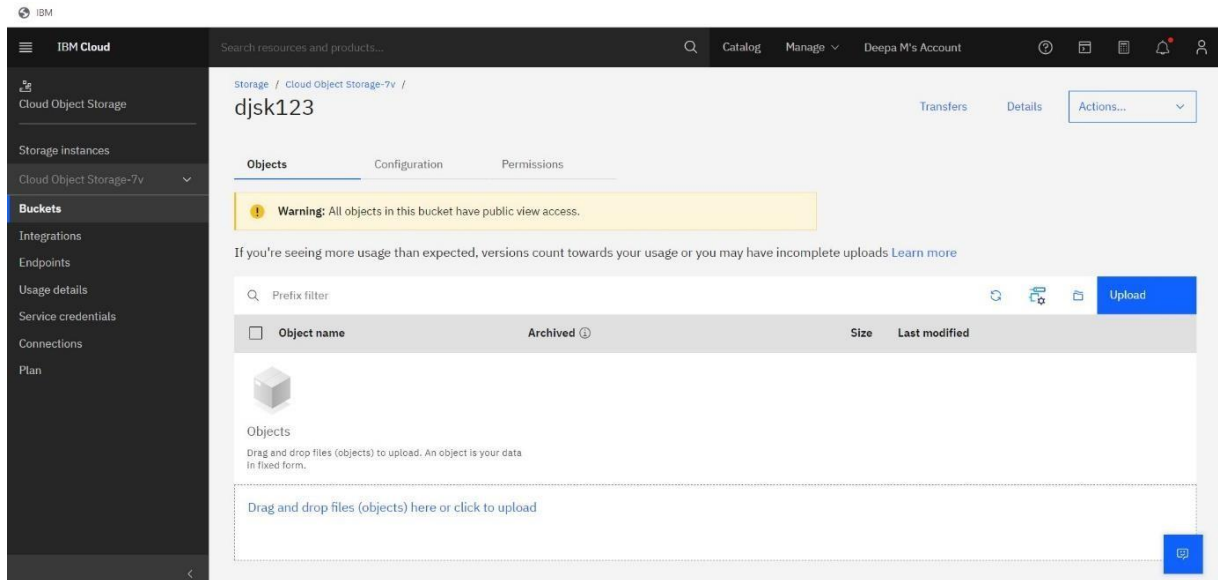


ASSIGNMENT-3

HTML PROGRAMMING

Question -1:

1. Create a Bucket in IBM object storage.



Question -2:

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

Solution:

```
<DOCTYPE html>
<html>
<head>
<title>First Website</title>
</head>
<body>
<h1>This is Images</h1>

</body>
<body>

</body>
<body>

```

```

</body>
<body>

</body>
<body>

</body>
</html>

```

This is Images



IBM Cloud

Cloud Object Storage

Storage instances

Cloud Object Storage-7v

Buckets

Integrations

Endpoints

Usage details

Service credentials

Connections

Plan

Search resources and products...

Q Catalog Manage Deepa M's Account

Storage / Cloud Object Storage-7v / djsk123

Transfers Details Actions...

Objects Configuration Permissions

Warning: All objects in this bucket have public view access.

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

Q Prefix filter

Refresh Download Upload

| <input type="checkbox"/> | Object name | Archived ⓘ | Size | Last modified | |
|--------------------------|-------------|------------|-----------|--------------------|---|
| <input type="checkbox"/> | A...tml | | 964 bytes | 2022-10-29 2:38 PM | ⋮ |
| <input type="checkbox"/> | A...tml | | 1.1 KB | 2022-10-29 2:35 PM | ⋮ |

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

⊞

Question -3:

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

Solution:

```
<DOCTYPE html>
<html>
<head>
<title>First Website</title>
</head>
<body>
<h1>This is a heading</h1>
<p>This is all images</p>

</body>
<body>

</body>
<body>

</body>
<body>

</body>
<body>

</body>
</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">
<link rel= "stylesheet"
<title>document</title>
</head>
<body>
<label for="files">select multiple files</label>
<input type="file" id="files" multiple="multiple" accept="image/jpeg,image/png,image/jpg">
<output id="result">
<script src="script.js"></script>
</body>
<files.length;i++>[
if(! files [i].type.match("image"))continue;
const pic reader new filereader():
pic reader.addeventlistener("load",function(event){
const picfile = event.target;
const div = document.createelement("div");
```

This is a heading

This is all images



IBM Cloud

Search resources and products...

Storage / Cloud Object Storage-7v /

djsk123

Transfers Details Actions...

Objects Configuration Permissions

Warning: All objects in this bucket have public view access.

