IBM GEN AI REPORT

Phase 2: Implementation & Execution: AI-Powered Document Digitization and Analysis using IBM Watsonx

VIT Bhopal University

Name : Asmita Sarkar
Registration Number : 22BSA10100
Email : asmitasarkar2022@vitbhopal.ac.in

AI-Powered Document Digitization and Analysis using IBM Watsonx

---

Objective:

To automate the extraction of textual and tabular content from highly structured documents (PDFs) using IBM Watsonx.ai Text Extraction API, making them searchable and analyzable for downstream applications such as analytics, compliance, or archiving.

Tools & Technologies:

- **IBM Watsonx.ai**
- **IBM Cloud Object Storage**
- **IBM Boto3 (Python SDK)**
- **Python**
- **Jupyter Notebook / IBM Watson Studio**
- **JSON & Markdown for output**

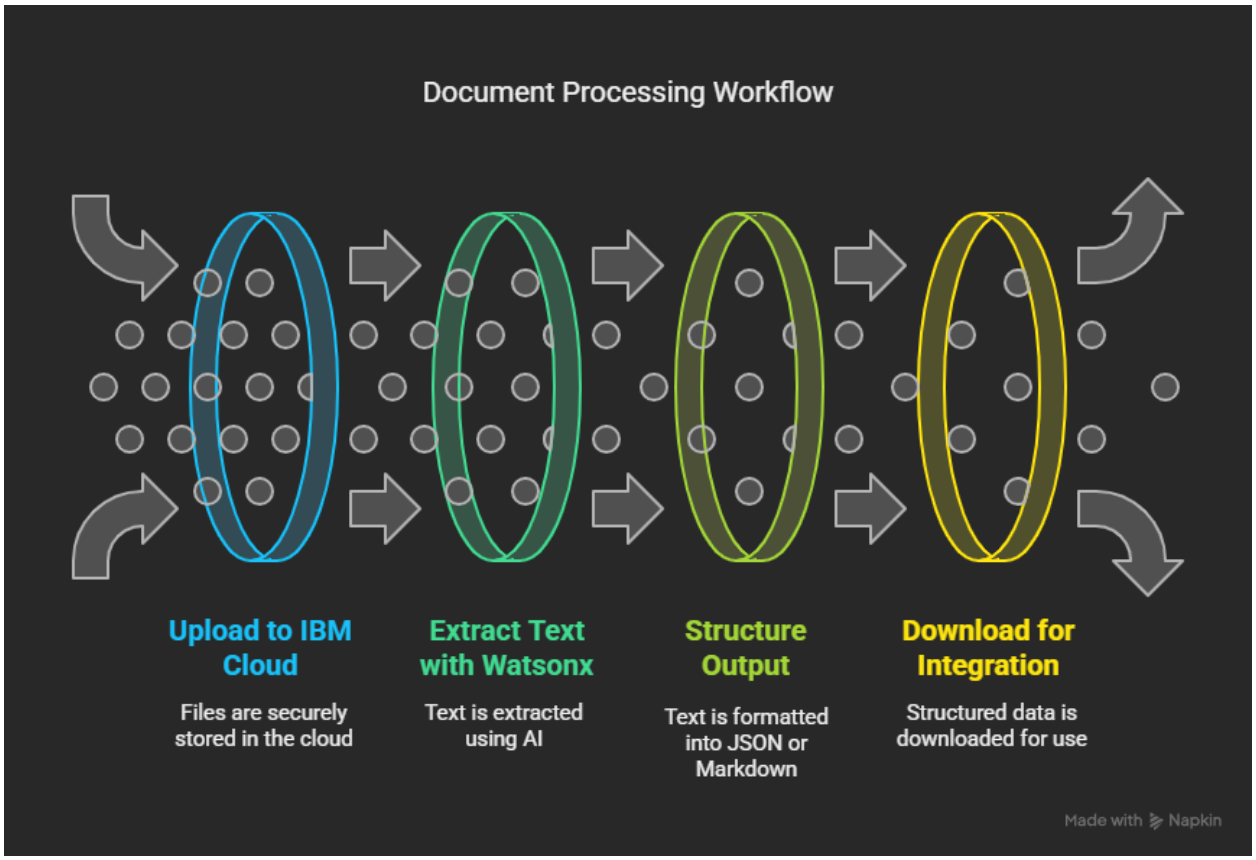
Implementation Steps:

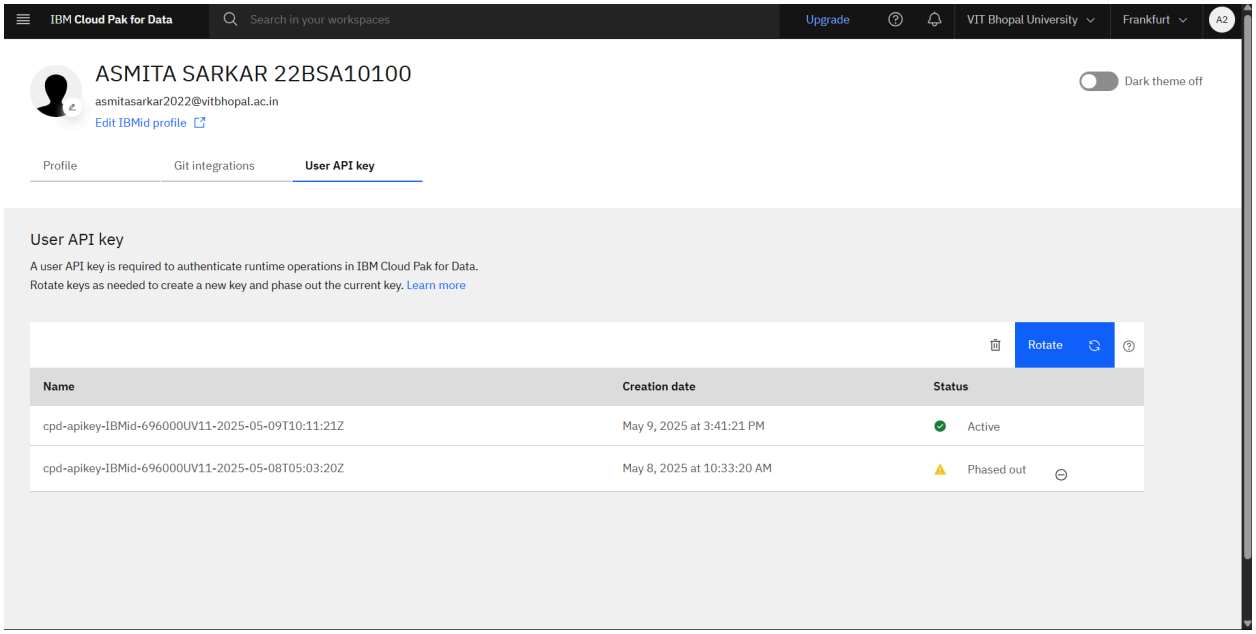
Step 1: Initial Setup

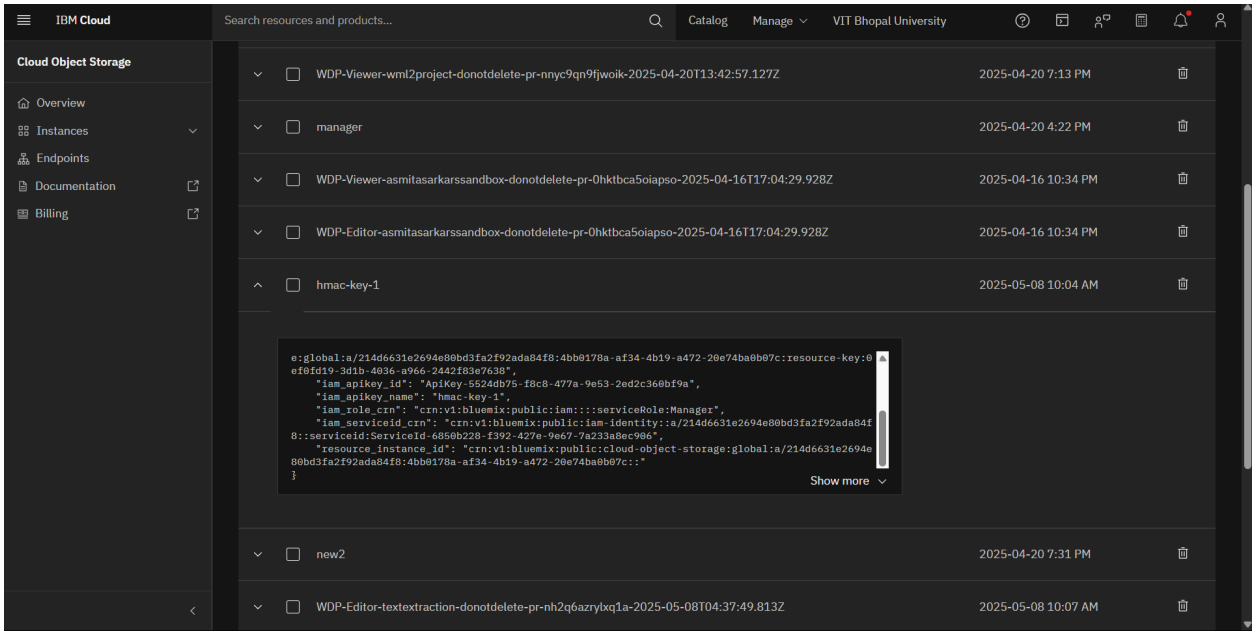
- **IBM Cloud Account Setup:** A cloud account was created with access to:
    - IBM Cloud Object Storage (COS)

