

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

Scenario : A healthcare organization uses IBM Cloud services to store and manage sensitive patient medical records. The organization aims to ensure that the data is confidential, its integrity is maintained, and it is available whenever needed.

Show the implementation for both Cloudant and object storage to meet the objective

Steps and screenshots :

1. Visit Cloudant dashboard and enable CORS settings

The screenshot shows the IBM Cloudant CORS settings page in a Firefox browser. The URL is https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#account/cors. The page title is "Cloudant Dashboard - account". On the left, there's a sidebar with icons for Monitoring, Databases, Replication, Active Tasks, Account (which is selected), Support, and Documentation. The main content area has tabs for Capacity, Announcements, CORS (which is selected and highlighted in blue), and Settings. Below the tabs, a message states: "Cross-Origin Resource Sharing (CORS) lets you connect to remote servers directly from the browser. Using CORS, you can host browser-based apps on static pages, and load data directly from Cloudant." It then says "CORS is currently disabled." and features a prominent blue "Enable CORS" button. A red box and arrow highlight this button. At the bottom left, there's an "IBM Cloudant" logo and "Log Out IBMid-6940008Y8M".

Name - Yash Lakhtariya

Enrollment number - 21162101012

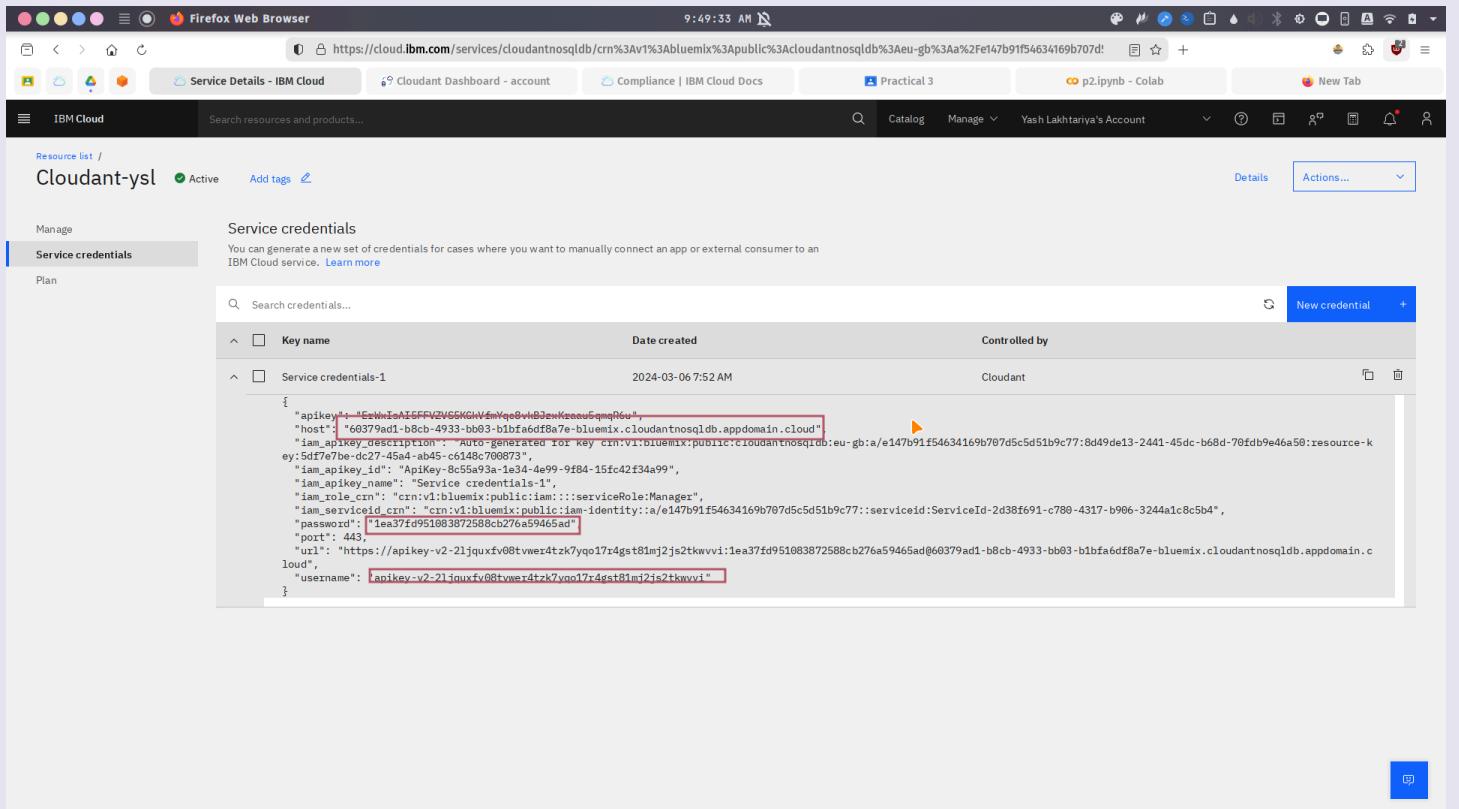
Branch - CBA Batch - 71

CS Practical 3

The screenshot shows a Firefox browser window with the URL <https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#account/cors>. The page title is "Account". On the left, there's a sidebar with links: Monitoring, Databases, Replication, Active Tasks, Account (which is selected), Support, and Documentation. The main content area has tabs: Capacity, Announcements, CORS (which is selected and highlighted in blue), and Settings. A success message "CORS settings updated." is displayed at the top right. Below it, a note says "Cross-Origin Resource Sharing (CORS) lets you connect to remote servers directly from the browser. Using CORS, you can host browser-based apps on static pages, and load data directly from Cloudant." It states "CORS is currently enabled." and has a "Disable CORS" button. Under "Origin Domains", it says "Databases will accept requests from these domains:" and shows two options: "All domains (*)" (selected and highlighted with a red box and orange arrow) and "Restrict to specific domains". At the bottom left, there's an "IBM Cloudant" logo and "Log Out IBMid-6940008Y8M".

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

2. Copy the Cloudant service credentials and using API, fetch all databases' list



The screenshot shows the IBM Cloud Service Details interface for a 'Cloudant-yasl' service. The 'Service credentials' tab is selected. A table lists a single credential entry:

Key name	Date created	Controlled by
Service credentials-1	2024-03-06 7:52 AM	Cloudant

The credential details are as follows:

```
{  
  "apikey": "EswIaAICFFV2VSCKGIVmYeo8v1bJewKeeas5qmgR6u",  
  "host": "60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud",  
  "iam_apkey_description": "Auto-generated for key crn:v1:bluemix:public:cloudantnosqldb:eu-gb:a/e147b91f54634169b707d5c5d51b9c77:8d49de13-2441-45dc-b68d-70fdb9e46a50:resource-key:5dfe7be-dc27-4544-ab45-c6148c700873",  
  "iam_apkey_id": "ApiKey-8c55a93a-1e34-4e99-9fb4-15fc422f34a99",  
  "iam_apkey_name": "Service credentials-1",  
  "iam_role_crn": "crn:v1:bluemix:public:iam:::serviceRole:Manager",  
  "iam_serviceid_crn": "crn:v1:bluemix:public:iam::identity:a/e147b91f54634169b707d5c5d51b9c77:serviceId:ServiceId-2d38f691-c780-4317-b906-3244a1c8c5b4",  
  "password": "1ea37fd95108387258cb276a59465ad",  
  "port": 443,  
  "url": "https://apikey-v2-21jquxfv08twer4tzk7yqo17r4gst81mj2js2tkwv1:1ea37fd95108387258cb276a59465ad@60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud",  
  "username": "1anikey-v2-21jquxfv08twer4tzk7yqo17r4gst81mj2js2tkwv1"  
}
```

Code :

```
<!DOCTYPE html>  
  
<html lang="en">  
  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Cloudant CORS Example</title>  
</head>  
  
<body>  
  <h1>Cloudant CORS Example</h1>  
  <button id="fetchDatabases">Databases Aavadyo</button>  
  <pre id="result"></pre>
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

```
<script>
    document.getElementById('fetchDatabases').addEventListener('click',
function() {
    const url =
"https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appd
omain.cloud/_all_dbs";
    const username =
'apikey-v2-21jquxfv08tvwer4tzk7yqo17r4gst81mj2js2tkwvvi';
    const password = '1ea37fd951083872588cb276a59465ad';
    const headers = new Headers();
    headers.append('Authorization', 'Basic ' + btoa(username + ':' +
password));

    fetch(url, { method: 'GET', headers: headers })
        .then(response => response.json())
        .then(data => {
            document.getElementById('result').textContent =
JSON.stringify(data, null, 2);
        })
        .catch(error => {
            document.getElementById('result').textContent = 'Fetch
error: ' + error.message;
        });
    });
</script>
</body>
</html>
```

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA

Batch - 71

CS Practical 3

The screenshot shows a Visual Studio Code interface with the title "Visual Studio Code - URL Handler". The status bar indicates the time as 9:59:27 AM. The Explorer sidebar on the left shows a project structure for "PRACTICAL_3" containing "node_modules", "public", ".gitignore", "app.js", "package-lock.json", and "package.json". The "index.html" file is open in the editor, displaying the following code:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Cloudant CORS Example</title>
  </head>
  <body>
    <h1>Cloudant CORS Example</h1>
    <button id="fetchDatabases">Databases Aavadyo</button>
    <pre id="result"></pre>

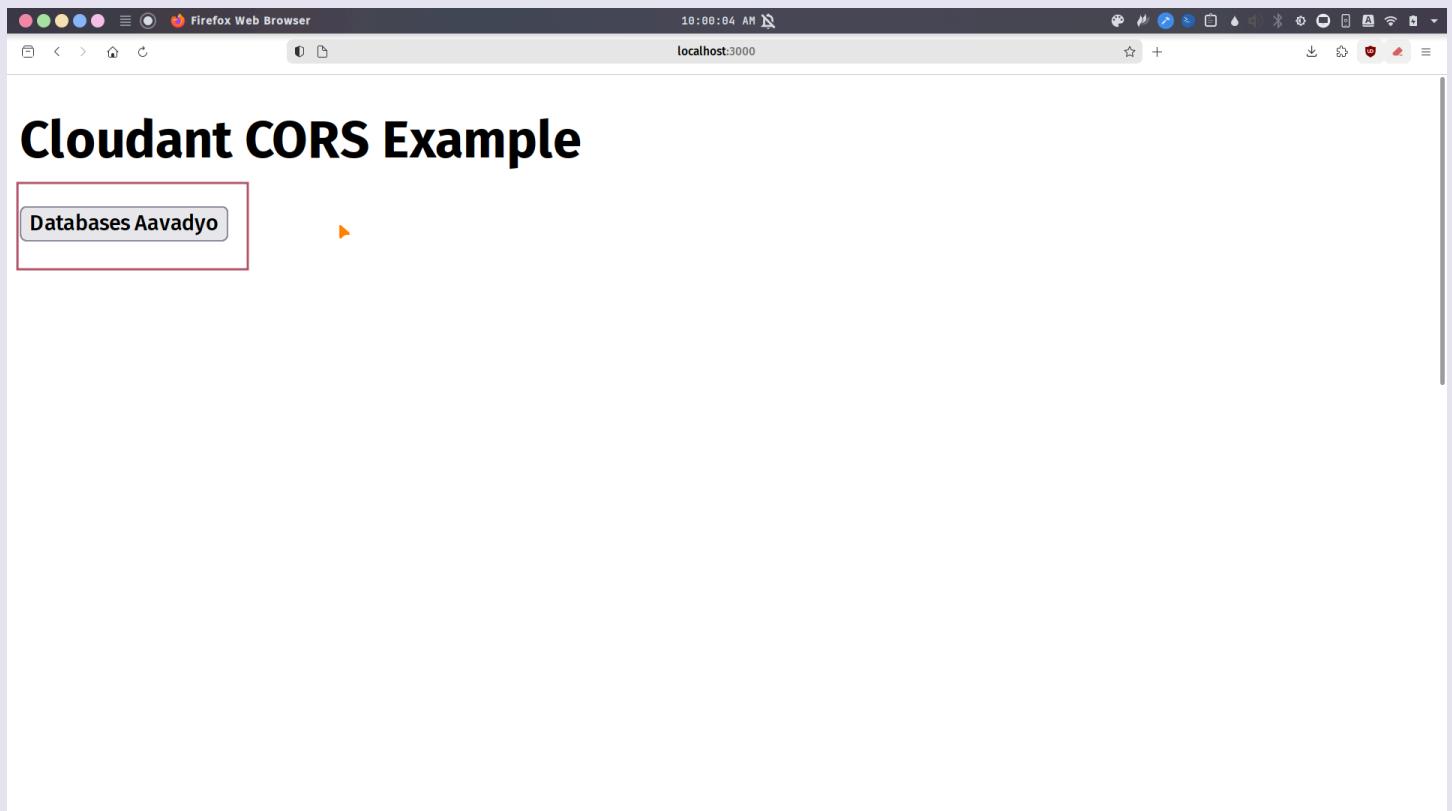
    <script>
      document.getElementById('fetchDatabases').addEventListener('click', function() {
        const url = "https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud/_all_dbs";
        const username = 'apikey-v2-2ljquxfv08tvver4tzk7yqo17r4gst81mj2js2tkwvvi';
        const password = '1ea37fd951083872588cb276a59465ad';
        const headers = new Headers();
        headers.append('Authorization', 'Basic ' + btoa(username + ':' + password));

        fetch(url, { method: 'GET', headers: headers })
          .then(response => response.json())
          .then(data => {
            document.getElementById('result').textContent = JSON.stringify(data, null, 2);
          })
          .catch(error => {
            document.getElementById('result').textContent = 'Fetch error: ' + error.message;
          });
      });
    </script>
  </body>
</html>
```

The code implements a simple web application that triggers a database query when the "fetchDatabases" button is clicked. It uses the Fetch API to make a GET request to a Cloudant database endpoint, handling the response and displaying the results in an HTML pre element.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