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

Prefix filter

| <input type="checkbox"/> | Object name | Archived ⓘ | Size | Last modified | |
|--------------------------|-------------|------------|--------|--------------------|---|
| <input type="checkbox"/> | A... .html | | 1.1 KB | 2022-10-29 2:35 PM | ⋮ |

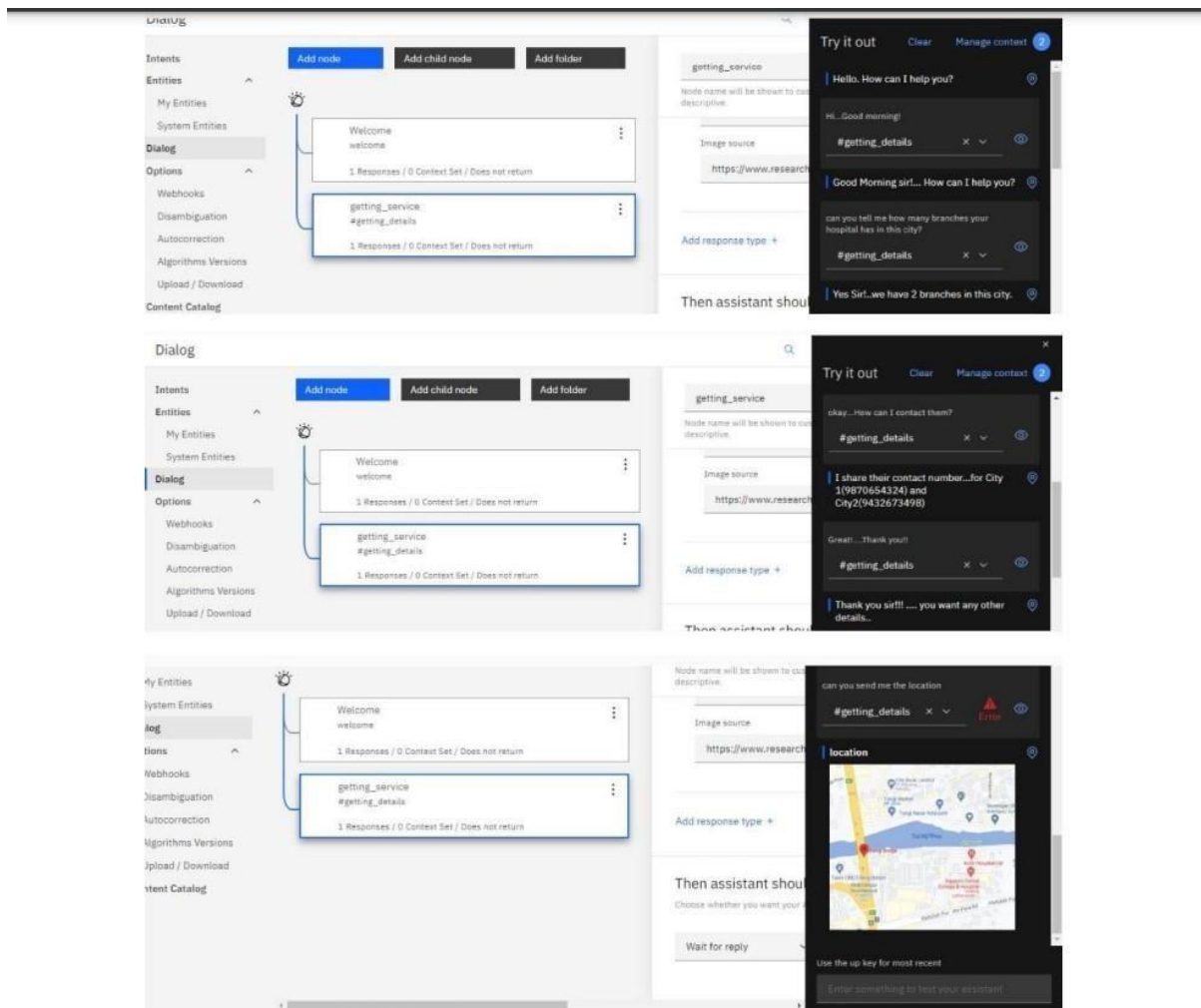
Drag and drop files (objects) here or click to upload

Upload

Question -4:

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

Solution:



Question -5:

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

Footer

Solution:

```
{
  "intents": [
    {
      "intent": "email_data",
      "examples": [
        {
          "text": "Can you email it to @email"
        },
        {
          "text": "Can you email this file to @sys-person"
        },
        {
          "text": "Can you forward this data to @sys-person"
        }
      ]
    }
  ]
}
```

```
{
"text": "Can you forward this file to @sys-person?"
},
{
"text": "Can you send this data to @sys-person"
},
{
"text": "Can you send this to @sys-person"
},
{
"text": "Email me this data"
},
{
"text": "Email the data at @email"
},
{
"text": "Email the data to @email and @email"
},
{
"text": "Email this file to @sys-person"
},
{
"text": "forward it to @email"
},
{
"text": "Forward this data to @email"
},
{
"text": "Forward this data to @email and @email"
},
{
"text": "Forward this data to me by email"
},
{
"text": "Forward this file to @email"
},
{
"text": "Forward this file to my email"
},
{
"text": "Forward this file to @sys-person"
},
{
"text": "Forward this file to yara.rizk@ibm.com"
},
{
"text": "forward this to @email"
},
{
"text": "Send the data to @email, @email, and @email"
},
}
```

```
{
  "text": "Send this data over email"
},
{
  "text": "Send this data to @sys-person"
}
],
"description": "Send the data as attachment over email"
},
{
  "intent": "export_csv",
  "examples": [
    {
      "text": "create a comma separated csv file"
    },
    {
      "text": "create a csv file"
    },
    {
      "text": "create a csv file based on the data"
    },
    {
      "text": "create a csv file for the query results"
    },
    {
      "text": "create a csv file from this data"
    },
    {
      "text": "Create a csv from the query"
    },
    {
      "text": "create an excel file for this query"
    },
    {
      "text": "create an excel file from this data"
    },
    {
      "text": "create an excel file with this data"
    },
    {
      "text": "export as csv"
    },
    {
      "text": "export as csv file"
    },
    {
      "text": "Export data in comma separated values"
    },
    {
      "text": "export data to csv"
    },
  ],
}
```

```
{
  "text": "export the data"
},
{
  "text": "export the data as a csv"
},
{
  "text": "export the data as a csv file"
},
{
  "text": "export the data as csv"
},
{
  "text": "export the data as csv file"
},
{
  "text": "export the data in a csv"
},
{
  "text": "Export the data in csv format"
},
{
  "text": "export the data to a file"
},
{
  "text": "export the data to csv"
},
{
  "text": "export this data to a csv file"
},
{
  "text": "export this data to a file"
},
{
  "text": "export to a csv file"
},
{
  "text": "export to csv"
},
{
  "text": "export to csv file"
},
{
  "text": "export to file"
},
{
  "text": "i want a csv file containing the data"
},
{
  "text": "i want an excel file containing this data"
},
}
```



```
{
"text": "produce a csv file from the data"
},
{
"text": "produce a csv file from this data"
},
{
"text": "put the data in a csv file"
},
{
"text": "Put the data in csv"
},
{
"text": "put this data in a csv file"
},
{
"text": "save my data as csv"
},
{
"text": "save the data as csv"
},
{
"text": "save the data as csv file"
},
{
"text": "save the data in a csv"
},
{
"text": "save the data in csv file"
},
{
"text": "save the data to file"
},
{
"text": "save this data as csv"
},
{
"text": "save this data in a csv file"
},
{
"text": "save this data to a file"
},
{
"text": "save to file"
},
{
"text": "Send csv to me"
},
{
"text": "Send me a csv file"
},
}
```

```

{
  "text": "Send me a csv file for this data"
},
{
  "text": "Send me the data in excel format"
}
],
"description": "export data in csv format"
},
"entities": [
  {
    "entity": "email",
    "values": [
      {
        "type": "patterns",
        "value": "{word}@{word}.{word}",
        "patterns": [
          "\\b[A-Za-z0-9._%+-]+@[A-Za-z0-9-]+\\.([A-Za-z]{2,})\\b"
        ]
      }
    ],
    "fuzzy_match": false
  },
  {
    "entity": "sys-person",
    "values": [],
    "fuzzy_match": true
  }
],
"metadata": {
  "api_version": {
    "major_version": "v2",
    "minor_version": "2018-11-08"
  }
},
"dialog_nodes": [
  {
    "type": "standard",
    "title": "Anything else",
    "output": {
      "generic": [
        {
          "values": [
            {
              "text": "I didn't understand. You can try rephrasing."
            }
          ]
        },
        {
          "text": "Can you reword your statement? I'm not understanding."
        }
      ]
    }
  }
]

```

```
"text": "I didn't get your meaning."
}
],
"response_type": "text",
"selection_policy": "sequential"
}
],
},
"conditions": "anything_else",
"dialog_node": "Anything else",
"previous_sibling": "node_4_1573677581190"
},
{
"type": "event_handler",
"output": {
"generic": [
{
"values": [],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"parent": "slot_9_1573677592451",
"event_name": "focus",
"dialog_node": "handler_5_1573677592466",
"previous_sibling": "handler_7_1573677592466"
},
{
"type": "event_handler",
"parent": "slot_9_1573677592451",
"context": {
"email": "@email"
},
"conditions": "@email",
"event_name": "input",
"dialog_node": "handler_7_1573677592466"
},
{
"type": "frame",
"conditions": "#email_data",
"dialog_node": "node_4_1573677581190",
"previous_sibling": "Welcome"
},
{
"type": "slot",
"parent": "node_4_1573677581190",
"variable": "$email",
"dialog_node": "slot_9_1573677592451"
},
{
```

```
"type": "standard",
"title": "Welcome",
"output": {
  "generic": [
    {
      "values": [
        {
          "text": "Hello. How can I help you?"
        }
      ],
      "response_type": "text",
      "selection_policy": "sequential"
    }
  ],
  "conditions": "welcome",
  "dialog_node": "Welcome"
},
"counterexamples": [],
"system_settings": {
  "tooling": {
    "store_generic_responses": true
  },
  "off_topic": {
    "enabled": true
  },
  "disambiguation": {
    "prompt": "Did you mean:",
    "none_of_the_above_prompt": "None of the above"
  },
  "human_agent_assist": {
    "prompt": "Did you mean:"
  },
  "spelling_auto_correct": true
},
"learning_opt_out": false,
"name": "Export-data-WA",
"language": "en",
"description": "Skill to export data"
}
```

