL 

<https://company-logo-images.s3.us-south.cloud-object-storage.appdomain.cloud/style.css>



HTML code:

```
<html>
  <head>
    <title>Home page</title>
    <link rel="stylesheet" href="https://company-logo-images.s3.us-south.cloud-
object-storage.appdomain.cloud/style.css">

  </head>
  <body>
    <h1>Hello</h1>
    <h3>Lets look upon thes css file containing IBM cloud storage<a
href="https://company-logo-images.s3.us-south.cloud-object-
storage.appdomain.cloud/style.css">click here</a></h3>

  </body>
</html>
```

HTML code output:

Hello

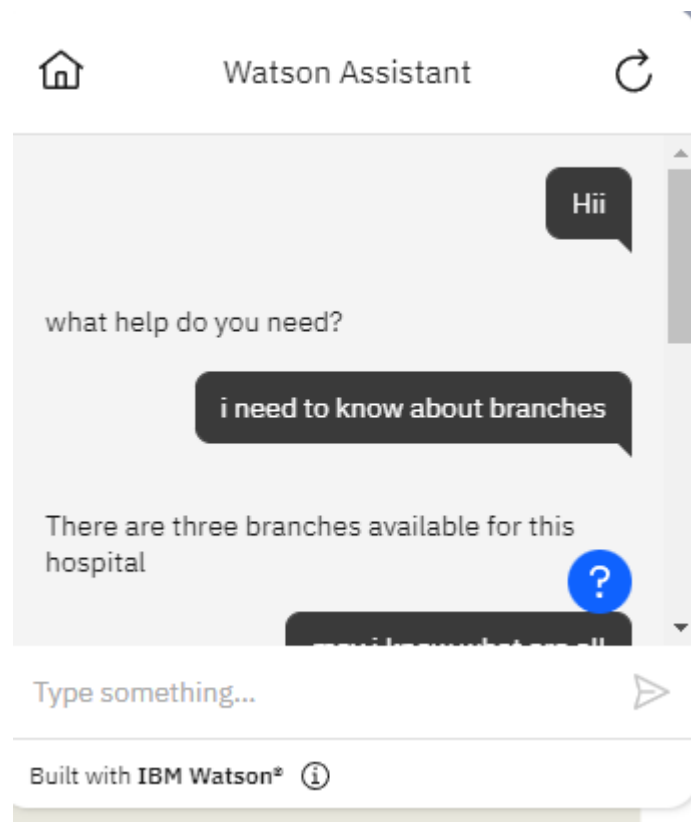
Lets look upon thes css file containing IBM cloud storage[click here](https://company-logo-images.s3.us-south.cloud-object-storage.appdomain.cloud/style.css)

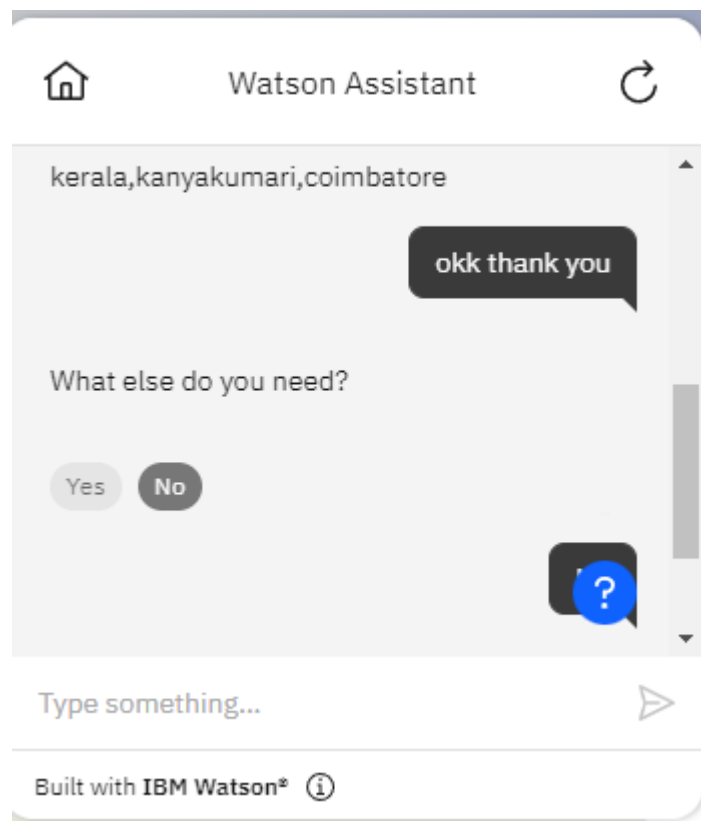
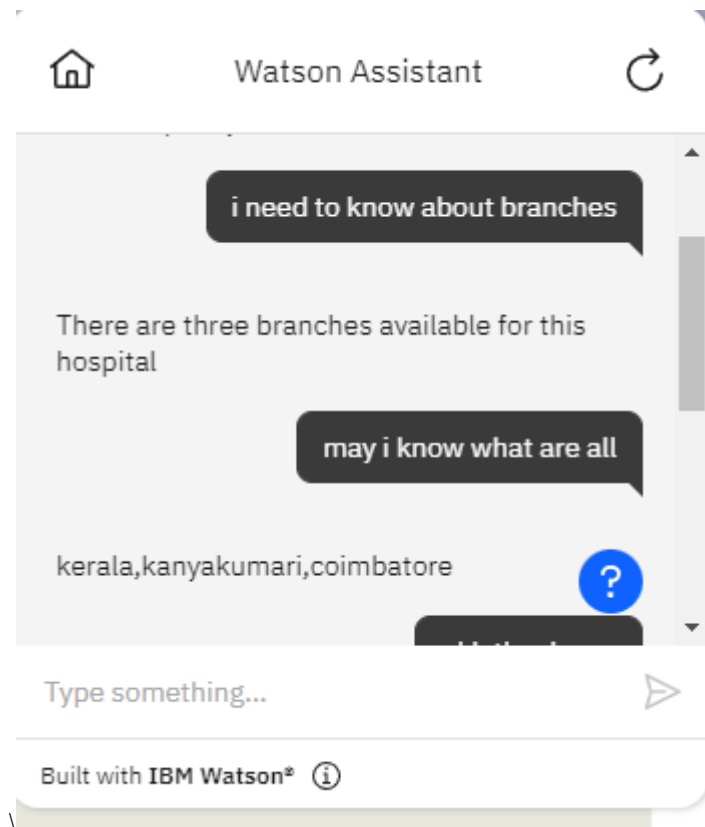
Question 4