3. Try running API through nodejs

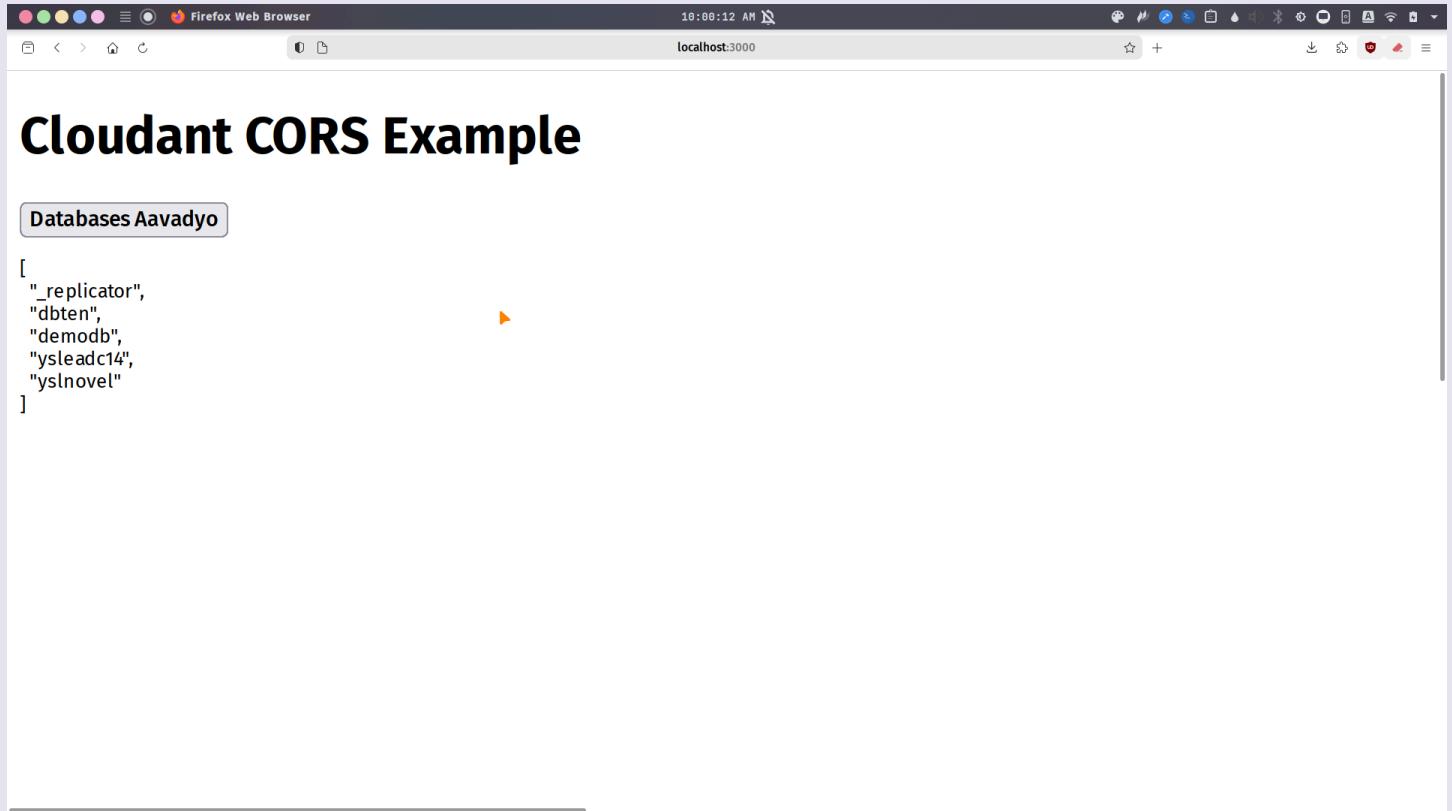


Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3



The screenshot shows a Firefox browser window with the title "Firefox Web Browser". The address bar displays "localhost:3000". The main content area shows the following JSON response:

```
[  
  "_replicator",  
  "dbten",  
  "demodb",  
  "ysleadc14",  
  "yslnovel"  
]
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

4. Also if the restricted IP address is specified in CORS, the access can be secured

The screenshot shows the 'CORS' tab selected in the Cloudant Dashboard account settings. It indicates that CORS is currently enabled. Under 'Origin Domains', the 'Restrict to specific domains' option is selected, and the URL 'http://localhost:3000' is listed. A red box highlights this URL entry. A red arrow points upwards from the bottom right towards the 'Add Domain' button.

Firefox Web Browser

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#account/cors

Service Details - IBM Cloud Cloudant Dashboard - account Compliance | IBM Cloud Docs Practical 3 p2.ipynb - Colab New Tab

Account

Monitoring Announcements CORS Settings

Cross-Origin Resource Sharing (CORS) lets you connect to remote servers directly from the browser. Using CORS, you can host browser-based apps on static pages, and load data directly from Cloudant.

CORS is currently enabled.

Disable CORS

Origin Domains

Databases will accept requests from these domains:

All domains (*)
 Restrict to specific domains

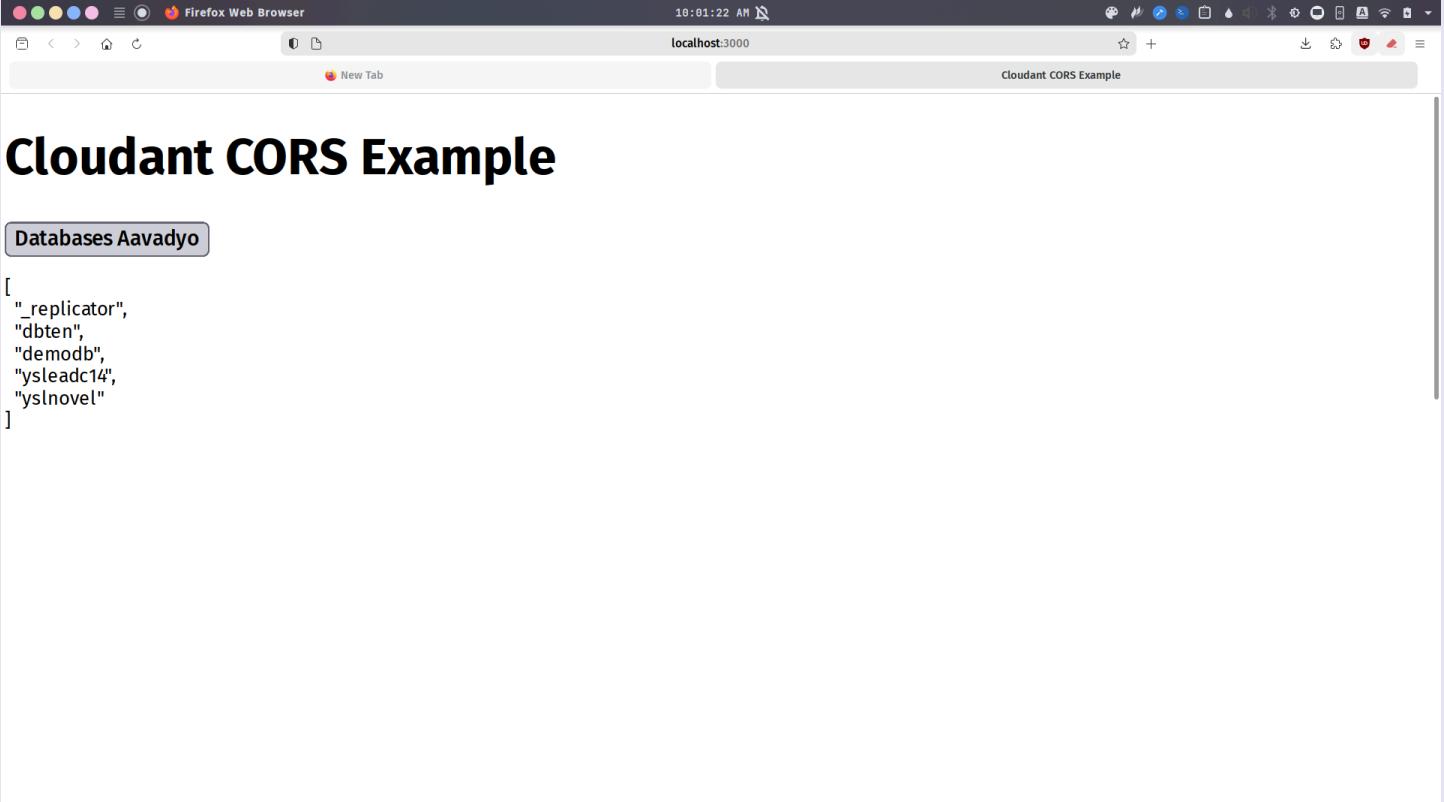
http://localhost:3000

Add Domain

IBM Cloudant

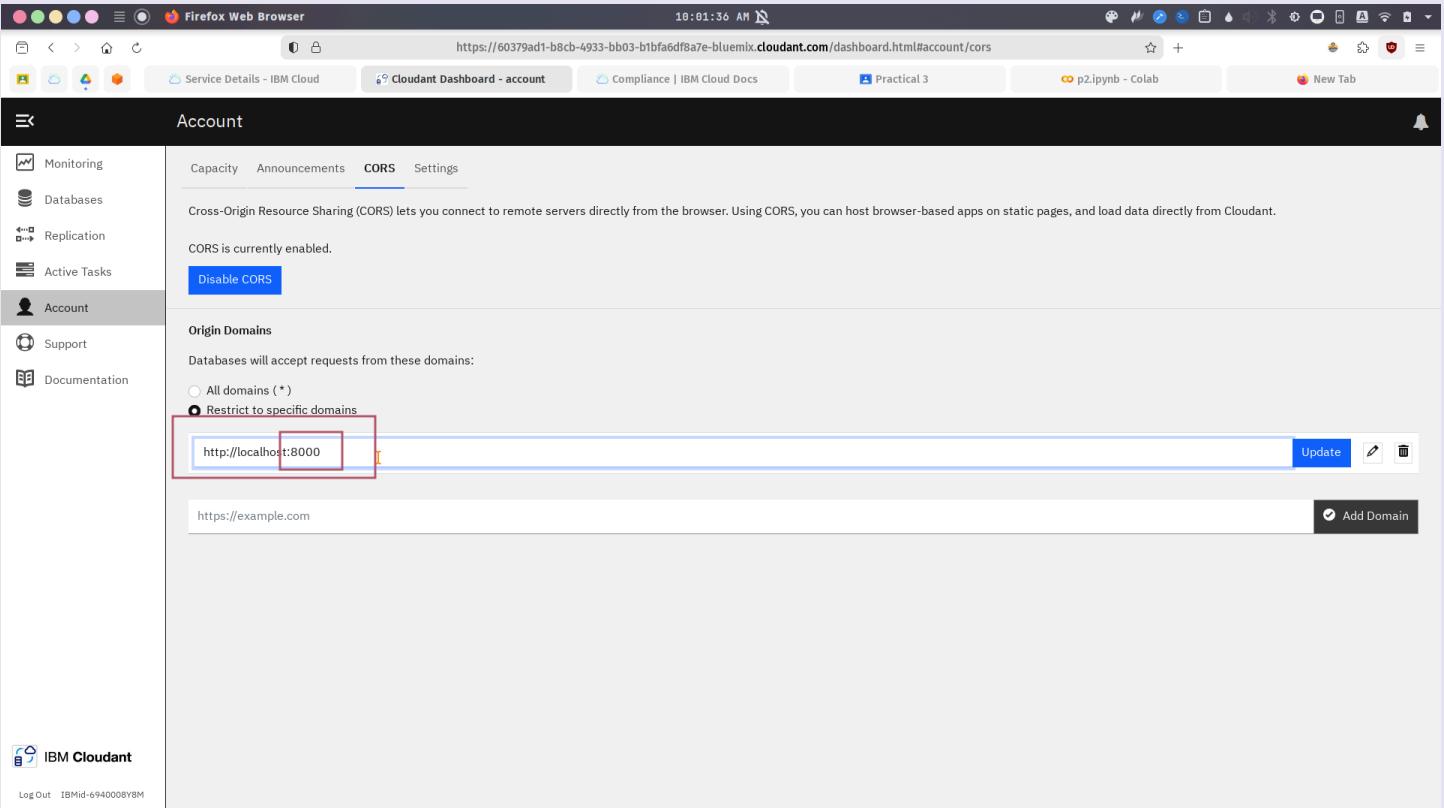
Log Out IBMid-6940008Y8M

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3



The screenshot shows a Firefox browser window with the title "Cloudant CORS Example". The address bar indicates the URL is "localhost:3000". The main content area displays a JSON array of database names:

```
[  
  "_replicator",  
  "dbten",  
  "demodb",  
  "ysleadc14",  
  "ysslNovel"  
]
```



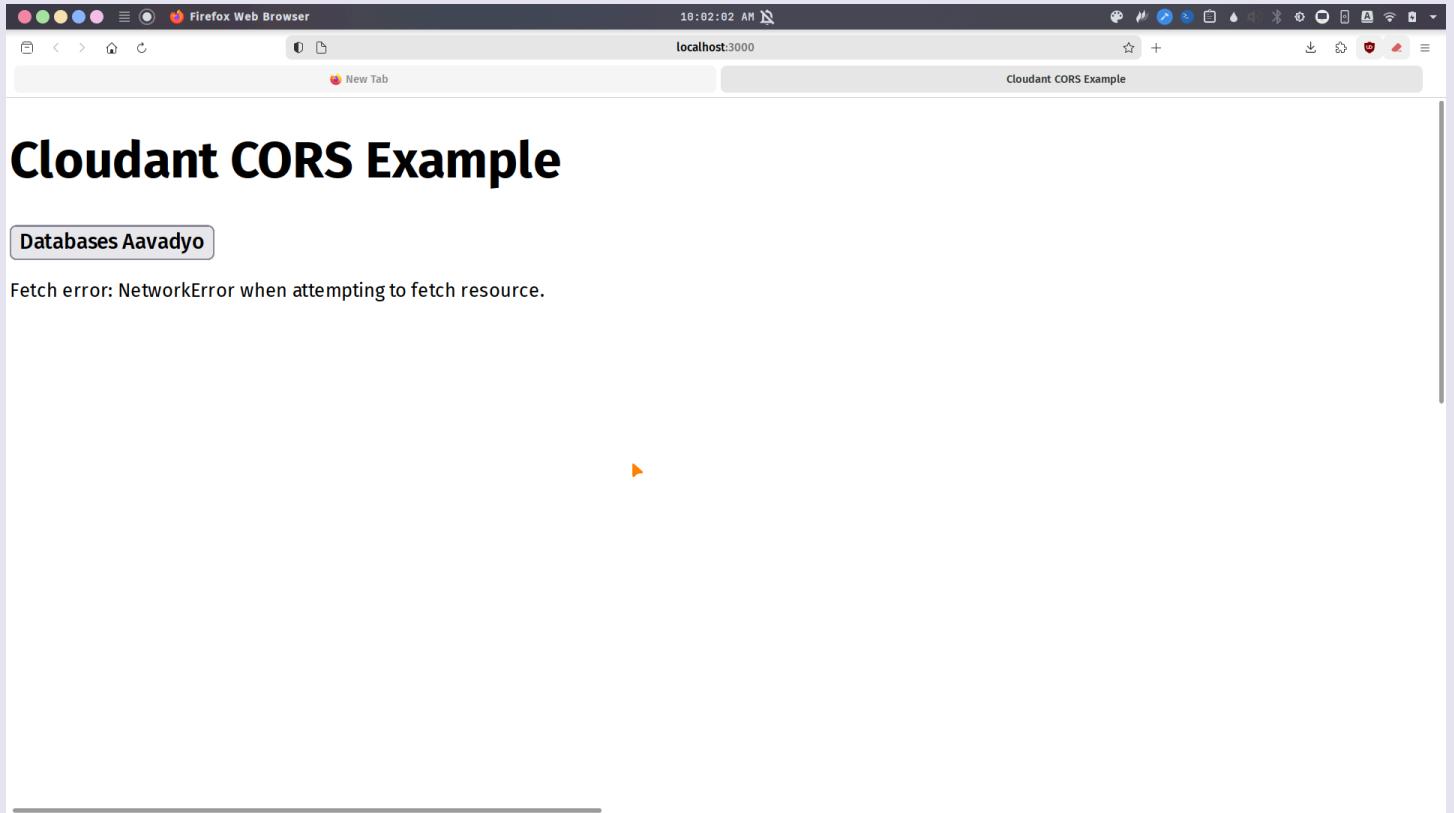
The screenshot shows the "Account" section of the Cloudant Dashboard. The left sidebar includes links for Monitoring, Databases, Replication, Active Tasks, Account, Support, and Documentation. The main content area has tabs for Capacity, Announcements, CORS (which is selected), and Settings. A note explains that CORS allows connecting to remote servers directly from the browser. It states that CORS is currently enabled and provides a "Disable CORS" button. Under "Origin Domains", it says databases will accept requests from specific domains. Two options are shown: "All domains (*)" and "Restrict to specific domains". The "Restrict to specific domains" option is selected, and a red box highlights the input field containing "http://localhost:8000". Below this, there is a link to "https://example.com" and a "Add Domain" button.

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3



(Unauthorized error due to restricted IP access)

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

5. Now, choose any database to check access through user generated API keys

The screenshot shows a Firefox browser window displaying the IBM Cloudant Dashboard. The URL is https://60379ad1-b8cb-4933-bb03-b1bfa6d8a7e-bluemix.cloudant.com/dashboard.html#/all_dbs. The dashboard has a sidebar on the left with links like Monitoring, Databases (which is selected), Replication, Active Tasks, Account, Support, and Documentation. The main area is titled 'Databases' and lists five databases: '_replicator', 'dbten', 'demodb' (highlighted with a red box), 'ysleadc14', and 'yslnovel'. Each database row includes columns for Name, Size, # of Docs, Partitioned, and Actions (with icons for edit, lock, and delete). At the bottom, there's a footer with 'IBM Cloudant' logo, 'Log Out', and 'IBMid-6940008Y8M'. The status bar at the bottom right shows 'Showing 1-5 of 5 databases. Databases per page: 20'.

Name	Size	# of Docs	Partitioned	Actions
_replicator	2.5 KB	1	No	[Edit, Lock, Delete]
dbten	351 bytes	4	No	[Edit, Lock, Delete]
demodb	277 bytes	6	No	[Edit, Lock, Delete]
ysleadc14	8.1 KB	1	No	[Edit, Lock, Delete]
yslnovel	386 bytes	4	No	[Edit, Lock, Delete]

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

6. Create new API key with necessary permissions and use with respective code

The screenshot shows the Cloudant Dashboard interface in a Firefox browser. The URL is <https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/database/demodb/permissions>. The page title is "Cloudant Dashboard - database/". The left sidebar lists various database management options: Monitoring, Databases (selected), Replication, Active Tasks, Account, Support, and Documentation. The main content area displays a table of users and API keys with permissions for the "demodb" database. One row shows the user "60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix" with checkboxes for permissions: _admin (checked), _reader (checked), _writer (checked), and _replicator (checked). Another row shows "Unauthenticated connections" with all permission checkboxes unchecked. Below the table, there is a "Grant database permissions to:" input field with "Username or API Key" placeholder text and a "Grant Permissions" button. At the bottom of the page, a note states "API keys can grant programmatic access to Cloudant databases. If you generate an API key, you can also manage that key's permissions." A red box highlights the "Generate API Key" button.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

Firefox Web Browser

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/database/demodb/permissions

Cloudant Dashboard - database /

Practical 3

Document ID { JSON

Monitoring

Databases

Replication

Active Tasks

Account

Support

Documentation

demodb

All Documents

Query

Permissions

Changes

Design Documents

Cloudant users and API keys with permissions on demodb.

Filter

60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix

apikey-646f21ee29fc4c4bac919e857a579dc

Unauthenticated connections

_admin _reader _writer _replicator

Grant database permissions to: Username or API Key

API keys can grant programmatic access to Cloudant databases. If you generate an API key, you can also manage that key's permissions.

Key: apikey-646f21ee29fc4c4bac919e857a579dc
Password: 454b0e57ed867bd8dc17103243104fc9e68c595

This message will expire in 2:58. Please make a note of the API key password; for security reasons, the password cannot be shown again.

Generate API Key

IBM Cloudant

Log Out IBMid-694000BY8M

Visual Studio Code - URL Handler

EXPLORER

PRACTICAL_3

node_modules

public

public2

index.html

index2.html

gitignore

apikey_demo

app.js

app2.js

package-lock.json

package.json

app2.js

index.html

apikey_demo

public2

index.html

index2.html

gitignore

apikey_demo

app.js

app2.js

package-lock.json

package.json

1 <!DOCTYPE html>

2 <html lang="en">

3 <head>

4 <meta charset="UTF-8">

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

6 <title>Cloudant API Example</title>

7 </head>

8 <body>

9 <h1>Cloudant API Example</h1>

10 <button id="fetchData">Databases Aavadyo</button>

11 <pre id="result"></pre>

12 <script>

13 document.getElementById('fetchData').addEventListener('click', function () {

14 const apiKey = '00655546aad648cbaf4889d37c66f0e4';

15 const apiPassword = '536c6054f940f30a839228b951b6affb784b6310';

16 const url = "https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud/demodb/_all_docs?include_docs=true";

17 const headers = new Headers();

18 headers.append('Authorization', 'Basic ' + btoa(apiKey + ':' + apiPassword));

19 fetch(url, { method: 'GET', headers: headers })

20 .then(response => response.json())

21 .then(data => {

22 document.getElementById('result').textContent = JSON.stringify(data, null, 2);

23 })

24 .catch(error => {

25 document.getElementById('result').textContent = 'Fetch error: ' + error.message;

26 })

27 });

28 <!-- #16-33 document.getElementById('fetchData').addEventListener

29 </script>

30 </body>

31 </html>

Ln 17, Col 1(145 selected) Spaces:4 LF HTML Prettier

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

Code :

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Cloudant API Example</title>
</head>

<body>
    <h1>Cloudant API Example</h1>
    <button id="fetchData">Databases Aavadyo</button>
    <pre id="result"></pre>

    <script>
        document.getElementById('fetchData').addEventListener('click',
function () {
    const apiKey = 'apikey-60655546aad648cbaf4889d37c66f0e4';
    const apiPassword = '536c6054f940f30a839228b951b6affb784b6310';
    const url =
"https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud/demodb/_all_docs?include_docs=true";

    const headers = new Headers();
    headers.append('Authorization', 'Basic ' + btoa(apiKey + ':' + apiPassword));

    fetch(url, { method: 'GET', headers: headers })
        .then(response => response.json())
        .then(data => {
            document.getElementById('result').textContent =

```

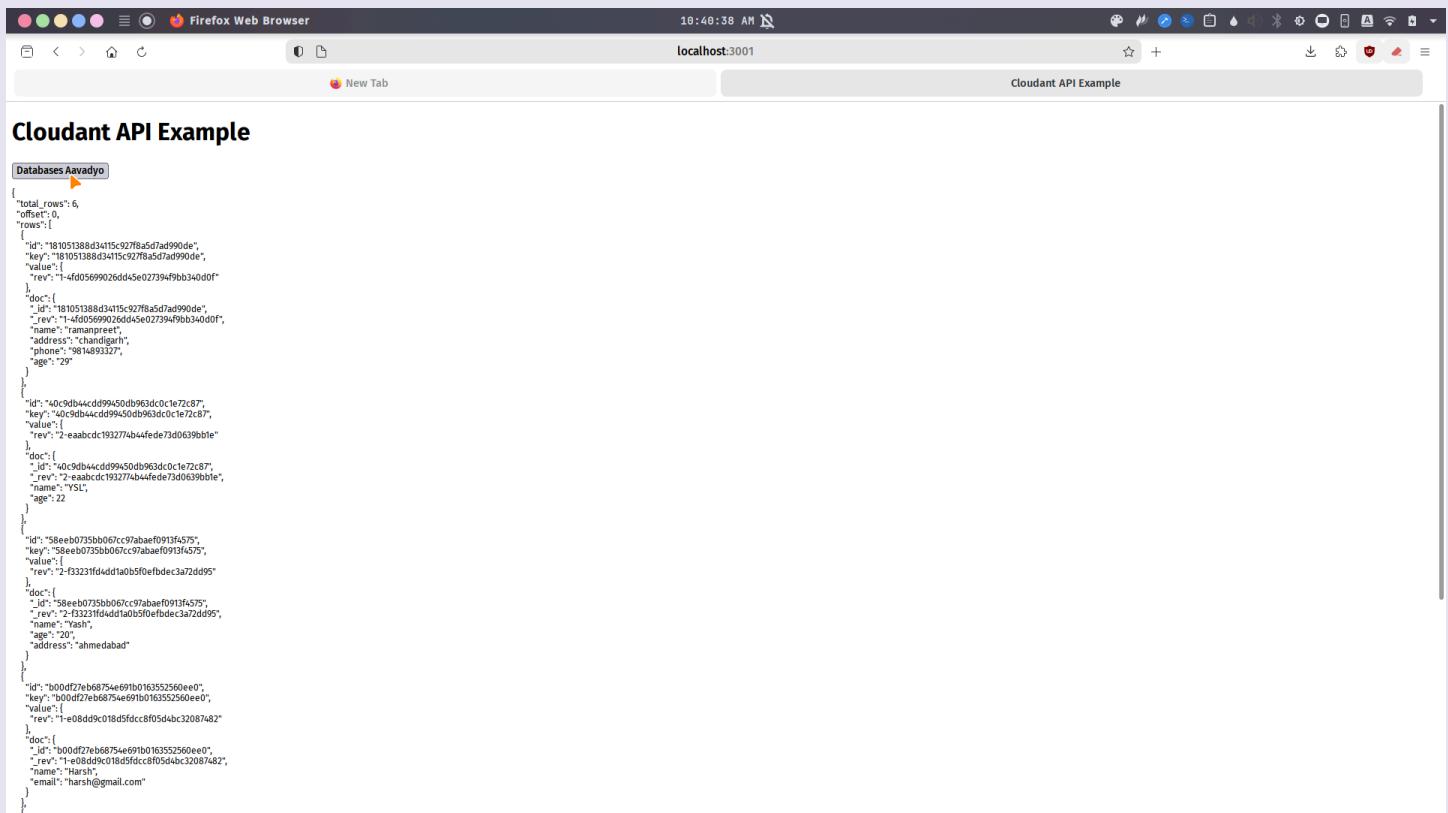
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

```
JSON.stringify(data, null, 2);

        })
        .catch(error => {
            document.getElementById('result').textContent = 'Fetch
error: ' + error.message;
        });
    });

</script>
</body>

</html>
```



Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

7. But if access is restricted through api user, then it will not be allowed to do operations (like here no read access => can't list databases)

The screenshot shows the Cloudant Dashboard interface for the 'demodb' database. On the left sidebar, under the 'Databases' section, the 'Permissions' option is selected. In the main content area, a table lists users and API keys with permissions on 'demodb'. One row is highlighted with a red box, showing the API key 'apikey-60655546aad648cbaf4889d37c66f0e4'. The permissions column for this key includes checkboxes for '_admin', '_reader', '_writer', and '_replicator'. The '_reader' checkbox is currently checked and highlighted with a red box and an orange cursor, indicating it is being unselected. Below the table, there is a 'Grant database permissions to:' input field and a 'Grant Permissions' button. A message box at the bottom provides details about API keys and their expiration.

		_admin	_reader	_writer	_replicator
60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix	apikey-60655546aad648cbaf4889d37c66f0e4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

API keys can grant programmatic access to Cloudant databases. If you generate an API key, you can also manage that key's permissions. [Generate API Key](#)

Key: **apikey-60655546aad648cbaf4889d37c66f0e4**
Password: **536c6054f940f30a839228b951b6affb784b6310** [Copy](#)

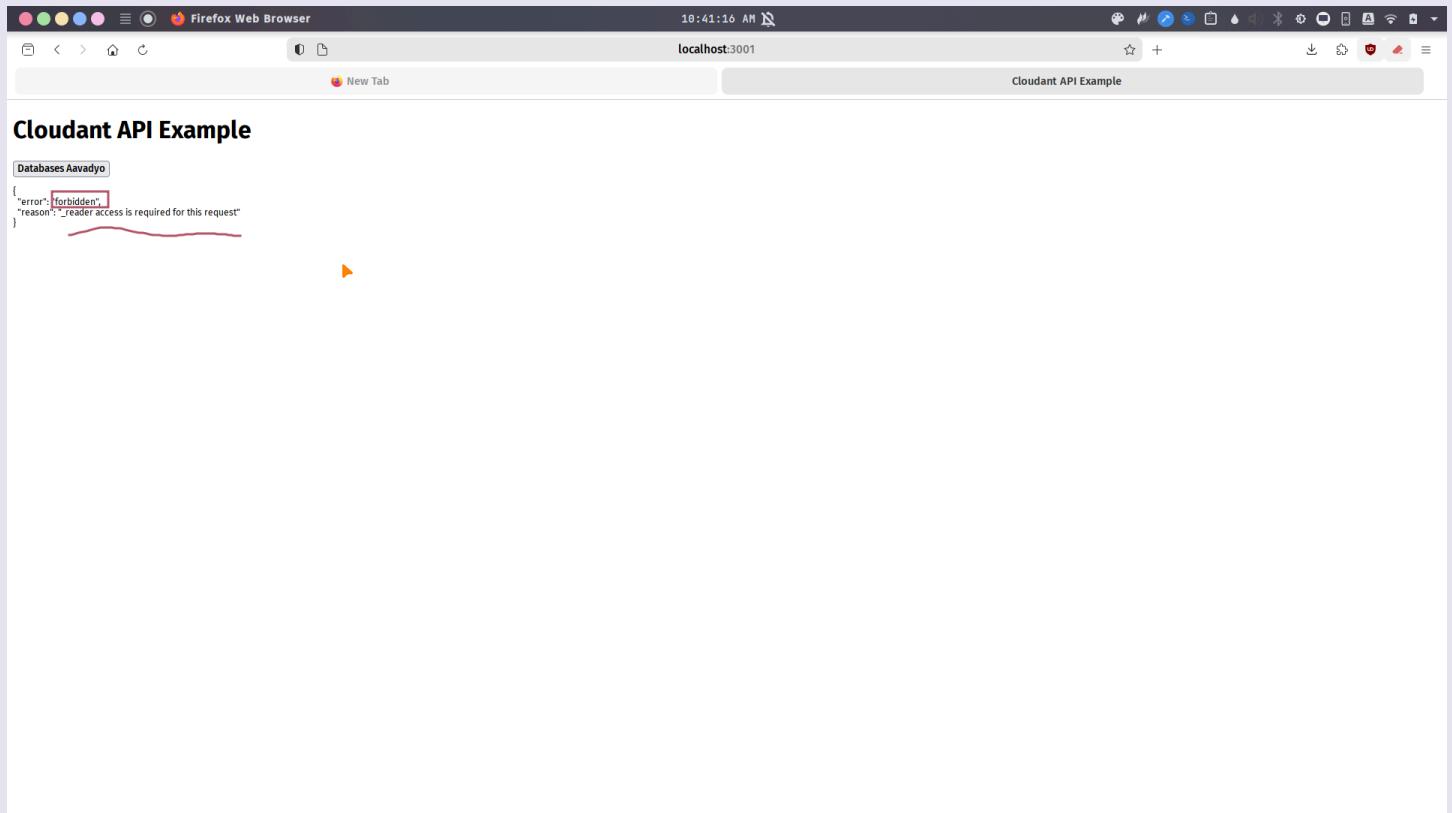
This message will expire in 2:58. Please make a note of the API key password; for security reasons, the password cannot be shown again.

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

8. Now create new database to check replication security

The screenshot shows the IBM Cloudant Dashboard in a Firefox browser. The left sidebar has links for Monitoring, Databases (which is selected), Replication, Active Tasks, Account, Support, and Documentation. The main area displays a table of existing databases:

Name	Size	# of Docs	Partitioned	Action
_replicator	2.5 KB	1	No	[Edit]
dbten	351 bytes	4	No	[Edit]
demodb	277 bytes	6	No	[Edit]
ysleadc14	8.1 KB	1	No	[Edit]
yslnovel	386 bytes	4	No	[Edit]

A modal window titled "Create Database" is open on the right. It contains a "Database name" input field with "demodb2" typed in, which is highlighted with a red box. Below it are two radio buttons: "Non-partitioned - recommended for most workloads" (selected) and "Partitioned". A question "Which should I choose?" is followed by a large red hand-drawn arrow pointing down towards the "Create" button. At the bottom of the modal are "Cancel" and "Create" buttons.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

9. Add previously generated API key user with necessary permissions to this new destination db

The screenshot shows the IBM Cloudant Dashboard for the database 'demodb2'. On the left, there's a sidebar with various options like Monitoring, Databases, Replication, Active Tasks, Account, Support, and Documentation. The main area is titled 'Permissions' under 'Databases'. It lists 'Cloudant users and API keys with permissions on demodb2'. There's a table with columns for '_admin', '_reader', '_writer', and '_replicator'. One row shows a user with all permissions checked. Below this is a section for 'Unauthenticated connections' with empty checkboxes. At the bottom, there's a text input field with the placeholder 'Grant database permissions to:' containing the value 'apikey-60655546aad648cbaf4889d37c66f0e4'. To the right of this input field is a large red-bordered button labeled 'Grant Permissions'.

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

Firefox Web Browser 10:54:37 AM

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/database/demodb2/permissions

Service Details - IBM Cloud Cloudant Dashboard - database / Compliance | IBM Cloud Docs Practical 3 p2.ipynb - Colab New Tab

demodb2

All Documents +

Databases +

Monitoring +

Replication +

Active Tasks +

Account +

Support +

Documentation +

Cloudant users and API keys with permissions on demodb2.

Filter

	_admin	_reader	_writer	_replicator
60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
apikey-60655546aad648cbaf4889d37c66f0e4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Unauthenticated connections	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Grant database permissions to: Username or API Key Grant Permissions

API keys can grant programmatic access to Cloudant databases. If you generate an API key, you can also manage that key's permissions. Generate API Key

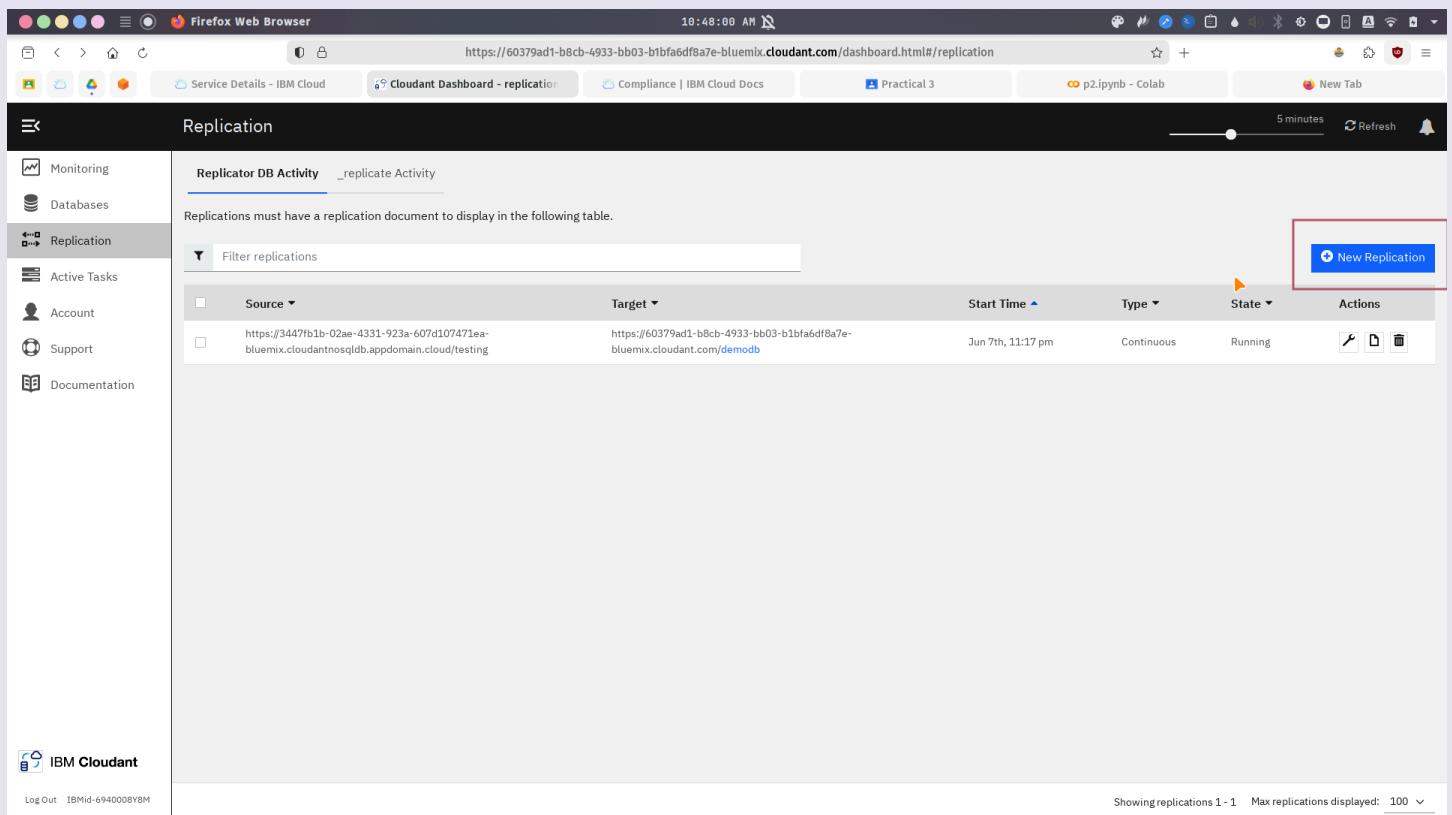
IBM Cloudant

Log Out IBMid-6940008Y8M

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/database/demodb2/_all_docs?startkey="_design"&endkey="_design0"

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

10. Perform new replication using API key user selecting both source and destination databases



The screenshot shows the IBM Cloudant Replication dashboard. On the left, there's a sidebar with icons for Monitoring, Databases, Replication (which is selected), Active Tasks, Account, Support, and Documentation. The main area has a title 'Replication' and a sub-section 'Replicator DB Activity'. It displays a table of replication activities. One row is shown in the table:

Source	Target	Start Time	Type	State	Actions
https://3447fb1b-02ae-4331-923a-607d107471ea-bluemix.cloudantnosqldb.appdomain.cloud/testing	https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/demodb	Jun 7th, 11:17 pm	Continuous	Running	

A red box highlights the 'New Replication' button at the top right of the table area. At the bottom of the dashboard, there are links for 'Log Out' and 'IBMid-6940008Y8M', and a status message 'Showing replications 1 - 1 Max replications displayed: 100'.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

Firefox Web Browser 10:51:44 AM

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/replication/_create

Service Details - IBM Cloud Cloudant Dashboard - replication Compliance | IBM Cloud Docs Practical 3 p2.ipynb - Colab New Tab

Job Configuration

Source

Type: Local database
Name: demodb
Authentication: Cloudant Username or API Key
apikey-60655546aad648cbaf4889d37c66f0e4
For security purposes, the IBM Cloudant team recommends that you use IAM API keys or IBM Cloudant legacy authentication API keys rather than account-level credentials for replication jobs.

Target

Type: Existing local database
Name: demodb2
Authentication: Cloudant Username or API Key
apikey-60655546aad648cbaf4889d37c66f0e4
For security purposes, the IBM Cloudant team recommends that you use IAM API keys or IBM Cloudant legacy authentication API keys rather than account-level credentials for replication jobs.

Options

Replication type: One time
Replication document: Custom ID (optional)

Clear Start Replication

IBM Cloudant

Log Out IBMid-6940008Y8M



Firefox Web Browser 10:53:18 AM

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/replication/_create

Service Details - IBM Cloud Cloudant Dashboard - replication Compliance | IBM Cloud Docs Practical 3 p2.ipynb - Colab New Tab

Job Configuration

Source

Type: Local database
Name: demodb
Authentication: Cloudant Username or API Key
apikey-60655546aad648cbaf4889d37c66f0e4
For security purposes, the IBM Cloudant team recommends that you use IAM API keys or IBM Cloudant legacy authentication API keys rather than account-level credentials for replication jobs.

Target

Type: Existing local database
Name: demodb2
Authentication: Cloudant Username or API Key
apikey-60655546aad648cbaf4889d37c66f0e4
For security purposes, the IBM Cloudant team recommends that you use IAM API keys or IBM Cloudant legacy authentication API keys rather than account-level credentials for replication jobs.

Options

Replication type: One time
Replication document: Custom ID (optional)

Clear Start Replication

IBM Cloudant

Log Out IBMid-6940008Y8M



Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

Firefox Web Browser 10:54:37 AM

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/database/demodb2/permissions

Service Details - IBM Cloud Cloudant Dashboard - database / Compliance | IBM Cloud Docs Practical 3 p2.ipynb - Colab New Tab

demodb2

All Documents +

Databases +

Monitoring +

Replication +

Active Tasks +

Account +

Support +

Documentation +

Cloudant users and API keys with permissions on demodb2.

Filter

	_admin	_reader	_writer	_replicator
60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
apikey-60655546aad648cbaf4889d37c66f0e4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Unauthenticated connections	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Grant database permissions to: Username or API Key Grant Permissions

API keys can grant programmatic access to Cloudant databases. If you generate an API key, you can also manage that key's permissions. Generate API Key

IBM Cloudant

Log Out IBMid-6940008Y8M

https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#/database/demodb2/_all_docs?startkey="_design"&endkey="_design0"

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

11. Now, check the same access through nodejs code by creating a new record in source db

The screenshot shows the Cloudant Dashboard for the database 'demodb2'. On the left sidebar, under 'Databases', 'Permissions' is selected. In the main area, it lists 'Cloudant users and API keys with permissions on demodb2'. There are two entries: '60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix' and 'apikey-60655546aad648cbaf4889d37c66f0e4'. Under each entry, there are four checkboxes for roles: '_admin', '_reader', '_writer', and '_replicator'. The '_replicator' checkbox for the second entry is highlighted with a red box. At the bottom, there is a 'Grant database permissions to:' input field and a 'Grant Permissions' button. A note at the bottom says 'API keys can grant programmatic access to Cloudant databases. If you generate an API key, you can also manage that key's permissions.' and a 'Generate API Key' button.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

The screenshot shows a Firefox browser window with the URL https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#database/demodb/_new. The title bar says "Cloudant Dashboard...Abase/Demodb/_new - Firefox Web Browser". The main content area is titled "demodb > New Document". A modal dialog box is open with the heading "Create Document" and a "Cancel" button. Inside the dialog, there is a JSON code editor containing the following document:

```
1 {  
2   "_id": "8b09990cc743a896b9a9da988a2348738",  
3   "name": "haribol"  
4 }
```

The sidebar on the left includes icons for user profile, database, and other dashboard options. At the bottom left of the dashboard area, there is a "Log Out" link.

Code :

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Cloudant API Example</title>  
</head>  
<body>  
  <h1>Cloudant API Example</h1>  
  <button id="replicateData">Replicate Data</button>  
  <pre id="result"></pre>  
  
<script>  
  document.getElementById('replicateData').addEventListener('click',
```

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

```
function() {
    const apiKey = 'apikey-60655546aad648cbaf4889d37c66f0e4';
    const apiPassword = '536c6054f940f30a839228b951b6affb784b6310';
    const sourceUrl =
'https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud/demodb/_all_docs?include_docs=true';
    const targetUrl =
'https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosqldb.appdomain.cloud/demodb2';

    const headers = new Headers();
    headers.append('Authorization', 'Basic ' + btoa(apiKey + ':' + apiPassword));
    headers.append('Content-Type', 'application/json');

    fetch(sourceUrl, { method: 'GET', headers: headers })
        .then(response => response.json())
        .then(data => {
            const docs = data.rows.map(row => row.doc);
            docs.forEach(doc => {
                replicateDocument(doc, targetUrl, headers);
            });
        })
        .catch(error => {
            document.getElementById('result').textContent =
'Replication error: ' + error;
        });
    });

    function replicateDocument(doc, targetUrl, headers) {
        const docUrl = `${targetUrl}/${doc._id}`;
        fetch(docUrl, { method: 'GET', headers: headers })
    }
}
```

Name - Yash Lakhtariya

Enrollment number - 21162101012

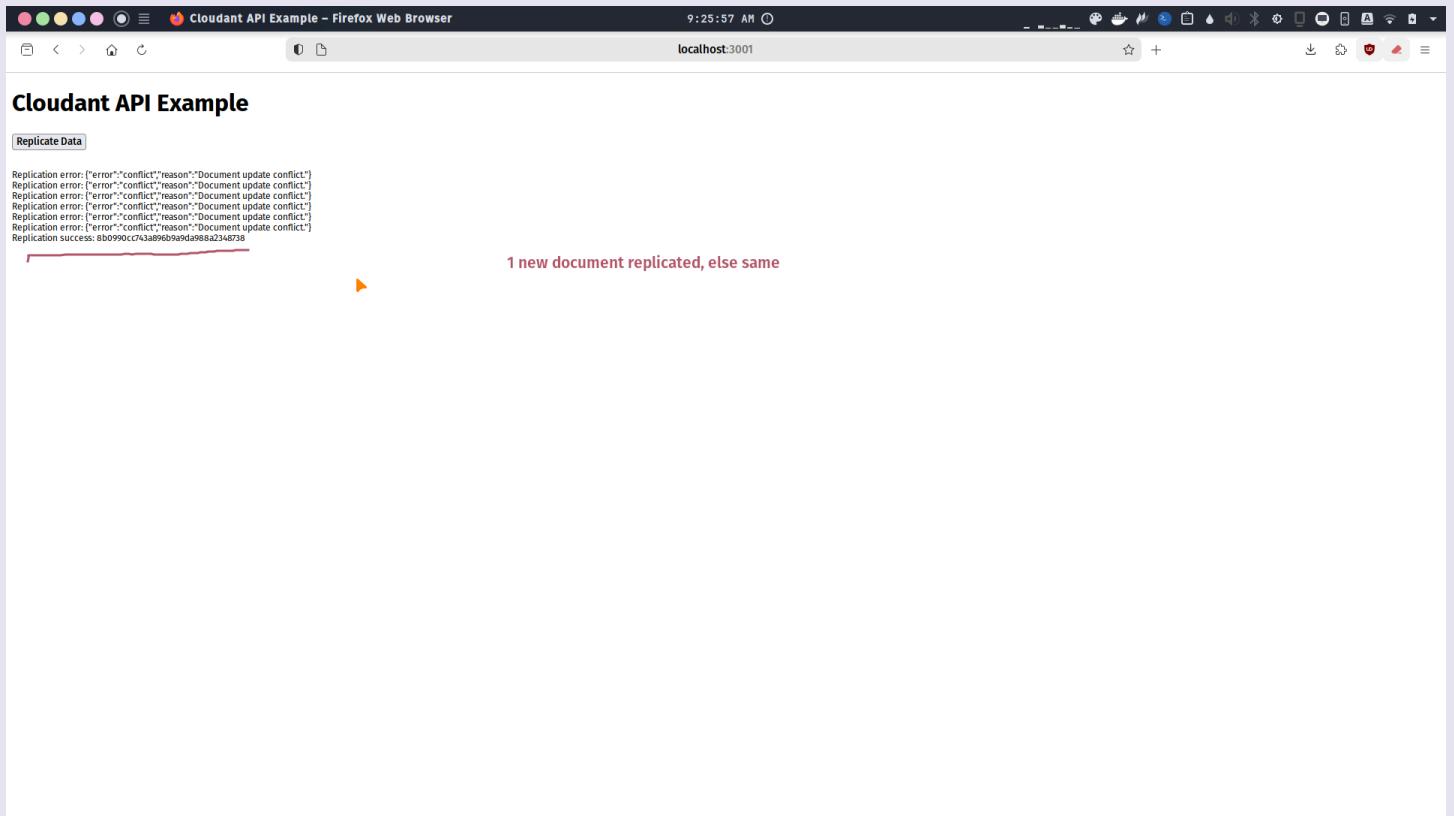
Branch - CBA Batch - 71

CS Practical 3

```
.then(response => {
    if (response.status === 404) {
        // Document does not exist, create it
        return fetch(targetUrl, {
            method: 'POST',
            headers: headers,
            body: JSON.stringify(doc)
        });
    } else {
        return response.json().then(existingDoc => {
            // Document exists, update it
            doc._rev = existingDoc._rev;
            return fetch(docUrl, {
                method: 'PUT',
                headers: headers,
                body: JSON.stringify(doc)
            });
        });
    }
})
.then(response => response.json())
.then(data => {
    if (data.ok) {
        document.getElementById('result').textContent +=
`\nReplication success: ${doc._id}`;
    } else {
        document.getElementById('result').textContent +=
`\nReplication error: ${JSON.stringify(data)}`;
    }
})
.catch(error => {
    document.getElementById('result').textContent +=
`\nReplication error: ${error.message}`;
})
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

```
});  
}  
</script>  
</body>  
</html>
```



(As seen here, duplicates are preserved and new one is replicated)

Note : reader access for source and writer-replicator access for destination is needed as seen here for successful replication

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

12. Now, for object storage part, create new roles as specified with HMAC credentials in existing instance : Manager, Reader, Object Writer

The screenshot shows the IBM Cloud Object Storage interface. A modal window titled "Create Credentials" is open in the center. The "Name:" field contains "Manageryo" and the "Role:" dropdown is set to "Manager". A red box highlights the "Key name" dropdown, which lists several existing entries like "WDP-Project-Man" and "admin-97c14b2". A red arrow points from the "Key name" dropdown towards the "Role:" dropdown. The "Add" button at the bottom right of the modal is highlighted with a red box. In the background, a list of existing credentials is visible, showing names like "fordb2", "editor-97c14b2", and "eadexam", each with a timestamp and a delete icon.

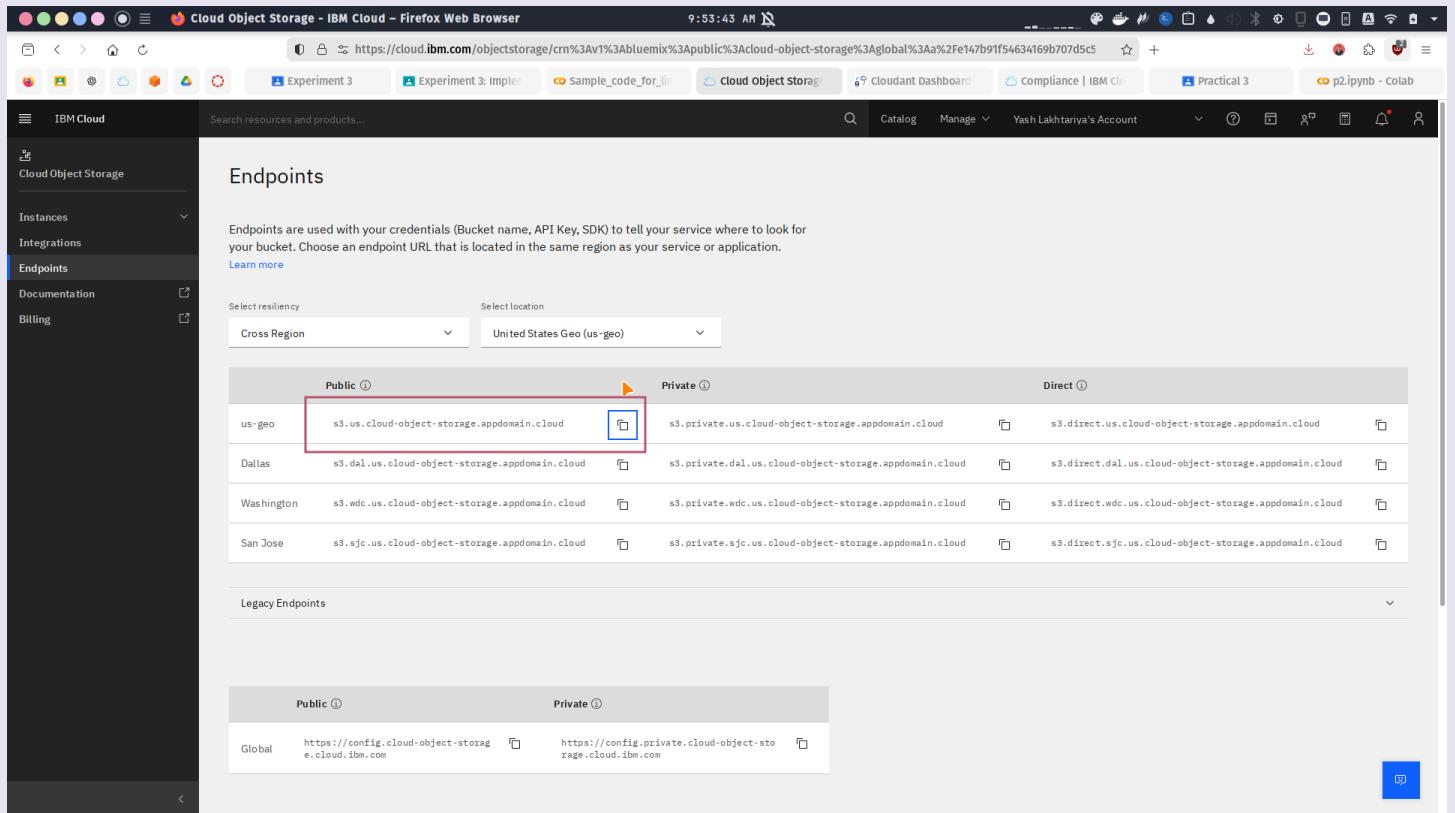
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

The screenshot shows the 'Cloud Object Storage - IBM Cloud - Firefox Web Browser' interface. The URL is <https://cloud.ibm.com/objectstorage/crn%3Av1%3Abluemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c5>. The page title is 'Cloud Object Storage'. The main content area shows 'Instances / Cloud Object Storage-ysl'. A modal window titled 'Create Credentials' is open, prompting for a 'Name' (set to 'Saado Reader') and a 'Role' (set to 'Reader'). Below these fields is a dropdown for 'Select Service ID (Optional)' with 'Auto Generated' selected. A toggle switch for 'Include HMAC Credential' is set to 'On'. At the bottom of the modal are 'Cancel' and 'Add' buttons. In the background, a list of existing credentials is visible, including 'WDP-Project-Main', 'admin-97c14b2...', 'editor-97c14b2...', 'WDP-Viewer-reader', 'fordb2', 'VadadVastuSeva', 'WDP-Editor-eadcp17-donotdelete-pr-oxqxxpro7l2uo7', 'eadexam', and 'Manageryo'. The timestamp at the top right is 9:46:18 AM.

The screenshot shows the same 'Cloud Object Storage - IBM Cloud - Firefox Web Browser' interface. The URL is the same as the previous screenshot. The main content area shows 'Instances / Cloud Object Storage-ysl'. A modal window titled 'Create Credentials' is open, prompting for a 'Name' (set to 'Obj Lekhak') and a 'Role' (set to 'Object Writer'). Below these fields is a dropdown for 'Select Service ID (Optional)' with 'Auto Generated' selected. A toggle switch for 'Include HMAC Credential' is set to 'On'. At the bottom of the modal are 'Cancel' and 'Add' buttons. In the background, a list of existing credentials is visible, including 'WDP-Project-Main', 'admin-97c14b2...', 'editor-97c14b2...', 'fordb2', 'Saado Reader', 'VadadVastuSeva', 'WDP-Editor-eadcp17-donotdelete-pr-oxqxxpro7l2uo7', and 'admin-68b79e49-bb70-4a24-8442-a6b966313d73'. The timestamp at the top right is 9:47:03 AM.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

13. Copy the location URL to access through third party apps like Postman

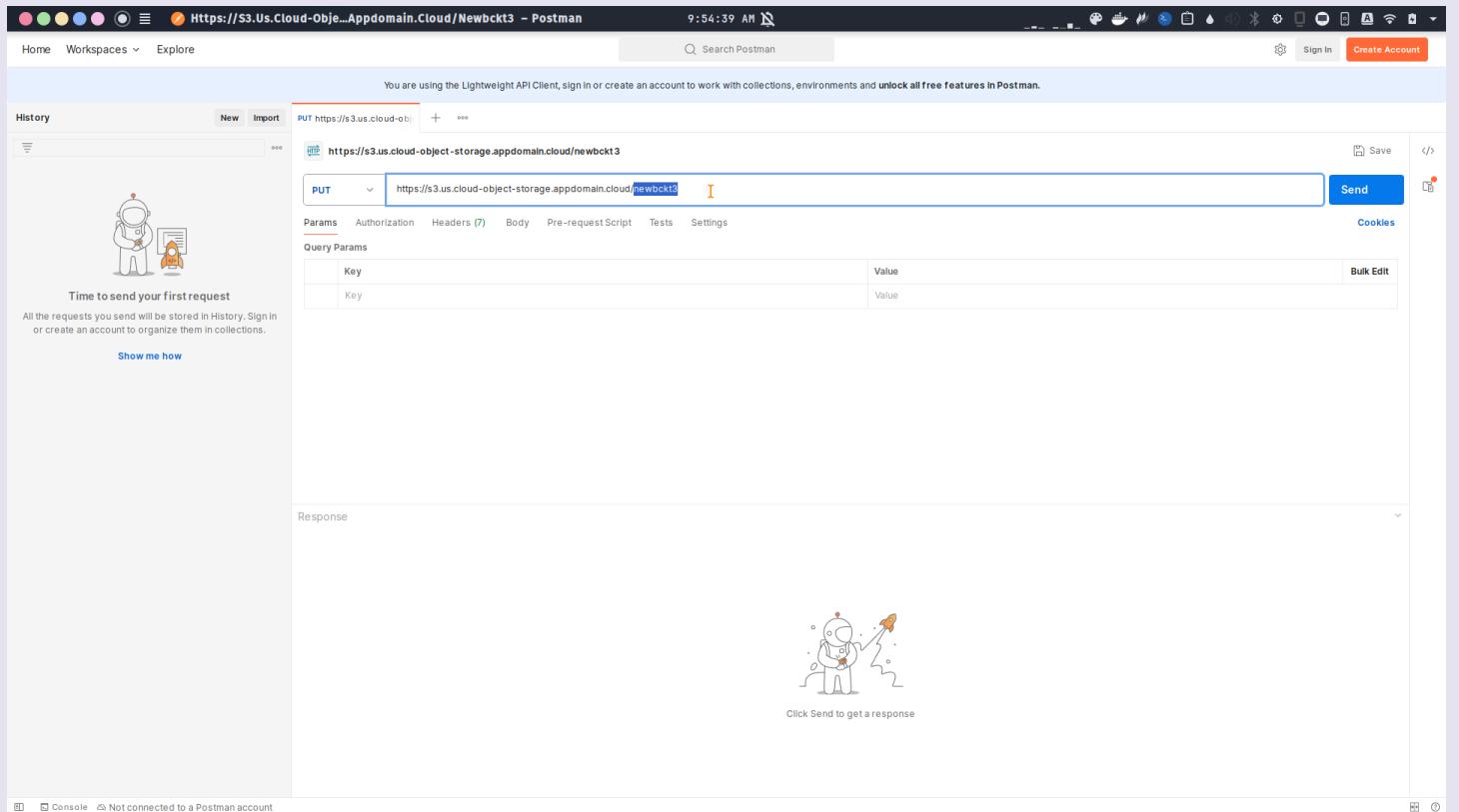


The screenshot shows the IBM Cloud Cloud Object Storage Endpoints page. The left sidebar is collapsed, and the main content area is titled "Endpoints". A sub-header states: "Endpoints are used with your credentials (Bucket name, API Key, SDK) to tell your service where to look for your bucket. Choose an endpoint URL that is located in the same region as your service or application." Below this is a "Learn more" link. The "Select resiliency" dropdown is set to "Cross Region" and the "Select location" dropdown is set to "United States Geo (us-geo)". The main table has three columns: "Public" (highlighted with a red box and a blue arrow pointing to it), "Private", and "Direct". The "Public" column lists endpoints for us-geo, Dallas, Washington, and San Jose. The "Private" and "Direct" columns list corresponding private and direct endpoints for each location. Below this table is a "Legacy Endpoints" section which is currently empty. At the bottom right of the main content area is a blue "Copy" button.

Region	Public	Private	Direct
us-geo	s3.us.cloud-object-storage.appdomain.cloud	s3.private.us.cloud-object-storage.appdomain.cloud	s3.direct.us.cloud-object-storage.appdomain.cloud
Dallas	s3.dal.us.cloud-object-storage.appdomain.cloud	s3.private.dal.us.cloud-object-storage.appdomain.cloud	s3.direct.dal.us.cloud-object-storage.appdomain.cloud
Washington	s3.wdc.us.cloud-object-storage.appdomain.cloud	s3.private.wdc.us.cloud-object-storage.appdomain.cloud	s3.direct.wdc.us.cloud-object-storage.appdomain.cloud
San Jose	s3.sjc.us.cloud-object-storage.appdomain.cloud	s3.private.sjc.us.cloud-object-storage.appdomain.cloud	s3.direct.sjc.us.cloud-object-storage.appdomain.cloud

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

14. Try creating new bucket through post request on copied URL/bucketname



The screenshot shows the Postman application interface. At the top, the URL is set to `PUT https://s3.us.cloud-object-storage.appdomain.cloud/newbckt3`. The request method is selected as `PUT`. Below the URL, there are tabs for `Params`, `Authorization`, `Headers (7)`, `Body`, `Pre-request Script`, `Tests`, and `Settings`. The `Body` tab is currently active. A table under the `Query Params` section has one row with a single column labeled `Key` and `Value`. On the right side of the interface, there is a `Send` button and a `Cookies` section. The main body area is titled `Response` and contains a small illustration of an astronaut pointing at a cloud-like shape with a speech bubble. Below the illustration, the text `Click Send to get a response` is visible. The bottom left corner of the interface shows the status `Not connected to a Postman account`.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

The screenshot shows a Postman interface with the following details:

- Request URL:** PUT https://s3.us.cloud-object-storage.appdomain.cloud/newbckt3
- Status:** 403 Forbidden
- Headers:** (7)
- Body:** (Pretty) XML response showing an error message:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>Anonymous bucket creation not allowed.</Message>
  <Resource>/newbckt3/<Resource>
  <RequestId>5664edf8-e73-4424-844a-d726f20e0e48</RequestId>
  <httpStatusCode>403</httpStatusCode>
</Error>
```
- Time:** 1981 ms
- Size:** 548 B

(Access is denied because IBM cloud security prevents unauthorized access)

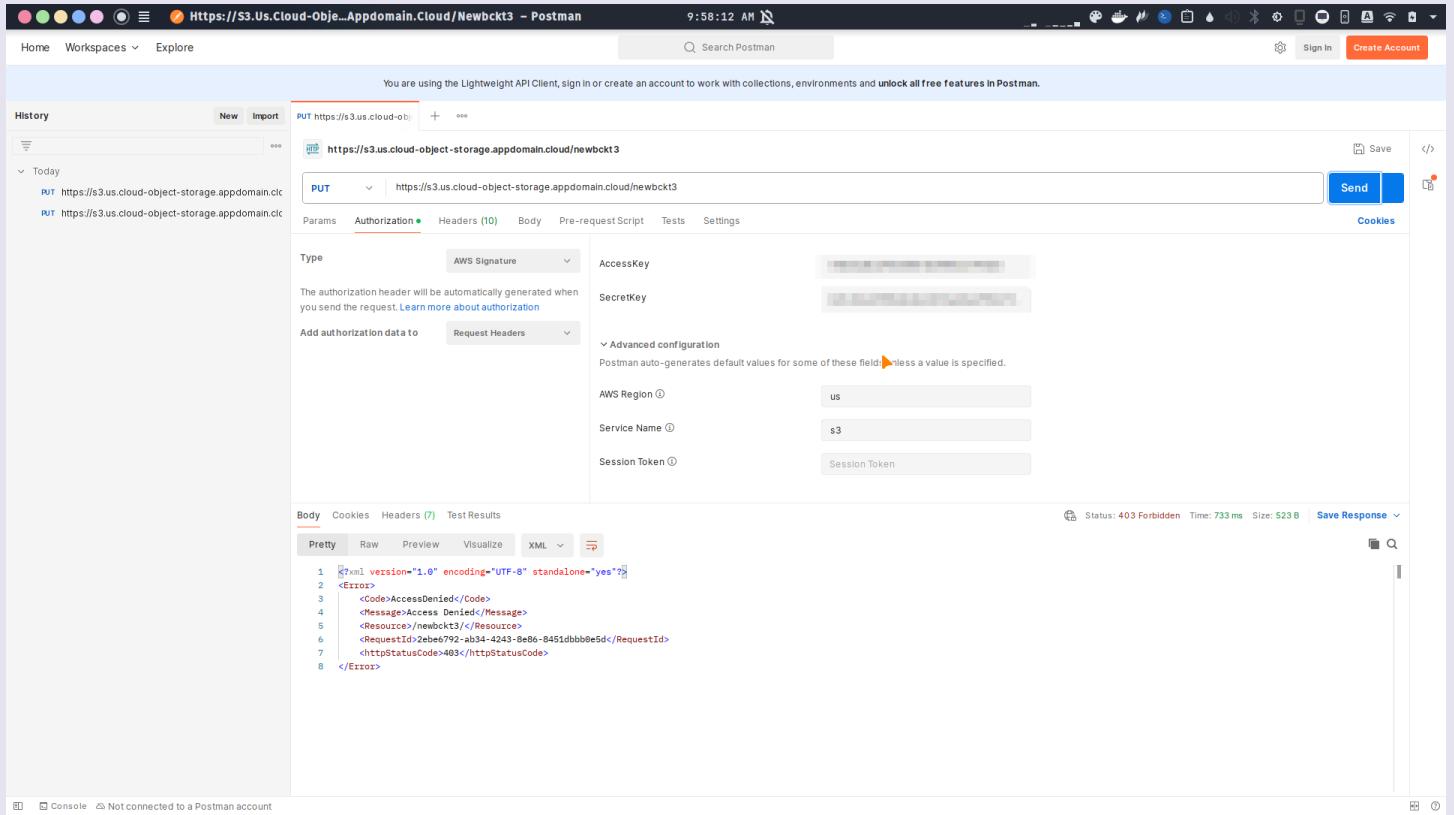
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

15. It should also fail for only reader role

The screenshot shows the 'Service credentials' tab of the Cloud Object Storage instance 'Cloud Object Storage-ysl'. A new credential named 'Saado Reader' has been created and is listed. The credential details are shown in a modal window:

```
[{"name": "Saado Reader", "access_key_id": "75845a5b474f433f81ab99f52c794b6", "secret_access_key": "8d1c9dedf35fb8e2bd492bad9e4f5fb473344331663d9336", "endpoints": "https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints", "iam_apikey_description": "Auto-generated for key crn:v1:bluemix:public:cloud-object-storage:global:a/e147b91f54634469b707d5c5d51b9c77:c2bbbeef9-875f-4e7c-a336-4397d819b138:resource-b-474f-43ff-81ab-99f52c794b6"}]
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3



The screenshot shows the Postman application interface. The URL in the header is `Https://S3.us.Cloud-Obj...Appdomain.Cloud/Newbckt3 - Postman`. The status bar indicates the time is 9:58:12 AM. The main window displays a failed `PUT` request to `https://s3.us.cloud-object-storage.appdomain.cloud/newbckt3`. The request type is set to `AWS Signature`, with `AccessKey` and `SecretKey` fields filled with placeholder values. The `AWS Region` is set to `us` and the `Service Name` is `s3`. The response body shows an XML error message:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <Resource>/newbckt3/<Resource>
  <RequestId>2ebef792-ab34-4243-8e86-8451dbbb0e5d</RequestId>
  <httpStatusCode>403</httpStatusCode>
</Error>
```

The status bar at the bottom right shows `Status: 403 Forbidden Time: 733 ms Size: 523 B Save Response`.

(Authorization should be of type AWS Signature for integration of apikey and user password for IBM Cloud Object Storage service)

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

16. Create Writer role and check, the new bucket can be successfully created

The screenshot shows the 'Create Credentials' dialog box overlaid on the IBM Cloud Object Storage interface. The dialog has the following fields:

- Name:** Purepuro Lekhak (highlighted with a red box)
- Role:** Writer (highlighted with a red box)
- Select Service ID (Optional):** Auto Generated
- Include HMAC Credential:** On (highlighted with a red box)

Below the dialog, the main interface shows a list of existing credentials:

Created	Action
2024-04-22 7:52 AM	[Delete]
2024-04-12 9:48 AM	[Delete]
2024-04-22 8:47 AM	[Delete]
2024-04-22 8:47 AM	[Delete]

At the bottom of the dialog, there are 'Cancel' and 'Add' buttons.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

Cloud Object Storage - IBM Cloud - Firefox Web Browser 10:01:28 AM

https://cloud.ibm.com/objectstorage/crn%3Av1%3Abuemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c

Experiment 3 Experiment 3: Implementation Sample_code_for_IBM_Cloud Cloud Object Storage Cloudant Dashboard Compliance | IBM Cloud Practical 3 p2.ipynb - Colab

IBM Cloud Search resources and products... Catalog Manage Yash Lakhtariya's Account Details Actions...

Cloud Object Storage Instances / Cloud Object Storage-ysl

Buckets Service credentials Instance Usage Plan

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. [Learn more](#)

Search: lekhak

Key name	Date created	Action
Obj Lekhak	2024-08-12 9:48 AM	Copy
Purepuro Lekhak	2024-08-12 10:01 AM	Copy

Show more ▾

Items per page: 25 1–2 of 2 items

1 1 of 1 page

Cloud Object Storage - IBM Cloud - Firefox Web Browser 10:01:28 AM

https://cloud.ibm.com/objectstorage/crn%3Av1%3Abuemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c

Experiment 3 Experiment 3: Implementation Sample_code_for_IBM_Cloud Cloud Object Storage Cloudant Dashboard Compliance | IBM Cloud Practical 3 p2.ipynb - Colab

IBM Cloud Search resources and products... Catalog Manage Yash Lakhtariya's Account Details Actions...

Cloud Object Storage Instances / Cloud Object Storage-ysl

Buckets Service credentials Instance Usage Plan

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. [Learn more](#)

Search: lekhak

Key name	Date created	Action
Obj Lekhak	2024-08-12 9:48 AM	Copy
Purepuro Lekhak	2024-08-12 10:01 AM	Copy

Show more ▾

Items per page: 25 1–2 of 2 items

1 1 of 1 page

Cloud Object Storage - IBM Cloud - Firefox Web Browser 10:01:28 AM

https://cloud.ibm.com/objectstorage/crn%3Av1%3Abuemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c

Experiment 3 Experiment 3: Implementation Sample_code_for_IBM_Cloud Cloud Object Storage Cloudant Dashboard Compliance | IBM Cloud Practical 3 p2.ipynb - Colab

IBM Cloud Search resources and products... Catalog Manage Yash Lakhtariya's Account Details Actions...

Cloud Object Storage Instances / Cloud Object Storage-ysl

Buckets Service credentials Instance Usage Plan

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. [Learn more](#)

Search: lekhak

Key name	Date created	Action
Obj Lekhak	2024-08-12 9:48 AM	Copy
Purepuro Lekhak	2024-08-12 10:01 AM	Copy

Show more ▾

Items per page: 25 1–2 of 2 items

1 1 of 1 page

Cloud Object Storage - IBM Cloud - Firefox Web Browser 10:03:40 AM

https://S3.U.S.Cloud-Object-Storage.Appdomain.Cloud/Newbckt3 - Postman 10:03:40 AM

Home Workspaces Explore Search Postman Sign In Create Account

You are using the Lightweight API Client, sign in or create an account to work with collections, environments and [unlock all free features in Postman](#).

History New Import

PUT https://s3.us.cloud-object-storage.appdomain.cloud/newbckt3

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

Type AWS Signature AccessKey SecretKey

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

Add authorization data to Request Headers

Advanced configuration Postman auto-generates default values for some of these fields unless a value is specified.

AWS Region us Service Name s3 Session Token Session Token

Send

Status: 200 OK Time: 1798 ms Size: 230 B Save Response

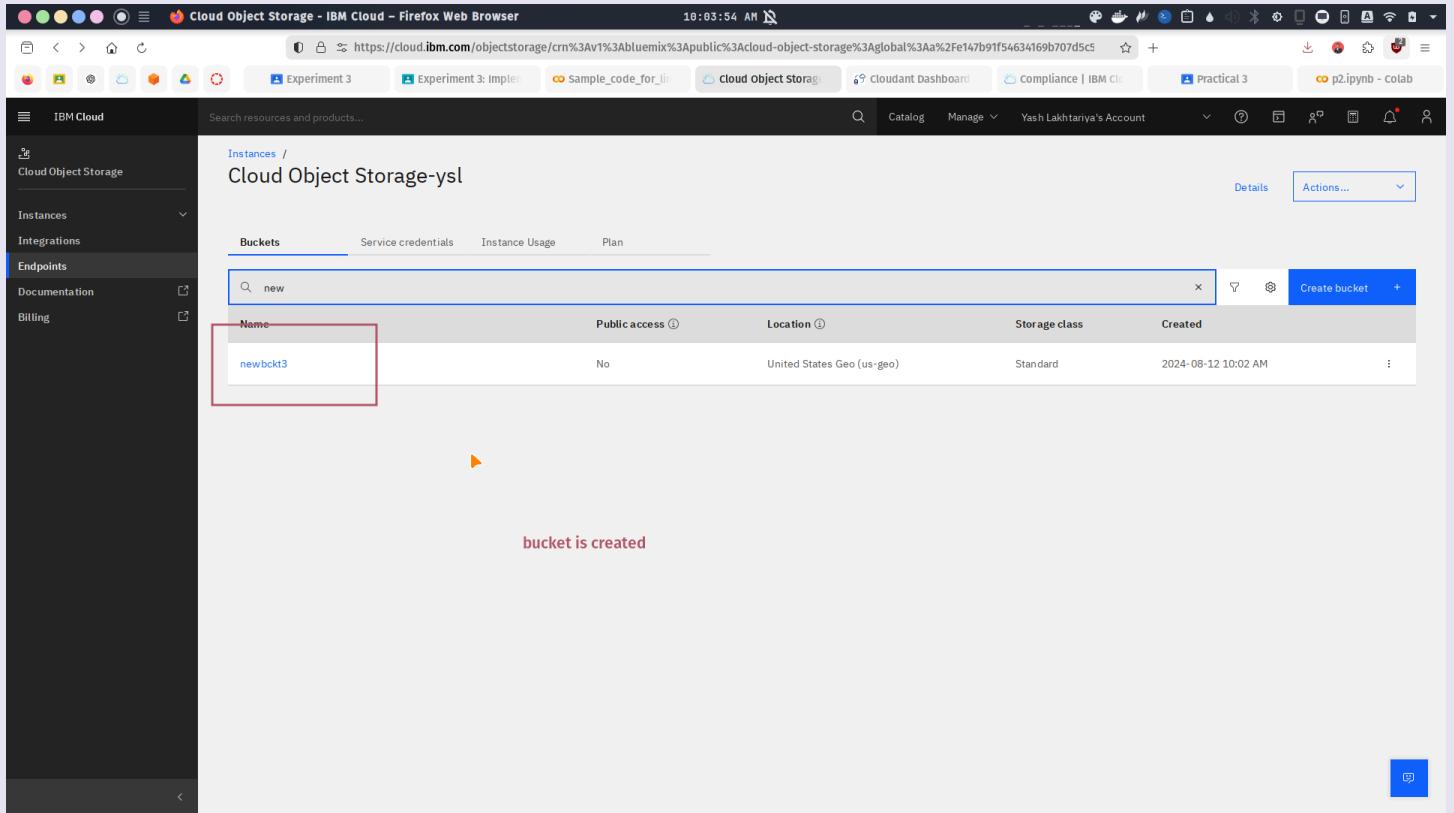
Body Cookies Headers (6) Test Results

Pretty Raw Preview Visualize Text

1

Console Not connected to a Postman account

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3



The screenshot shows the IBM Cloud Object Storage interface. On the left, there's a sidebar with options like Instances, Integrations, Endpoints, Documentation, and Billing. The main area is titled "Cloud Object Storage-ysl" and shows a table of buckets. A new bucket named "newbckt3" is listed, highlighted with a red box. The table columns are Name, Public access, Location, Storage class, and Created. The "newbckt3" row has "No" under Public access, "United States Geo (us-geo)" under Location, "Standard" under Storage class, and "2024-08-12 10:02 AM" under Created. A message "bucket is created" is displayed below the table.

Name	Public access	Location	Storage class	Created
newbckt3	No	United States Geo (us-geo)	Standard	2024-08-12 10:02 AM

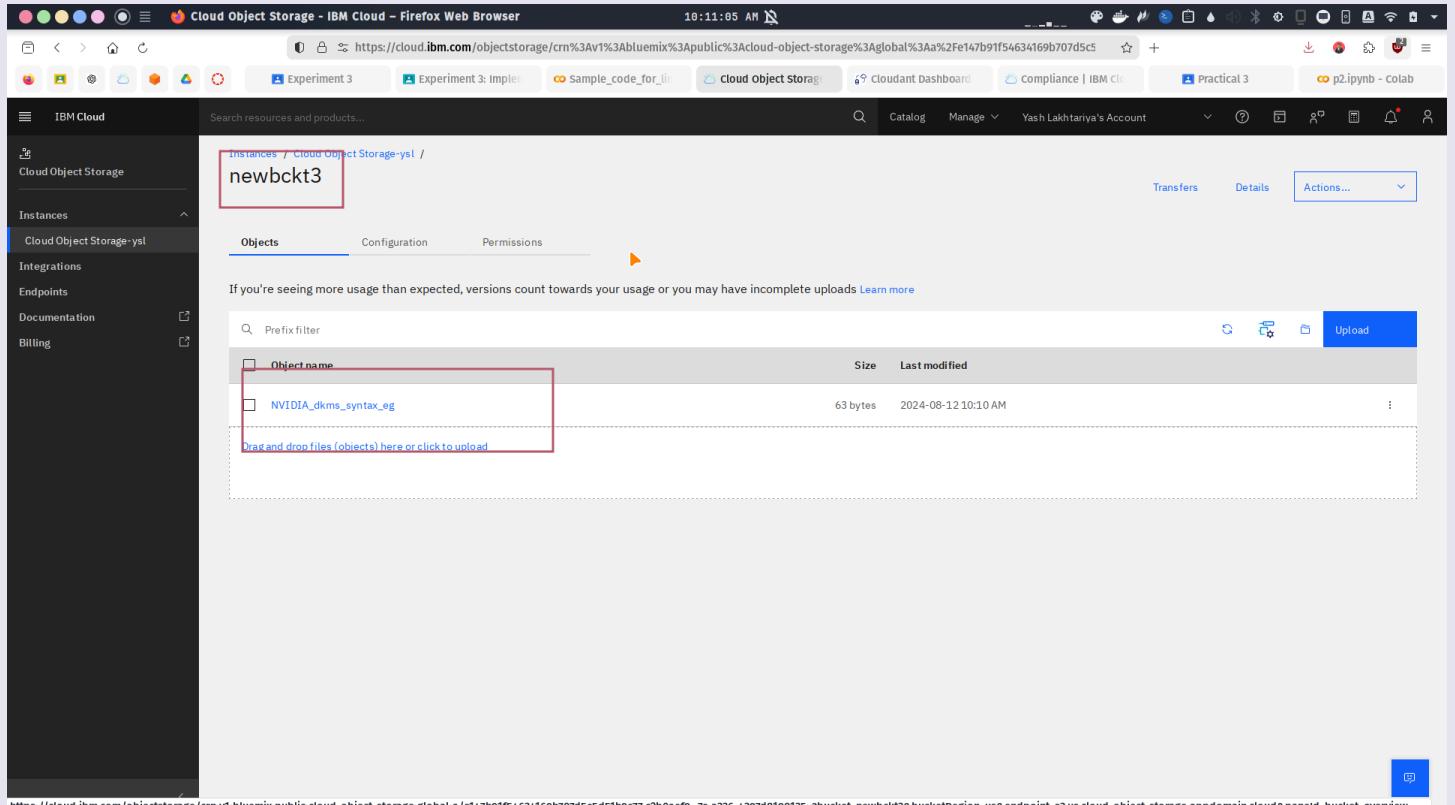
bucket is created

(As listed here, the new bucket is created)

(It can only be created by Writer or Manager role)

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

17. Add any file as object in the new bucket (say text file here)



The screenshot shows the IBM Cloud Object Storage interface. On the left, a sidebar lists 'IBM Cloud' services: Instances, Integrations, Endpoints, Documentation, and Billing. Under 'Instances', 'Cloud Object Storage-ysl' is selected. The main area shows a list of objects in the 'newbckt3' bucket. A red box highlights the bucket name 'newbckt3' in the path bar. Another red box highlights the file 'NVIDIA_dkms_syntax_eg' in the list. The file details are: Size: 63 bytes, Last modified: 2024-08-12 10:10 AM. Below the list is a placeholder text: 'Drag and drop files (objects) here or click to upload'.

Instances / Cloud Object Storage-ysl / newbckt3

Objects Configuration Permissions

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

Object name	Size	Last modified
NVIDIA_dkms_syntax_eg	63 bytes	2024-08-12 10:10 AM

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

https://cloud.ibm.com/objectstorage/crn:v1:bluemix:public:cloud-object-storage:global:a/e147b91f54634169b707d5c5d51b9c77:c2b0ae9...7c-a336-4397d8190135::bucket=newbckt3&bucketRegion=us&endpoint=s3.us.cloud-object-storage.appdomain.cloud&panelId=bucket_overview

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

18. Try getting its contents via GET request on URL/bucket/object

The screenshot shows the Postman application interface. The left sidebar displays a workspace with various collections and environments. The main area shows a GET request to the URL `https://s3.us.cloud-object-storage.appdomain.cloud/newbckt3/NVIDIA_dkms_syntax_eg`. The 'Authorization' tab is selected, showing 'AWS Signature' as the auth type. The 'Headers' tab lists several headers, including 'Content-Type: application/json'. The 'Body' tab contains the JSON payload:

```
dkms install --no-depmod nvidia/515.48.07 -k 5.18.2-zeni-1-zen
```

. The response status is 200 OK, with a time of 2.25 s and a size of 444 B. The text inside the specified object file is highlighted in red.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

19. Now, if multiple versions of same object is needed, it can be enabled in configuration

The screenshot shows the 'Object versioning' configuration page in the IBM Cloud Object Storage interface. The 'Enable' radio button is selected. A tooltip message is displayed below the button: 'After enabling versioning for your bucket:

- Enabling versioning may increase your bucket usage (bytes) as all versions of objects will be retained until deleted by the user.

' The 'Cancel' and 'Save' buttons are visible at the top right of the modal.

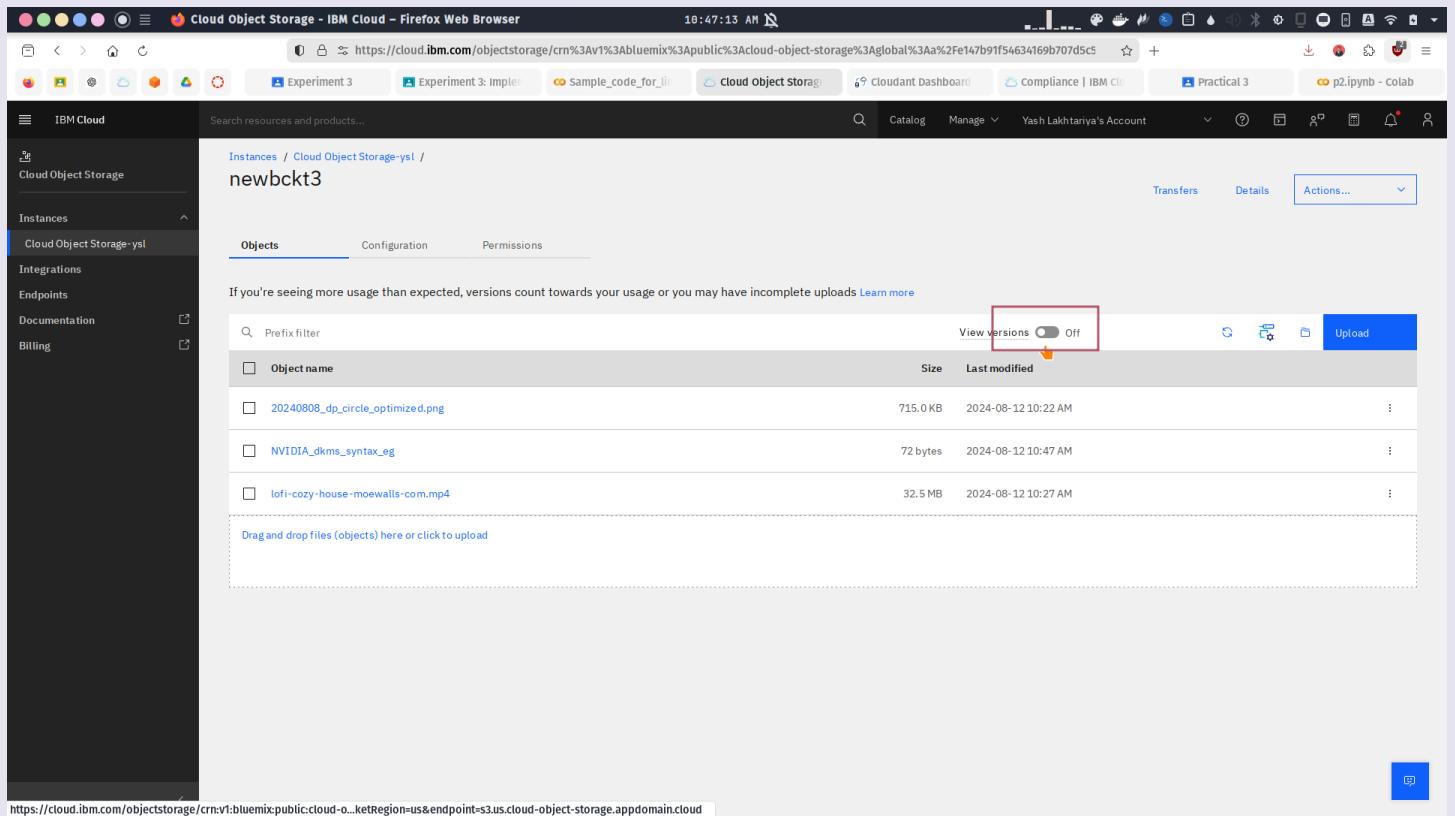
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

20. Try uploading changed file with same name and it should be allowed now

The screenshot shows the IBM Cloud Object Storage interface in a Firefox browser. The URL is <https://cloud.ibm.com/objectstorage/crn%3Av1%3Abuemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c5>. The page displays a list of objects in a bucket named 'newbckt3'. Three files are listed: '20240808_dp_circle_optimized.png' (715.0 KB, last modified 2024-08-12 10:22 AM), 'NVIDIA_dkms_syntax_eg' (63 bytes, last modified 2024-08-12 10:10 AM), and 'lofi-cozy-house-moewalls-com.mp4' (32.5 MB, last modified 2024-08-12 10:27 AM). To the right, a modal window titled 'Upload' is open, showing options for 'Standard transfer' (selected) and 'Aspera high-speed transfer'. It also shows a preview of the selected file 'NVIDIA_dkms_sy...' (72 bytes). The 'Upload' button at the bottom of the modal is highlighted with a yellow arrow.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

21. View Versions button can be used to show files with all versions including null (initial version)



The screenshot shows the IBM Cloud Object Storage interface. On the left, there's a sidebar with 'Instances' selected, showing 'Cloud Object Storage-ysl'. The main area shows a list of objects under 'newbckt3'. At the top right of the object list, there's a 'View versions' toggle switch, which is currently set to 'Off'. An orange arrow points to this toggle switch. Below the toggle, there are three columns: 'Object name', 'Size', and 'Last modified'. Three files are listed: '20240808_dp_circle_optimized.png' (715.0 KB, 2024-08-12 10:22 AM), 'NVIDIA_dkms_syntax_eg' (72 bytes, 2024-08-12 10:47 AM), and 'lofi-cozy-house-moewalls-com.mp4' (32.5 MB, 2024-08-12 10:27 AM). A 'Drag and drop files (objects) here or click to upload' area is visible at the bottom.

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

The screenshot shows the IBM Cloud Object Storage interface in a Firefox browser window. The URL is https://cloud.ibm.com/objectstorage/crn%3Av1%3Abluemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c. The page title is "Cloud Object Storage - IBM Cloud - Firefox Web Browser". The time is 10:47:46 AM.

The left sidebar shows the navigation menu: IBM Cloud, Instances, Cloud Object Storage-ysl, Integrations, Endpoints, Documentation, and Billing.

The main content area shows the "Instances / Cloud Object Storage-ysl / newbckt3" view. The "Objects" tab is selected, displaying a list of objects: Prefix filter, Object name (checkboxes for 20240808_dp_circle_optimized.png, NVIDIA_dkms_syntax_eg, and lofi-cozy-house-moewalls-com.mp4), and a placeholder for dragging files.

The "Versions" tab is selected, showing the details for the object "NVIDIA_dkms_syntax_eg". The table has columns: Version ID, Archive Type, Size, and Last modified.

Version ID	Archive Type	Size	Last modified
00000191-4504-b6a8-db00-e79552f6e0fe	Current version	72 bytes	2024-08-12 10:47 AM
null		63 bytes	2024-08-12 10:10 AM

Below the table, there are buttons for "Search" and "Lifecycle". At the bottom, there are pagination controls: "Items per page: 25", "1-2 of 2 items", and "1 of 1 page".

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

22. Try deleting the versions and changing default one

The screenshot shows the IBM Cloud Object Storage interface. On the left, the sidebar is visible with options like Instances, Integrations, Endpoints, Documentation, and Billing. The main area shows a list of objects under 'Cloud Object Storage-ysl / newbckt3'. The 'Objects' tab is selected. In the center, there's a table titled 'NVIDIA_dkms_syntax_eg' with three rows:

Version ID	Archive Type	Size	Last modified
00000191-4507-814e-3ea8-fe11cde2375a	Current version	72 bytes	2024-08-12 10:50 AM
null		63 bytes	2024-08-12 10:10 AM

At the bottom right of the table, there are buttons for 'Items per page' (set to 25), '1-2 of 2 items', and a dropdown for selecting a specific item. A context menu is open over the second row (the null version), with the 'Delete' option highlighted in red.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

23. Also try deleting the whole object, then it will not be permanently deleted, but delete marker will be set as current version

The screenshot shows the IBM Cloud Object Storage interface in a Firefox browser. The left sidebar includes links for Instances, Integrations, Endpoints, Documentation, and Billing. The main area displays a table of objects with columns for Object name, Size, and Last modified. A red box highlights a specific row for the file '00000191-4a74-fd8a-e03f-1d2a3b9495f3'. This row contains three buttons: 'Current version' (highlighted in blue), 'Delete marker', and a delete icon. The table also lists other files like '20240808_dp_circle_optimized.png' and 'lofi-cozy-house-moewalls-com.mp4'.

Object name	Size	Last modified
20240808_dp_circle_optimized.png	715.0 KB	2024-08-12 10:22 AM
null (Current version)		
NVIDIA_dkms_syntax_eg		
00000191-4a74-fd8a-e03f-1d2a3b9495f3 (Current version)	0 bytes	2024-08-13 12:07 PM
00000191-4a74-fd8a-e03f-1d2a3b9495f3 Delete marker		
00000191-4a74-fd8a-e03f-1d2a3b9495f3	74 bytes	2024-08-13 12:03 PM
00000191-450c-450a-218f-cc0-e734fd1c427d	74 bytes	2024-08-12 10:55 AM
00000191-450c-450a-218f-cc0-e734fd1c427d	74 bytes	2024-08-12 10:52 AM
00000191-450c-450a-218f-cc0-e734fd1c427d	72 bytes	2024-08-12 10:50 AM
lofi-cozy-house-moewalls-com.mp4		
null (Current version)	32.5 MB	2024-08-12 10:27 AM

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

The screenshot shows the IBM Cloud Object Storage interface. On the left, there's a sidebar with 'IBM Cloud' and 'Cloud Object Storage' sections. The main area is titled 'Objects' and lists several files. One file, 'NVIDIA_dkms_syntax_eg', has a context menu open over it. The menu includes options like 'Object details', 'Restore', 'Manage tags', 'Download', and 'Delete'. The 'Object details' option is highlighted with a red box.

Object name	Size	Last modified
20240808_dp_circle_optimized.png	715.0 KB	2024-08-12 10:22 AM
NVIDIA_dkms_syntax_eg		
00000191-4a74-fd8a-e03f-1d2a3b9495f3	0 bytes	2024-08-13 12:07 PM
00000191-4a70-f460-6479-493b7ecebdfb	74 bytes	2024-08-13 12:03 PM
00000191-450c-c70c-0420-5ae3302c2725	74 bytes	2024-08-12 10:55 AM
00000191-450a-218f-cc00-e734fd1c427d	74 bytes	2024-08-12 10:52 AM
00000191-4507-814e-3ea8-fe11cde2375a	72 bytes	2024-08-12 10:50 AM
lofi-cozy-house-moewalls-com.mp4	32.5 MB	2024-08-12 10:27 AM

This screenshot shows the detailed view for the 'NVIDIA_dkms_syntax_eg' object. At the top, there are download and delete buttons. Below that, tabs for 'Overview', 'Versions' (which is selected), 'Lifecycle', and 'Retention' are visible. The 'Versions' tab displays a table of file versions. The first version is marked as 'Current version'.

Version ID	Archive Type	Size	Last modified
00000191-4a74-fd8a-e03f-1d2a3b9495f3	0 bytes	2024-08-13 12:07 PM	
00000191-4a70-f460-6479-493b7ecebdfb	74 bytes	2024-08-13 12:03 PM	
00000191-450c-c70c-0420-5ae3302c2725	74 bytes	2024-08-12 10:55 AM	
00000191-450a-218f-cc00-e734fd1c427d	74 bytes	2024-08-12 10:52 AM	
00000191-4507-814e-3ea8-fe11cde2375a	72 bytes	2024-08-12 10:50 AM	

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

24. Now, change default version to previously used ones to restore object

The screenshot shows the IBM Cloud Object Storage interface. On the left, there's a sidebar with 'IBM Cloud' and 'Cloud Object Storage' sections. The main area displays a list of objects under 'Objects'. One object, 'NVIDIA_dkms_syntax_eg', is selected. The 'Versions' tab is active, showing a table of versions. The second version, with Version ID '00000191-4a74-fd8a-e03f-1d2a3b9495f3', is highlighted with a red box and has a red arrow pointing to its 'Make current vers...' button.

Version ID	Archive Type	Size	Last modified
00000191-4a74-fd8a-e03f-1d2a3b9495f3	Current version	0 bytes	2024-08-13 12:07 PM
00000191-4a70-f460-6479-493b7cecbdfb		74 bytes	2024-08-13 12:03 PM
00000191-450c-c70c-0420-5ae3302c2725		74 bytes	2024-08-13 12:03 PM
00000191-450a-218f-cc00-e734fd1c427d		74 bytes	2024-08-13 12:03 PM
00000191-4507-814e-3ea8-fe11cde2375a		72 bytes	2024-08-13 12:03 PM

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

The screenshot shows the IBM Cloud Object Storage interface in a Firefox browser window. The URL is https://cloud.ibm.com/objectstorage/crn%3Av1%3Abluemix%3Apublic%3Acloud-object-storage%3Aglobal%3Aa%2Fe147b91f54634169b707d5c5d51b5. The page displays a list of object versions for 'NVIDIA_dkms_syntax_eg'. The 'Versions' tab is selected. The table shows the following data:

Version ID	Archive Type	Size	Last modified
00000191-4a79-6a76-9557-9e20d4c7b914	Current version	74 bytes	2024-08-13 12:12 PM
00000191-4a79-333e-a93a-82ea255ee0b4		74 bytes	2024-08-13 12:12 PM
00000191-4a74-fd8a-e03f-1d2a3b9495f3	Delete marker	0 bytes	2024-08-13 12:07 PM
00000191-4a70-f460-6479-493b7cebdffb		74 bytes	2024-08-13 12:03 PM
00000191-450c-c70c-0420-5ae3302c2725		74 bytes	2024-08-12 10:55 AM
00000191-450a-218f-cc00-e734fd1c427d		74 bytes	2024-08-12 10:52 AM
00000191-4507-814e-3ea8-fe11cde2375a		72 bytes	2024-08-12 10:50 AM

Below the table, there are buttons for 'Items per page' (set to 25) and '1-7 of 7 items'. A message at the bottom says 'Drag and drop files (objects) here or click to upload'.

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

25. Now, install node package @cloudant/couchbackup

The screenshot shows a Visual Studio Code interface. In the Explorer sidebar, a project named 'PRACTICAL_3' is open, showing files like 'node_modules', 'public', 'public2', 'public3', 'index.html', 'apikey_modedb', 'app.js', 'app2.js', 'app3.js', 'package-lock.json', and 'package.json'. The 'package.json' file is selected and shown in the main editor area:

```
You, 5 seconds ago | 1 author (You)
1 {           You, 5 days ago * updated
2   "dependencies": {
3     "@cloudant/couchbackup": "^2.9.9",
4     "express": "4.19.2"
5   }
6 } <- #1-6 { "dependencies": { "@cloudant/couchbackup": "^2.9.9", "expres...
```

Below the editor, a terminal window displays the command and its output:

```
yarn .../Practical_3  If main ? ) @ v22.6.0  11:13  npm install @cloudant/couchbackup
npm warn EBADENGINE Unsupported engine {
npm warn EBADENGINE   package: '@ibm-cloud/cloudant@0.5.1',
npm warn EBADENGINE   required: { node: '^14 || ^16 || ^18' },
npm warn EBADENGINE   current: { node: 'v22.6.0', npm: '10.8.2' }
npm warn EBADENGINE }
npm warn deprecated inflight@0.1.6: This module is not supported, and leaks memory. Do not use it. Check out lru-cache if you want a good and tested way to coalesce async requests by a key value, which is much more comprehensive and powerful.
npm warn deprecated rimraf@3.0.2: Rimraf versions prior to v4 are no longer supported
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer supported

added 142 packages, and audited 207 packages in 37s
I
27 packages are looking for funding
  run 'npm fund' for details

4 vulnerabilities (2 moderate, 2 high)

To address all issues, run:
  npm audit fix
  You, 5 days ago  Ln 1, Col 1  Spaces: 2  UTF-8  LF  ↳ JSON  ↳ Prettier
```

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA Batch - 71

CS Practical 3

26. Copy the main URL from service credentials of Cloudant instance and use couchbackup command (here local project node module bin.js command, couchbackup without extension command if globally installed) to backup the database to a plain file

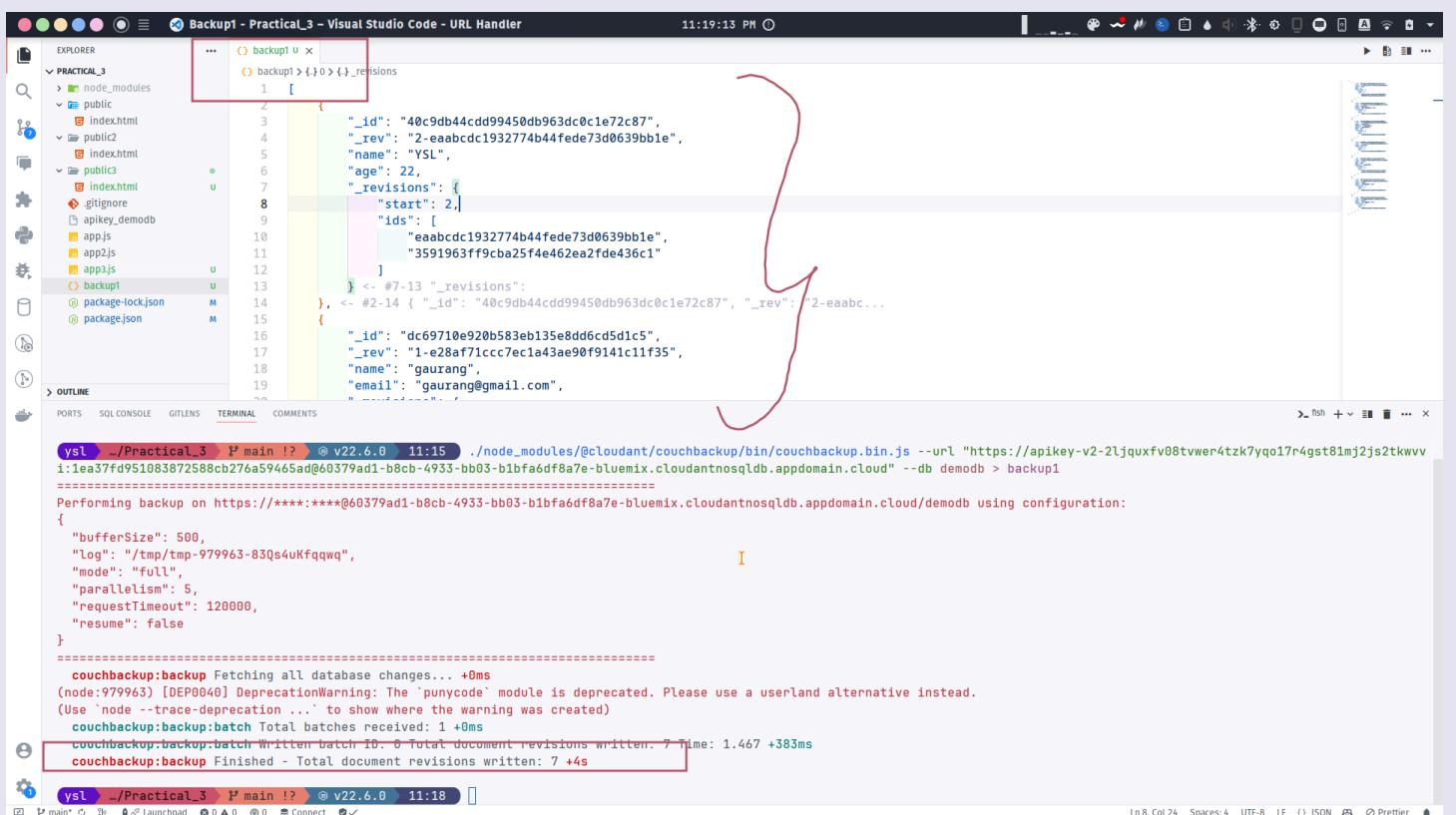
Commands :

a. For local installation inside project :

```
./path_to_project/node_modules/@cloudant/couchbackup/bin/couchbackup.bin.js --url "URL here" --db databasename > myfile
```

b. For global installation of couchbackup node module :

```
couchbackup --url "URL here" --db databasename > myfile
```



The screenshot shows a Visual Studio Code interface with the title bar "Backup1 - Practical_3 - Visual Studio Code - URL Handler" and the status bar "11:19:13 PM". The Explorer sidebar on the left shows a project structure with files like "node_modules", "public", "index.html", "index2.html", "public2", "index2.html", ".gitignore", "apikey_demo.js", "app.js", "app2.js", "app3.js", "backup1", "package-lock.json", and "package.json". The code editor displays a JSON document with several objects, some of which have their "name" field set to "YSL". The terminal at the bottom shows the command being run:

```
ysl ./Practical_3 !? @ v22.6.0 11:15 ./node_modules/@cloudant/couchbackup/bin/couchbackup.bin.js --url "https://apikey-v2-2ljquxfv08tvwer4tzk7yqo17r4gst81mj2js2tkwvv1:ea37fd951083872588cb276a59465ad60379ad1-b8cb-4933-bb03-b1bfafdf8a7e-bluemix.cloudantnosqlb.appdomain.cloud" --db demojs > backup1
=====
Performing backup on https://*****@60379ad1-b8cb-4933-bb03-b1bfafdf8a7e-bluemix.cloudantnosqlb.appdomain.cloud/demodbs using configuration:
{
  "bufferSize": 500,
  "log": "/tmp/tmp-979963-83qs4uKfqwq",
  "mode": "full",
  "parallelism": 5,
  "requestTimeout": 120000,
  "resume": false
}
=====
couchbackup:backup Fetching all database changes... +0ms
(node:979963) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
couchbackup:batch Total batches received: 1 +0ms
couchbackup:batch Written batch 10. 0 total document revisions written. Time: 1.467 +383ms
couchbackup:backup Finished - Total document revisions written: 7 +4s
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 71
CS Practical 3

27. Now, create new empty database and load data from backup file to it

The screenshot shows the Cloudant Dashboard interface in a Firefox browser. The URL is https://60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudant.com/dashboard.html#database/demodb3/_all_docs. The title bar says "Cloudant Dashboard.../Demodb3/_all_docs - Firefox Web Browser". The dashboard shows a sidebar with options like All Documents, Query, Permissions, Changes, and Design Documents. The main area is titled "demodb3" and contains a message "Database created successfully". Below this, there is a large cloud icon and the text "No Documents Found". At the bottom right, it says "Showing 0 documents. Documents per page: 20". A red box highlights the database name "demodb3" in the sidebar and the success message in the main area.

Commands :

a. For local installation :

```
./path_to_project/node_modules/@cloudant/couchbackup/bin/couchbackup.bin.js --url "URL here" --db databasename > myfile
```

b. For global one :

```
couchrestore --url "URL here" --db databasename < myfile
```

Name - Yash Lakhtariya

Enrollment number - 21162101012

Branch - CBA **Batch - 71**

CS Practical 3

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows a project structure for "PRACTICAL_3". The "public" folder contains "index.html", "index2.html", and "index3.html". The "public3" folder contains "index.html", ".gitignore", "apikey_demo.db", "app.js", "app2.js", "app3.js", and "backup1".
- Terminal:** The terminal window displays the command-line output of a "couchrestore" operation:

```
ysl .../Practical_3 $ main !? @ v22.6.0 11:21 ./node_modules/@cloudant/couchbackup/bin/couchrestore.bin.js --url "https://apikey-v2-2ljquxfv08tvwer4tzk7yqo17r4gst81mj2js2tkwv1:1ea37fd51083872588cb276a59465ad@60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosql_db.appdomain.cloud" --db demodb3 < backup1
=====
Performing restore on https://****:****@60379ad1-b8cb-4933-bb03-b1bfa6df8a7e-bluemix.cloudantnosql_db.appdomain.cloud/demodb3 using configuration:
{
  "bufferSize": 500,
  "parallelism": 5,
  "requestTimeout": 120000
}
=====
(node:98788) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
couchbackup:restore:batch restored 7 +0ms
couchbackup:restore finished { total: 7 } +0ms
```
- Status Bar:** Shows the current file is "main.js", the version is "v22.6.0", the date is "11:21", and the status bar includes icons for file operations like save, close, and connect.

	Id	key	value
<input type="checkbox"/>	181051388d34115c927f8a5d7ad990de	181051388d34115c927f8a5d7ad990de	{"rev": "1-4fd05699026dd45e027394f9bb340d0f"}
<input type="checkbox"/>	40c9db44cdd99450db963dc0c1e72c87	40c9db44cdd99450db963dc0c1e72c87	{"rev": "2-eaabcdc1932774b44febe73d0639bb1e"}
<input type="checkbox"/>	58eeb0735bb067cc97abae0f0913f4575	58eeb0735bb067cc97abae0f0913f4575	{"rev": "2-f32321fd4dd1a0b5f0efbdec3a72dd95"}
<input type="checkbox"/>	8b0990cc743a896b9a9da986a2348738	8b0990cc743a896b9a9da986a2348738	{"rev": "1-4ac2651119cd30d5c6ea46d059331441"}
<input type="checkbox"/>	b00df27eb68754e691b0163552560ee0	b00df27eb68754e691b0163552560ee0	{"rev": "1-e0dd9c018dfdfcc8f05d4bc32087482"}
<input type="checkbox"/>	b6c0313ad4c2d55c51b5f492e51c8c53	b6c0313ad4c2d55c51b5f492e51c8c53	{"rev": "1-de0b572a9e4f7eab50f79a244807a753"}
<input type="checkbox"/>	dc69710e920b583eb135e8dd6cd5d1c5	dc69710e920b583eb135e8dd6cd5d1c5	{"rev": "1-e2af71ccc7ec1a43ae0f9141c11f35"}