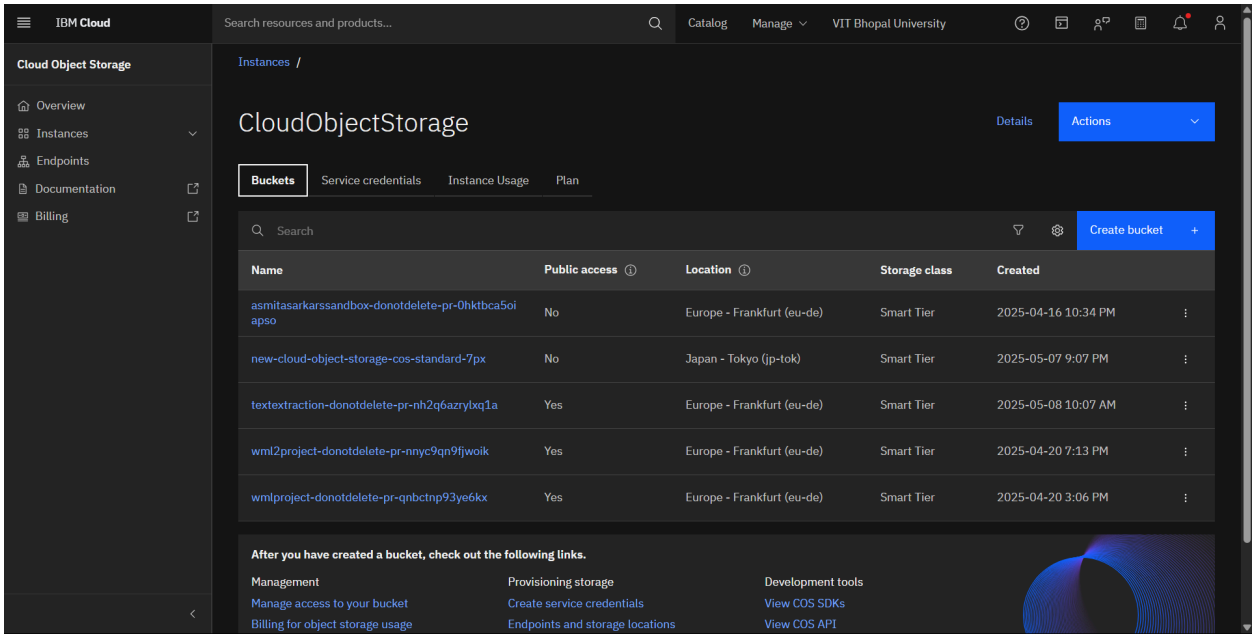
    - Watsonx.ai runtime

- **API Key & HMAC Credentials:** Task credentials were generated for secure and authenticated access.




IBM Cloud Pak for Data
Search in your workspaces
Upgrade
VIT Bhopal University
Frankfurt
A2
ASMITA SARKAR 22BSA10100
asmitasarkar2022@vitbhopal.ac.in
Edit IBMid profile
Dark theme off
Profile
Git integrations
User API key
User API key
A user API key is required to authenticate runtime operations in IBM Cloud Pak for Data.
Rotate keys as needed to create a new key and phase out the current key. Learn more
Rotate
Name
Creation date
Status
cpd-apikey-IBMid-696000UV11-2025-05-09T10:11:21Z
May 9, 2025 at 3:41:21 PM
Active
cpd-apikey-IBMid-696000UV11-2025-05-08T05:03:20Z
May 8, 2025 at 10:33:20 AM
Phased out

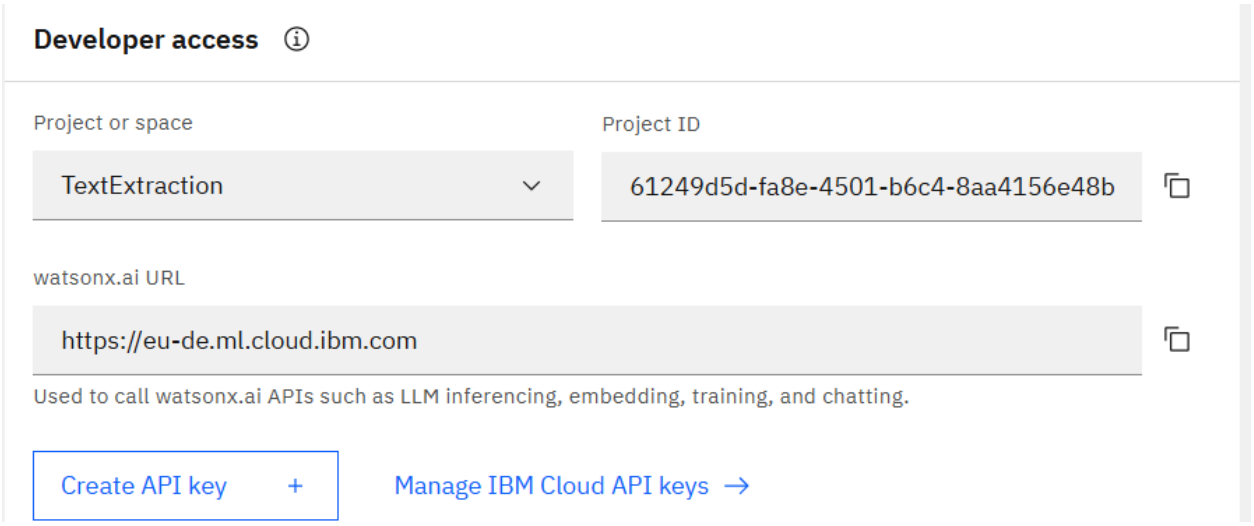

IBM Cloud
Search resources and products...
Catalog
Manage
VIT Bhopal University
Cloud Object Storage
Overview
Instances
Endpoints
Documentation
Billing
WDP-Viewer-wml2project-donotdelete-pr-nnyc9qn9fjwoik-2025-04-20T13:42:57.127Z
2025-04-20 7:13 PM
manager
2025-04-20 4:22 PM
WDP-Viewer-asmitasarkarssandbox-donotdelete-pr-0hktbca5oiapso-2025-04-16T17:04:29.928Z
2025-04-16 10:34 PM
WDP-Editor-asmitasarkarssandbox-donotdelete-pr-0hktbca5oiapso-2025-04-16T17:04:29.928Z
2025-04-16 10:34 PM
hmac-key-1
2025-05-08 10:04 AM
e:global:a/214d6631e2694e80bd3fa2f92ada84f8:4bb0178a-af34-4b19-a472-20e74ba0b07c:resource-key:0ef0fd19-3d1b-4036-a966-2442f83e7638",
"iam_apikey_id": "ApiKey-5524db75-f8c8-477a-9e53-2ed2c360bf9a",
"iam_apikey_name": "hmac-key-1",
"iam_role_crn": "crn:v1:bluemix:public:iam::::serviceRole:Manager",
"iam_serviceid_crn": "crn:v1:bluemix:public:iam-identity::a/214d6631e2694e80bd3fa2f92ada84f8::serviceid:ServiceId-6850b228-f392-427e-9e67-7a233a8ec906",
"resource_instance_id": "crn:v1:bluemix:public:cloud-object-storage:global:a/214d6631e2694e80bd3fa2f92ada84f8:4bb0178a-af34-4b19-a472-20e74ba0b07c::"
}
Show more
new2
2025-04-20 7:31 PM
WDP-Editor-textextraction-donotdelete-pr-nh2q6azrylxq1a-2025-05-08T04:37:49.813Z
2025-05-08 10:07 AM

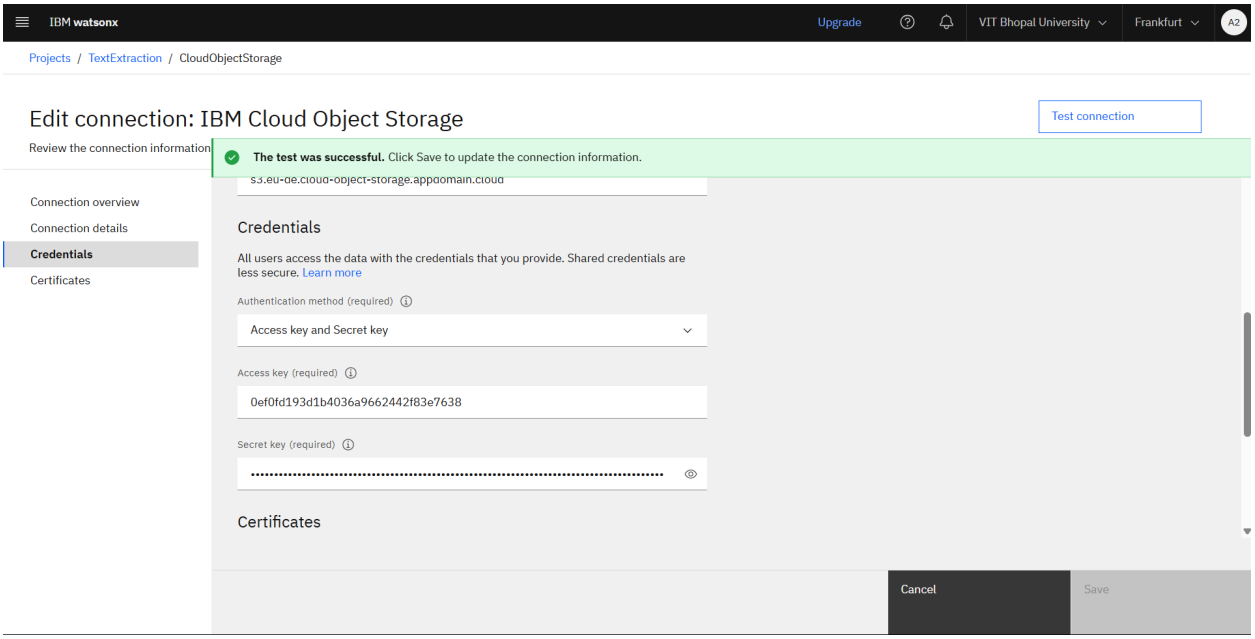


Step 2: IBM Cloud Object Storage Configuration

- **Bucket Creation:**
  - Created a custom bucket named `doc-extract-bucket-phase2`.
  - Uploaded multiple structured PDF documents.

- **Files uploaded include:**
  - `invoice_example1.pdf`
  - `report_sample2.pdf`
  - `compliance_doc3.pdf`



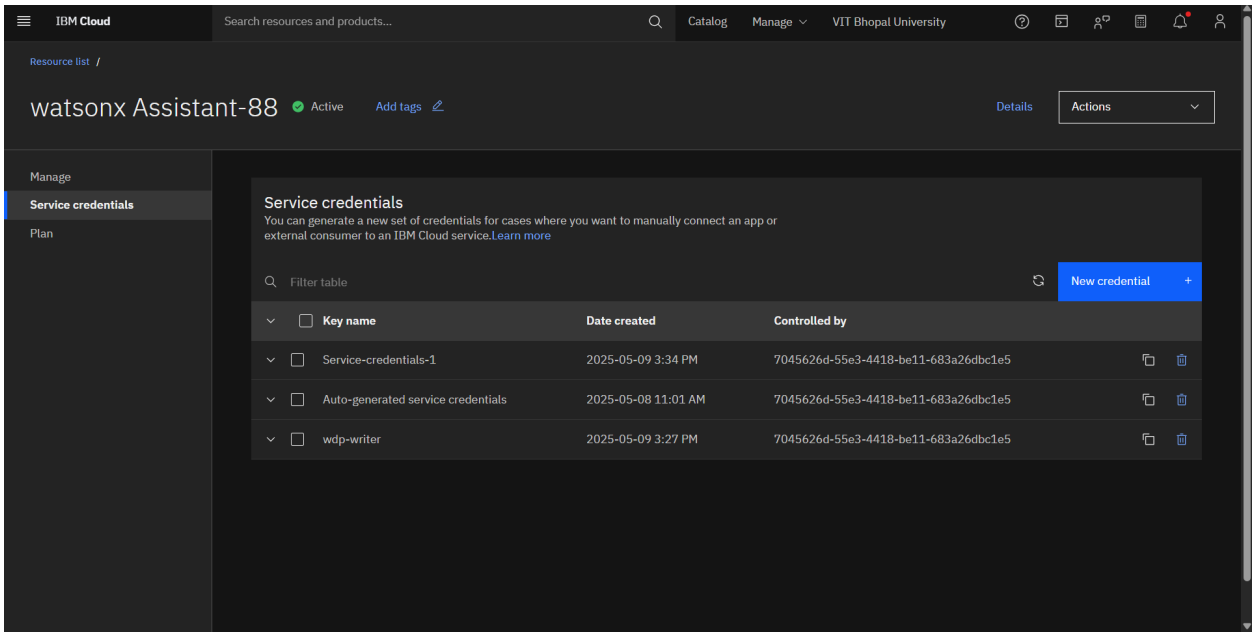


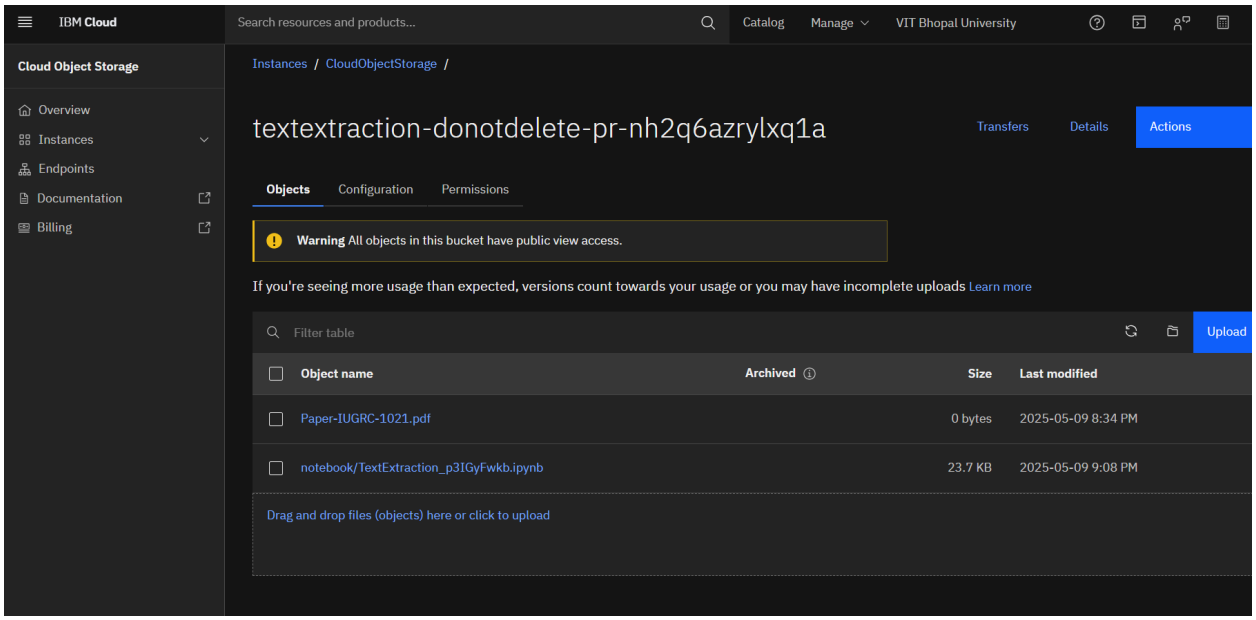
Step 3: Connecting Watsonx.ai to Cloud Storage

- Connection established between **Watsonx.ai project** and the **Cloud Object Storage bucket** using:
  - Bucket name
  - Public login endpoint
  - Access key and secret key (with HMAC enabled)

- **Validation:** Connection was successfully tested and verified.

- **Connection Asset ID:** Stored for use in Python script integration.





Step 4: Text Extraction via Python (Core Execution)

Python Packages Used:

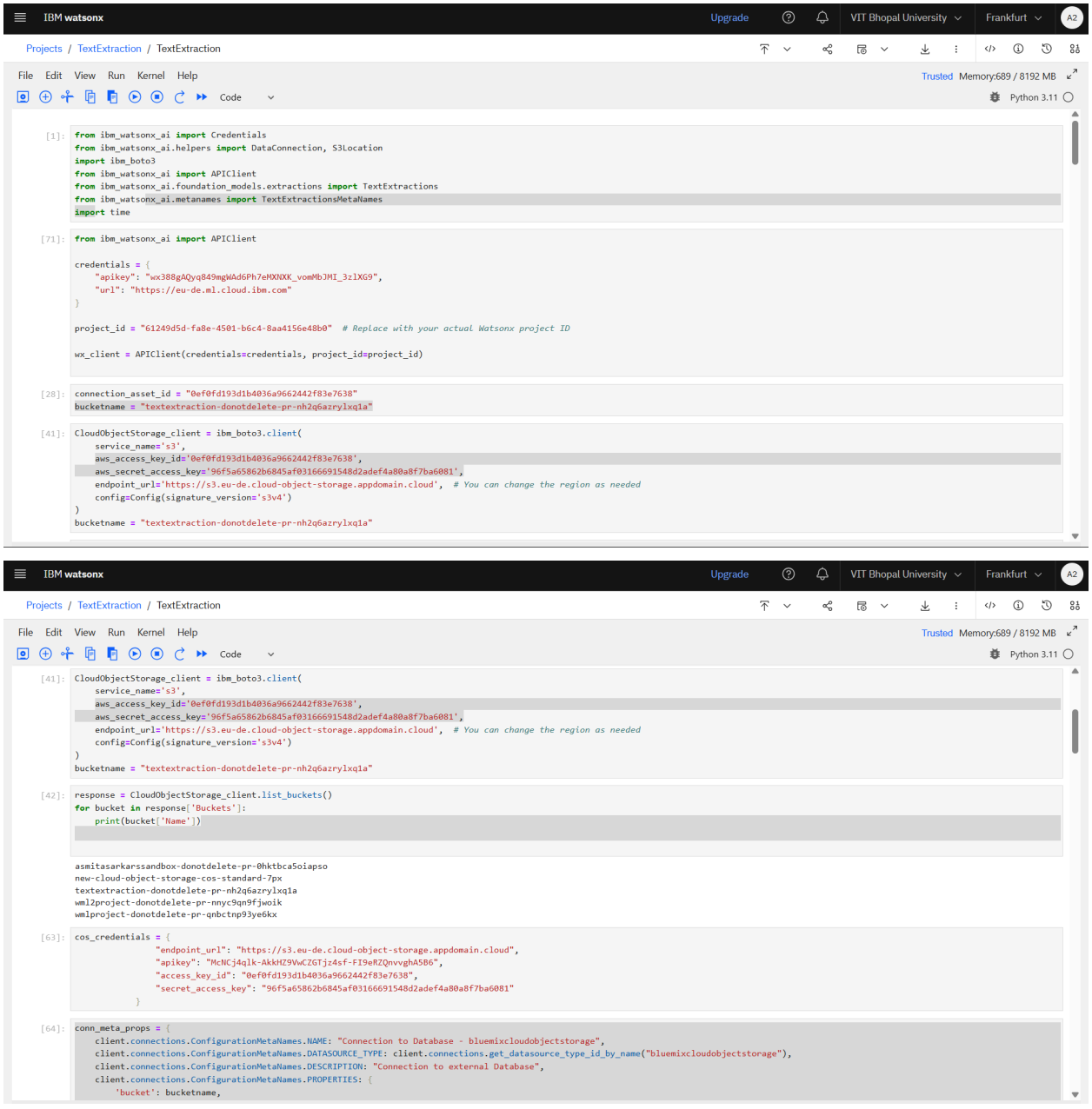
```
from ibm_watsonx_ai import Credentials, APIClient
```

```
from ibm_watsonx_ai.helpers import DataConnection, S3Location
```

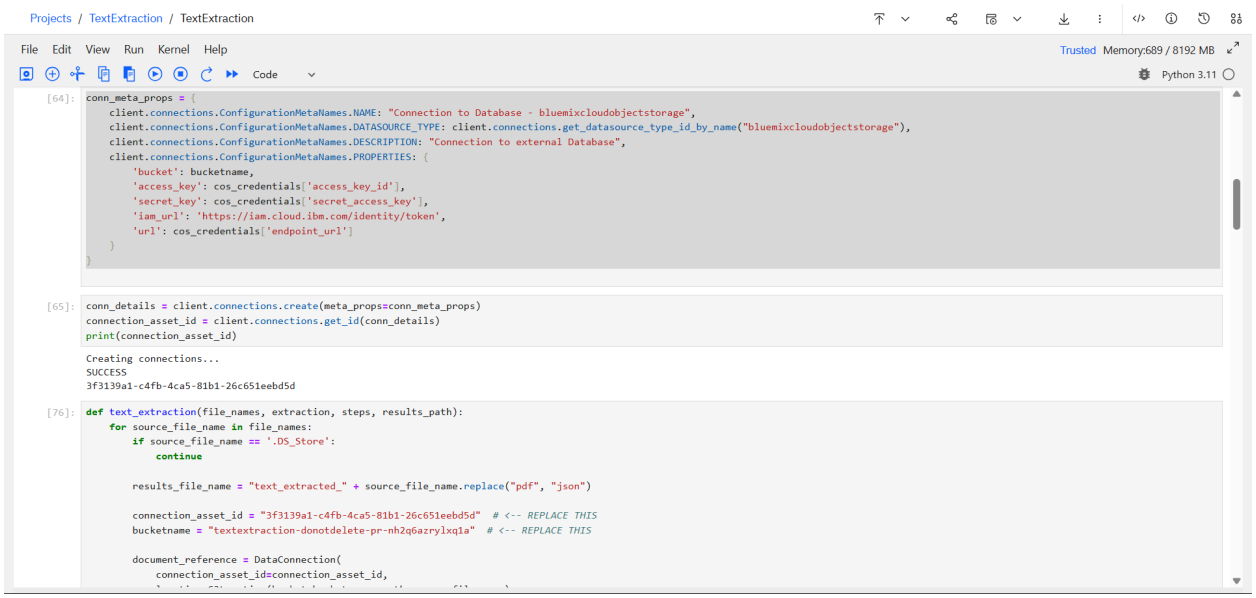
```
from ibm_watsonx_ai.foundation_models.extractions import
TextExtractions

from ibm_watsonx_ai.metanames import TextExtractionsMetaNames

import ibm_boto3, time
```

Initialization:

```
credentials = Credentials(url="https://us-south.ml.cloud.ibm.com",
api_key="<API_KEY>")

wx_client = APIClient(credentials=credentials,
project_id="<PROJECT_ID>")



bucketname = "doc-extract-bucket-phase2"

connection_asset_id = "<CONNECTION_ASSET_ID>"



CloudObjectStorage_client = ibm_boto3.client(

    service_name='s3',

    aws_access_key_id='<ACCESS_KEY>',

    aws_secret_access_key='<SECRET_KEY>',


endpoint_url='https://s3.us-south.cloud-object-storage.appdomain.cloud
'

)
```


IBM watsonx
Upgrade
VIT Bhopal University
Frankfurt
Projects / TextExtraction / TextExtraction
File Edit View Run Kernel Help
Trusted Memory:689 / 8192 MB
Code
Python 3.11
[1]:
from ibm_watsonx_ai import Credentials
from ibm_watsonx_ai.helpers import DataConnection, S3Location
import ibm_boto3
from ibm_watsonx_ai import APIClient
from ibm_watsonx_ai.foundation_models.extractions import TextExtractions
from ibm_watsonx_ai.metanames import TextExtractionsMetaNames
import time
[71]:
from ibm_watsonx_ai import APIClient
credentials = {
"apikey": "wx388gAQyq849mgWAd6Ph7eMXNXK_vomMbJMI_3zlXG9",
"url": "https://eu-de.ml.cloud.ibm.com"
}
project_id = "61249d5d-fa8e-4501-b6c4-8aa4156e48b0" # Replace with your actual Watsonx project ID
wx_client = APIClient(credentials=credentials, project_id=project_id)
[28]:
connection_asset_id = "0ef0fd193d1b4036a9662442f83e7638"
bucketname = "textextraction-donotdelete-pr-nh2q6azrylxq1a"
[41]:
CloudObjectStorage_client = ibm_boto3.client(
service_name='s3',
aws_access_key_id='0ef0fd193d1b4036a9662442f83e7638',
aws_secret_access_key='96f5a65862b6845af03166691548d2adef4a80a8f7ba6081',
endpoint_url='https://s3.eu-de.cloud-object-storage.appdomain.cloud', # You can change the region as needed
config=Config(signature_version='s3v4')
)
bucketname = "textextraction-donotdelete-pr-nh2q6azrylxq1a"

IBM watsonx
Upgrade
VIT Bhopal University
Frankfurt
Projects / TextExtraction / TextExtraction
File Edit View Run Kernel Help
Trusted Memory:689 / 8192 MB
Code
Python 3.11
[41]:
CloudObjectStorage_client = ibm_boto3.client(
service_name='s3',
aws_access_key_id='0ef0fd193d1b4036a9662442f83e7638',
aws_secret_access_key='96f5a65862b6845af03166691548d2adef4a80a8f7ba6081',
endpoint_url='https://s3.eu-de.cloud-object-storage.appdomain.cloud', # You can change the region as needed
config=Config(signature_version='s3v4')
)
bucketname = "textextraction-donotdelete-pr-nh2q6azrylxq1a"
[42]:
response = CloudObjectStorage_client.list_buckets()
for bucket in response['Buckets']:
print(bucket['Name'])
asmitasarkarssandbox-donotdelete-pr-0hktbca5oiapso
new-cloud-object-storage-cos-standard-7px
textextraction-donotdelete-pr-nh2q6azrylxq1a
wml2project-donotdelete-pr-nnyc9qn9fjwoik
wmlproject-donotdelete-pr-qnbctnp93ye6kx
[63]:
cos_credentials = {
"endpoint_url": "https://s3.eu-de.cloud-object-storage.appdomain.cloud",
"apikey": "McNCj4qlk-AkkHZ9VwCZGTjz4sf-FI9eRZQnvvghA5B6",
"access_key_id": "0ef0fd193d1b4036a9662442f83e7638",
"secret_access_key": "96f5a65862b6845af03166691548d2adef4a80a8f7ba6081"
}
[64]:
conn_meta_props = {
client.connections.ConfigurationMetaNames.NAME: "Connection to Database - bluemixcloudobjectstorage",
client.connections.ConfigurationMetaNames.DATASOURCE_TYPE: client.connections.get_datasource_type_id_by_name("bluemixcloudobjectstorage"),
client.connections.ConfigurationMetaNames.DESCRIPTION: "Connection to external Database",
client.connections.ConfigurationMetaNames.PROPERTIES: {
'bucket': bucketname,



Running Extraction:

```
response =
CloudObjectStorage_client.list_objects_v2(Bucket=bucketname)


if "Contents" in response:

    for obj in response["Contents"]:

        file_key = obj["Key"]

        if file_key.endswith(".pdf"):

            results_key = file_key.replace(".pdf", ".json")


            document_reference =
DataConnection(connection_asset_id=connection_asset_id,


location=S3Location(bucket=bucketname, path=file_key))
```

```
            results_reference =
DataConnection(connection_asset_id=connection_asset_id,


location=S3Location(bucket=bucketname, path=results_key))

            extraction = TextExtractions(api_client=wx_client,
project_id="<PROJECT_ID>")

            steps = {

                TextExtractionsMetaNames.OCR: {'language_list':
['en']},

                TextExtractionsMetaNames.TABLE_PROCESSING: {'enabled':
True}

            }

            extraction.run_job(

                document_reference=document_reference,

                results_reference=results_reference,

                steps=steps

            )
```

Step 5: Verification and Results

- **Output format:** Extracted data was saved as `.json` in the same bucket.
- **Sample Output Files:**
    - `invoice_example1.json`
    - `report_sample2.json`
- **Verification Done Using:**

    - IBM Cloud Storage Explorer
    - Python to fetch and view file content locally

Example Output Snippet:


```
{
  "title": "Monthly Sales Report",
  "date": "2025-03-12",
  "tables": [
    {
      "header": ["Product", "Quantity", "Price"],
      "rows": [
        ["Widget A", "10", "$25"],
        ["Widget B", "5", "$45"]
      ]
    }
  ]
}
```

Key Outcomes Achieved:

- Successful end-to-end setup and automation for document ingestion and text extraction.
- Clear JSON format outputs obtained for structured analysis.
- Demonstrated integration between object storage and AI processing services using APIs.

- Validated output against known document content.

Challenges & Solutions:

Challenge	Solution
Connection issues with bucket	Used manual editing and verified HMAC credentials
Delay in OCR response	Implemented `time.sleep()` between job submissions
Handling unsupported PDFs	Filtered input files to avoid `.DS_Store` or corrupted documents

---

Future Scope:

- Enhance with **text summarization** and **named entity recognition (NER)**.
- Extend to support image-based documents using Vision APIs.
- Build a **frontend interface** for real-time uploads and downloads.