Design a chatbot using IBM Watson assistant for hospitals. Ex: User comes with a query to know the branches for that hospital in your city. Submit the web URL of that chat bot as an assignment.

URL for the Chat Bot :

<https://web-chat.global.assistant.watson.appdomain.cloud/preview.html?backgroundImageURL=https%3A%2F%2Fus-south.assistant.watson.cloud.ibm.com%2Fpublic%2Fimages%2Fupx-41fd19c7-5c1e-4f7d-a76e-e7dc4ab91de9%3A%3Ab1f9a53e-bf29-425b-9906-cb88a17fdf86&integrationID=f4828607-0c90-4be4-a027-206601a1214d®ion=us-south&serviceInstanceID=41fd19c7-5c1e-4f7d-a76e-e7dc4ab91de9>







Watson Assistant



What else do you need?

Yes

No

No

Thank you for visiting

?

Type something...



Built with IBM Watson® ⓘ

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

HTML CODE:

```
<!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>Chat bot</title>
<style>
h1{
text-align: center;
color: #ffffff;
}
body{
background-color: #000000;
} input[type="text"],input[type="password"]{ height: 42px;
width: 400px;
font-family: cursive;
border-radius: 100px;
background-color:rgb(237, 15, 193) ;
} input[type="submit"]{ height: 42px;
width: 100px;
color: #000;
background-color: rgb(25, 14, 236);
font-size: medium ; margin-top: 20px; border-radius: 50px;
}
p{
color: #ffffff;
font-size: 28px;
}
form{
text-align: center;
}
small{
color: #ffffff;
}
img{
fill: auto;
}
</style>
</head>
<body>
<h1>KaavyaSenthil Welcomes you!!!</h1>
<h1>login to your account</h1>
<form>
<p>Enter-Username </p>
<input type="text" >
<br><br>
<p>Enter-Password </p>
```

```

<input type="password">
<br><br>
<input type="submit" value="login" >
<br><br>
<small>Click here to browse more <a
href="https://www.youtube.com/watch?v=E40UEoa19As"> info</a></small>
</form>
<script>
window.watsonAssistantChatOptions = {
  integrationID:"4d39fa65-a7c4-4129-94ad-e0aa2d1c8e0b", // The ID of this
integration.
region:"us-south", // The region your integration is hosted in.
serviceInstanceID:"41fd19c7-5c1e-4f7d-a76e-e7dc4ab91de9", // 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/preview/"
+ (window.watsonAssistantChatOptions.clientVersion || 'latest') +
"/WatsonAssistantChatEntry.js";
document.head.appendChild(t);
});
</script>
</body>
</html>

```

App.py code:

```

from flask import Flask, render_template

app = Flask(__name__)

@app.route("/")
def home():
    return render_template("index.html")

@app.route("/salvador")
def salvador():
    return "Hello, Salvador"

if __name__ == "__main__":
    app.run(debug=True)

```

HTML CODE OUTPUT:

KaavyaSenthil Welcomes you!!!

login to your account

Enter-Username

Enter-Password

login

Click here to browse more [info](#)

ASSISTANT OUTPUT:

