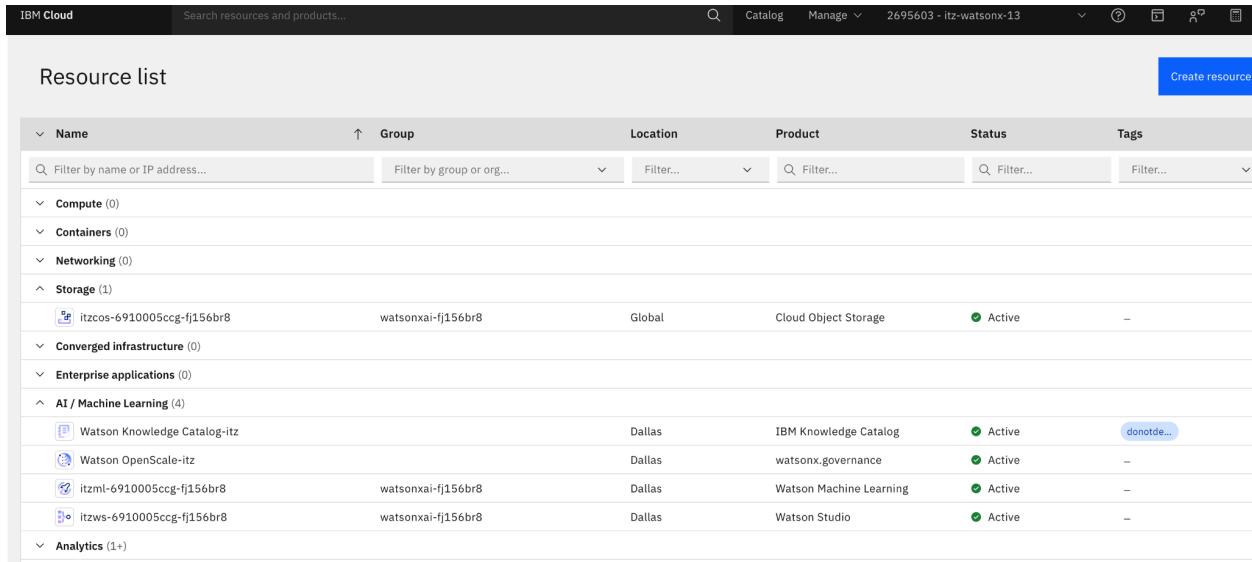


# LAB 1 Data Visualization and Creating Connections

Prereq. Ensure you have the sufficient resources required to do the lab (See Lab 0)



The screenshot shows the IBM Cloud Resource list interface. At the top, there is a navigation bar with 'IBM Cloud' on the left, a search bar 'Search resources and products...', and various icons and dropdown menus on the right. Below the navigation bar is a header titled 'Resource list' with a 'Create resource' button on the right. The main area is a table with columns: Name, Group, Location, Product, Status, and Tags. There are also filter buttons for each column. The table lists several resources under different groups:

Name	Group	Location	Product	Status	Tags
itcos-6910005ccg-fj156br8	watsonxai-fj156br8	Global	Cloud Object Storage	Active	-
Watson Knowledge Catalog-itz		Dallas	IBM Knowledge Catalog	Active	donotde...
Watson OpenScale-itz		Dallas	watsonx.governance	Active	-
itzml-6910005ccg-fj156br8	watsonxai-fj156br8	Dallas	Watson Machine Learning	Active	-
itzws-6910005ccg-fj156br8	watsonxai-fj156br8	Dallas	Watson Studio	Active	-

Step 1) Locate and open the project you created in Lab 0

The screenshot shows a 'Projects' interface with a single project entry. The project name is 'test-pro'. To the right of the project name is a timestamp '3 d ago'. A small user icon is also present next to the project name.

Step 2) Import a local CSV file to your sandbox project.

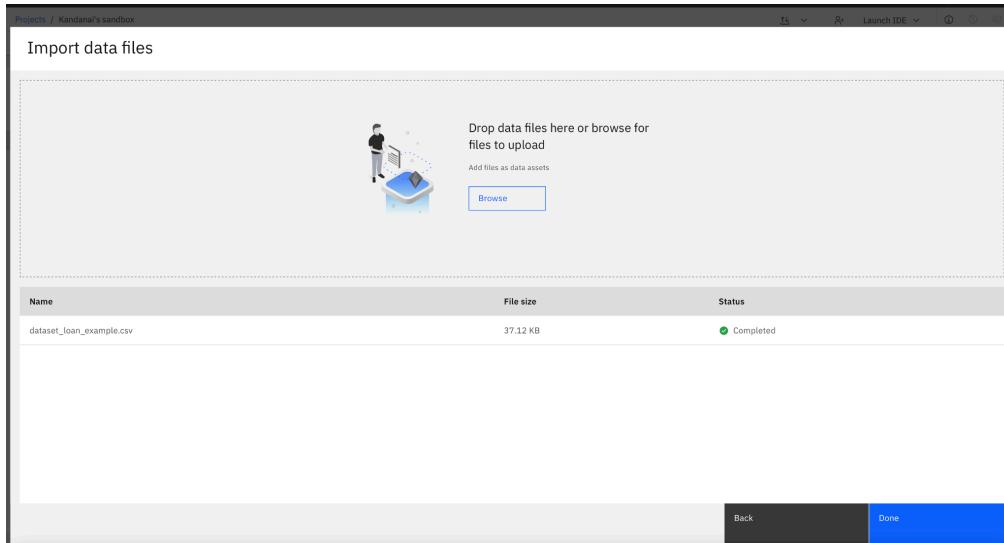
The screenshot shows the 'Assets' tab of a project. At the top right, there is a red box around the 'Import assets' button. Below the button, there is a 'New asset' button and a '+' sign. The main area shows a summary of assets ('0 asset') and a list of 'All assets'.

2.1) Go to the Assets tab then click “Import assets”.

2.2) Local file >> Data asset

The screenshot shows the 'Import assets' dialog box. On the left, there is a sidebar with options: 'Connected data', 'Local file' (which is selected and highlighted in blue), 'Project files', and 'Custom foundation model'. On the right, there is a main panel titled 'Import asset from local file'. It shows two categories: 'Data asset' and 'Model'. At the bottom of the dialog box are 'Cancel' and 'Import' buttons.

2.3) Browse/Drag and drop the provided csv file inside this Lab folder, “dataset\_loan\_example\_v2.csv”, then click on Done. Once Done, this will import the csv file to your project.



Step 3) Click on the 3-dots icon on the right of the dataset, then select “Prepare data” to preview the asset

The screenshot shows the 'Assets' tab in the IBM WatsonX interface. On the left, there's a sidebar with '3 assets' and 'All assets' selected. The main area shows a table of assets with columns for 'Name', 'Last modified', and 'Actions'. One asset, 'dataset\_loan\_example.csv', is selected. A context menu is open over this asset, with 'Prepare data' highlighted.

Projects / test-project / dataset\_loan\_example\_v2.csv / Data Refinery

Steps (1) X

Data source: dataset\_loan\_example\_v2.csv

1. Convert column type X

Automatically converted one or more columns to inferred data types. Strings that are converted to decimal use a dot (.) for the decimal symbol.

Auto-generated

Use a code template to add a step

Data Profile Visualizations

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CosapplicantIncome	LoanAmnt
1	Male	No	0	Graduate	No	5849	0	...
2	Male	Yes	1	Graduate	No	4583	1508	128
3	Male	Yes	0	Graduate	Yes	3000	0	66
4	Male	Yes	0	Not Graduate	No	2583	2358	120
5	Male	No	0	Graduate	No	6000	0	141
6	Male	Yes	2	Graduate	Yes	5417	4196	267
7	Male	Yes	0	Not Graduate	No	2333	1516	95
8	Male	Yes	3	Graduate	No	3036	2504	158
9	Male	Yes	2	Graduate	No	4006	1526	168
10	Male	Yes	1	Graduate	No	12841	10968	349
11	Male	Yes	2	Graduate	No	3200	700	70
12	Male	Yes	2	Graduate	---	2500	1840	109
13	Male	Yes	2	Graduate	No	3073	8106	200
14	Male	No	0	Graduate	No	1853	2840	114
15	Male	Yes	2	Graduate	No	1299	1086	17
16	Male	No	0	Graduate	No	4950	0	125
17	Male	No	1	Not Graduate	No	3596	0	100
18	Female	No	0	Graduate	No	3510	0	76
19	Male	Yes	0	Not Graduate	No	4887	0	133

New step +

Configure Viewing: 614 rows, 12 columns

About this asset X

Name: prep\_visualize\_data

Description: What is the purpose of this Data Refinery flow?

Asset details: Steps: 1

Associated assets: Source: dataset\_loan\_example\_v2.csv Target: dataset\_loan\_example\_v2\_c...

Last modified: Not yet saved

Created on: Not yet saved

You will be able to:

- 1) View the dataset (Your Raw data)
- 2) Data Profile to look at the distribution of the data ie. is the problem imbalance
- 3) Do some visualization ie. plotting graphs to perform visualizations

## After visualizing the data let's try importing the data into watson studio (Jupyter Notebook)

### Step 4) Import the data into watson studio

#### 4.1) New asset >> Work with data and models in Python or R notebooks

The screenshot shows the 'Assets' tab in the IBM Watson Studio interface. On the right side, there is a 'New asset' button with a red box drawn around it. The main area displays a list of assets under 'All assets', including a 'testing' notebook, a 'dataset\_lynn\_exercise.csv' CSV file, and a 'R2006-913005c9-d154a9' connection.

The screenshot shows a modal dialog titled 'What do you want to do?'. It asks the user to select a task based on their goal. In the 'Recent' section, there is a card for 'Work with data and models in Python or R notebooks' with a 'Learn more' link. Below this, in the 'Prepare data' section, there are four cards: 'Connect to a data source', 'Ground gen AI with vectorized documents', 'Prepare and visualize data', and 'Define reusable sets of parameters'. Each card has a corresponding icon and a 'with Connection' note.

## 4.2) Enter any name to create the notebook and select the Runtime to be “Runtime 24.1 on Python 3.11 XS (2 vCPU 8 GB RAM)

Work with data and models in Python or R notebooks

Define the details to create a notebook asset and open it in the Jupyter notebook editor tool.

+ New

Sample

Local file

URL

Define details

Name: testing

Description (optional): What's the purpose of this notebook

Define configuration

Select runtime: Runtime 24.1 on Python 3.11 XS (2 vCPU 8 GB RAM)

The selected runtime has 2 vCPU and 8 GB RAM.  
It consumes 1 capacity unit per hour.  
[Learn more about capacity unit hours and Watson Studio pricing plans.](#)

Language: Python 3.11

Cancel Create

4.3) Here you will see the jupyter notebook being provisioned. Click on the </> icon on the top right of the workbook, then select “Read data”.

IBM WatsonX

Projects / Kandanai's sandbox / testing

Not Trusted Memory:148 / 8192 MB Python 3.11

Code Snippets

Code Snippets x

Make your data science workflow more simple and productive with Code Snippets.

[...]

Data Ingestion

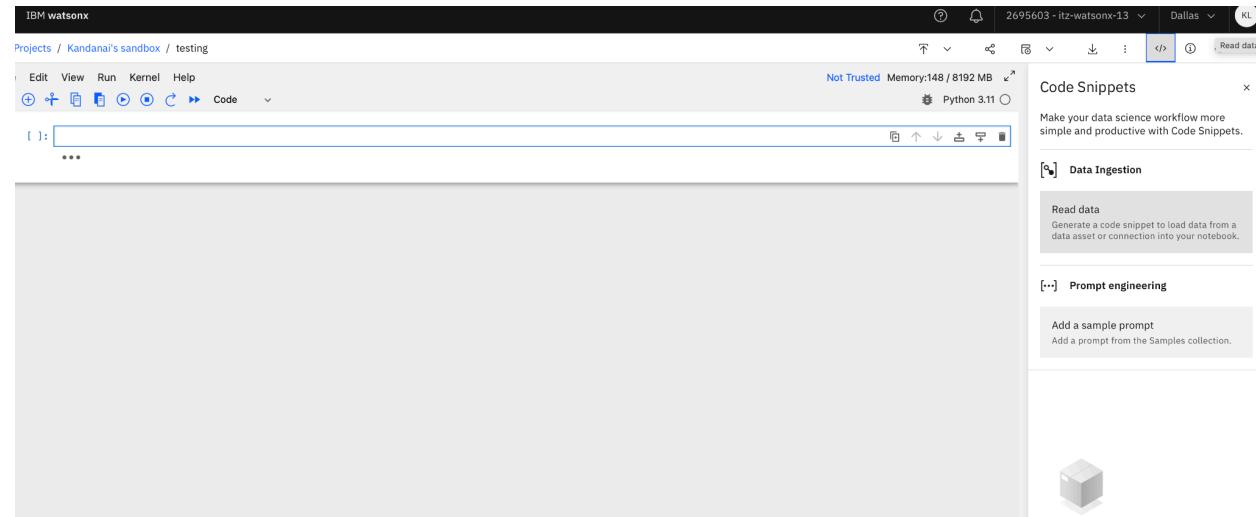
Read data

Generate a code snippet to load data from a data asset or connection into your notebook.

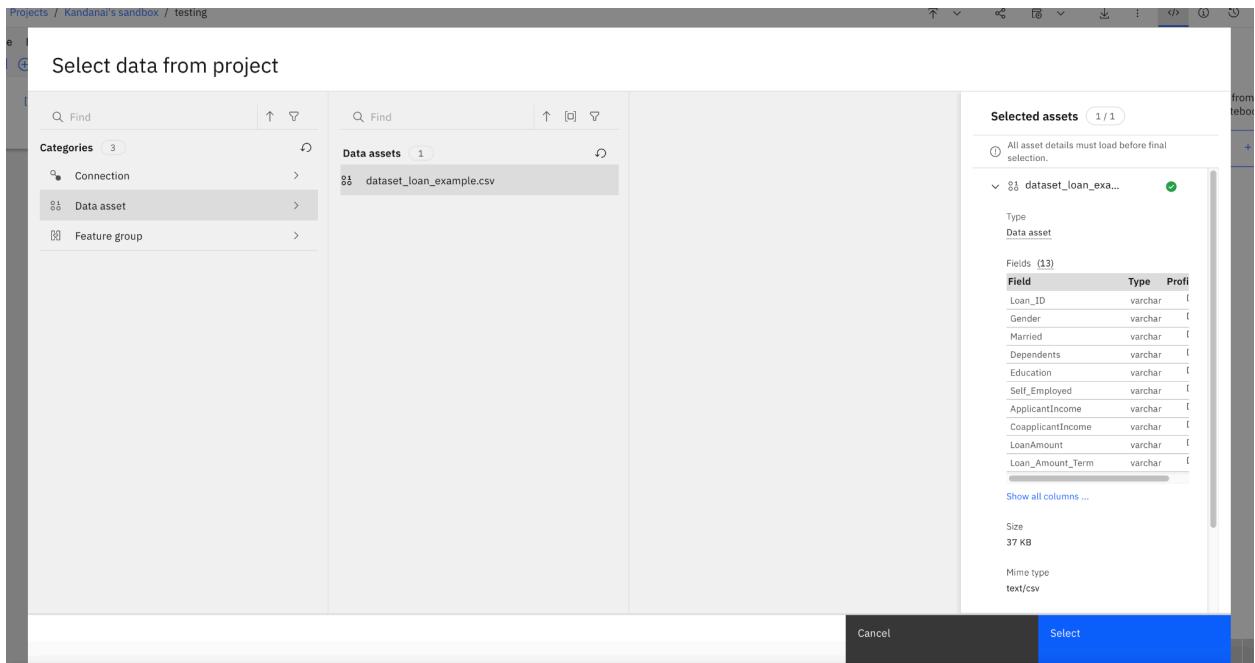
[...] Prompt engineering

Add a sample prompt

Add a prompt from the Samples collection.



4.4) Select the csv file you have just uploaded in the data asset tab, then click insert code to cell



4.5) Click “insert code to cell” and Watson studio will automatically generate code to create connections with the data asset and write code to create pandas dataframe. Run the code to view the result.

```

File Edit View Run Kernel Help
Trusted Memory:299 / 8192 MB
Python 3.11

[1]:
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.

cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='JMoqmtJ2NFSXKA_gldoKKh50f4abA8jBf6a553',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/identity/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us-south.cloud-object-storage.appdomain.cloud')

bucket = 'testpro-donotdelete-pr-xu1scbsjm72g'
object_key = 'dataset_loan_example_v2.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_1 = pd.read_csv(body)
df_1.head(10)

```

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Y
0	Male	No	0.0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	Male	Yes	1.0	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	Male	Yes	0.0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	Male	Yes	0.0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	Male	No	0.0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y
5	Male	Yes	2.0	Graduate	Yes	5417	4196.0	267.0	360.0	1.0	Urban	Y
6	Male	Yes	0.0	Not Graduate	No	2333	1516.0	95.0	360.0	1.0	Urban	Y
7	Male	Yes	3.0	Graduate	No	3036	2504.0	158.0	360.0	0.0	Semiurban	N

## LAB 1 Data Visualization and Creating Connections (Optional)

Optionally, instead of uploading the data directly into the project, we can **create a cloud storage** and create connections to that cloud storage. Here we will create an IBM Cloud Object Storage and repeat the steps before.

Step 1) Go to Infrastructure >> Storage >> Object Storage >> Instances

The screenshot shows the IBM Cloud Infrastructure dashboard. The left sidebar navigation includes sections like Dashboard, Projects, Resource list, Containers, Infrastructure, Observability, Security, API Management, Cloud Pak for Data, Code Engine, Container Registry, DevOps, Interconnectivity, Kubernetes, OpenShift, Partner Center, Quantum, and SAP. Under Infrastructure, the Storage section is expanded, showing Compute, Network, and Storage. The Storage section has three sub-options: Block storage volumes, Block storage snapshots, and Backup policies. Below these is a red box highlighting the 'Object storage' option. To the right of the main navigation bar, there are cards for Watson Assistant, API Connect, and a section titled 'Planned maintenance' which lists '7 Upcoming events'.

Step 2) Create a custom bucket

The screenshot shows the 'Instances' page for Object Storage. The top navigation bar includes a search bar, Catalog, Manage, and a specific instance identifier '2695a03 - cz-watson-13'. Below the navigation is a breadcrumb trail 'Instances / itzcos-6910005ccg-fj156br8'. The main content area has tabs for Buckets, Service credentials, Instance Usage, and Plan. The Buckets tab is selected and displays a table of existing buckets. A red box highlights the 'Create bucket' button at the top right of the table header. The table columns include Name, Public access, Location, Storage class, and Created. The first three rows of the table are:

Name	Public access	Location	Storage class	Created
charanopsandbox-donotdelete-pr-mssayp0t7pyil	No	United States - Dallas (us-south)	Standard	2024-09-20 4:47 PM
kandanaissaandbox-donotdelete-pr-1evix2j8qj2dh	No	United States - Dallas (us-south)	Standard	2024-09-20 11:16 AM
kandanaissaandbox-donotdelete-pr-3ai0shua4obrnh	No	United States - Dallas (us-south)	Standard	2024-09-20 12:25 PM

Instances / itzcos-6910005ccg-fj156br8 / Create bucket

Get started by creating a bucket to store unstructured data. A bucket is a storage resource available in IBM Cloud Object Storage service. The bucket can be used to organize objects (storage data) along with their metadata. Create a custom bucket of your own, or choose from our pre-defined configurations.

**Create a Custom Bucket**

Create a bucket by selecting bucket configurations that meet your object storage needs.

Free Tier requirement: To use Free Tier with 5GB of free usage for 12 months, create a Smart Tier bucket in any location.

**Quickly get started**

**Template** **Free-Tier**

Create a Smart Tier storage class bucket in a region close to you and a service credential to connect your application.

This template offers Free Tier with 5GB of free storage for 12 months, in us-east.

**Archive your data**

**Template**

Create a Smart Tier storage class bucket in a region close to you with an archive rule and a service credential to connect your application.

**Host a static website**

**Template**

Create a Smart Tier storage class bucket in a region with static web hosting configuration and a service credential to connect your application.

**Create** →      **Create** →      **Create** →      **Create** →

### Step 3) Give an a name to your custom bucket and ensure it is selected in the United States region (us-south)

itzcos-6910005ccg-fj156br8 / Create custom bucket

Unique bucket name  
ibm-enablement-bucket-1

**Bucket naming rules:**

- Must be unique across the **whole** IBM Cloud Object Storage system
- Do not use any personal information (any part of a name, address, financial or security accounts or SSN)
- Must start and end in alphanumeric characters (3 to 63)
- Characters allowed: lowercase, numbers and nonconsecutive dots and hyphens

**Resiliency** View options

A bucket's resiliency is defined by the endpoint used to create it.

**Cross Region** Highest availability

**Regional** Best performance

**Single Site** Data sovereignty

⚠ Resiliency cannot be modified after provisioning

**Location** View options

Select a location where you would like your data to be physically stored.

United States - Dallas (us-south)

⚠ Location cannot be modified after provisioning

**Storage class** View pricing

Choose storage class based on how often you expect to read the stored data. Pricing varies for each option.

**Smart Tier** **Free-Tier enabled**

Smart Tier automatically gives you the lowest storage rate

**Standard**

For active workloads that require higher performance and

**Create bucket**



## Step 4) Upload the provided CSV file in the repo to the cloud storage bucket.

The screenshot shows two views of the IBM Cloud Storage interface for the bucket 'ibm-enablement-bucket-1'.

**Top View:** A modal window titled 'Upload' is open on the right side. It displays the 'Choose upload type' section with 'Standard transfer' selected (indicated by a checked radio button). The 'Standard transfer' description states: '200 MB size limit - Ideal for smaller files such as html, images, and documents.' There is also an unchecked checkbox for 'Enable large file (200 MB+)' uploads from the web on this bucket. Below this, the 'Aspera high-speed transfer' section is shown with its description and an 'Install Aspera' link. The 'Upload files (objects)' section contains three buttons: 'Drag and drop files and folders or click to upload', 'Upload files', and 'Upload folders'. At the bottom of the modal are 'Cancel' and 'Upload' buttons.

**Bottom View:** The main storage interface shows the uploaded file 'dataset\_loan\_example.csv'. The file details are listed as follows:

Object name	Archived	Size	Last modified
dataset_loan_example.csv	(Data Engine)	37.1 KB	2024-09-25 3:18 PM

A drag-and-drop area for uploading new files is visible at the bottom left of the interface.

Step 5) Return to the watsonx.ai interface and create a connection. Please select the datasource to be IBM Cloud Object Storage.

The screenshot shows the watsonx.ai interface with the search bar containing "Search for a task or tool". Below the search bar, there are two main sections: "Recent" and "Prepare data".

**Recent:**

- Work with data and models in Python or R notebooks (with Jupyter notebook editor)
- Connect to a data source (with Connection)

**Prepare data:**

- Connect to a data source (with Connection)
- Ground gen AI with vectorized documents (with Vector indexes)
- Prepare and visualize data (with Data Refinery)
- Define reusable sets of parameters (with Parameter set)

The screenshot shows the "Add connection" dialog. On the left, a sidebar lists various connection types, with "IBM Cloud Object Storage" selected. The main area displays the "IBM Cloud Object Storage" connector details.

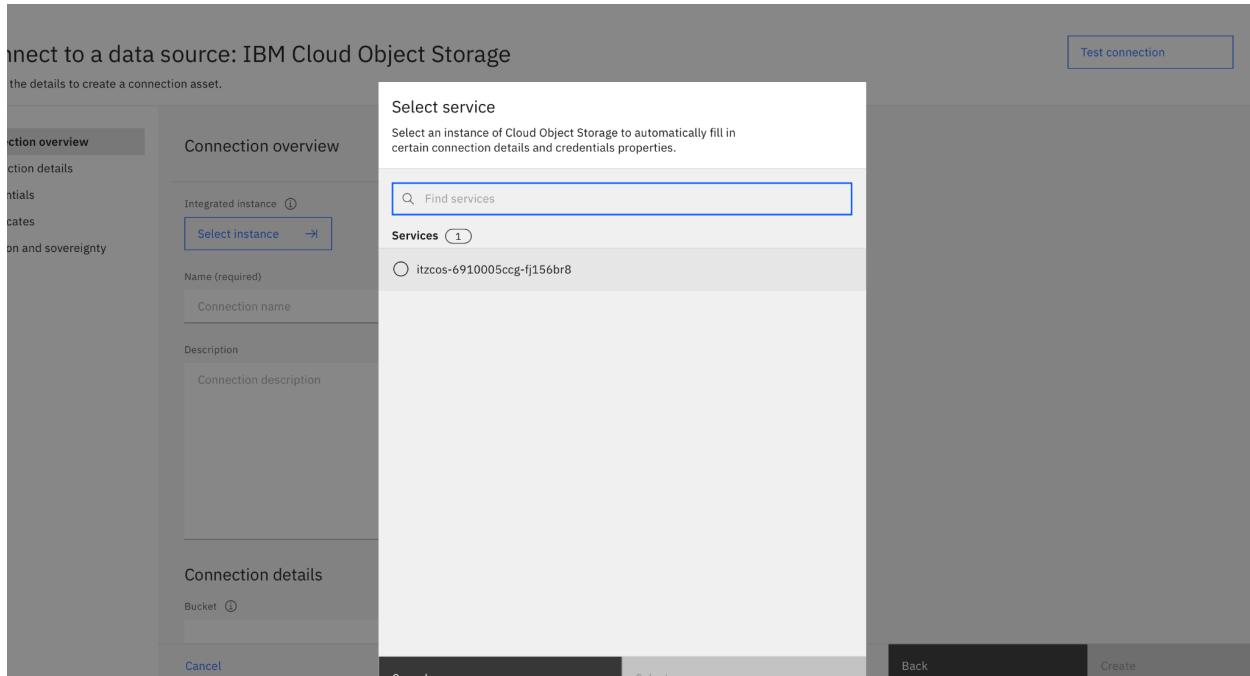
**IBM Cloud Object Storage**

Select a connector:  IBM Cloud Object Storage (infrastructure)

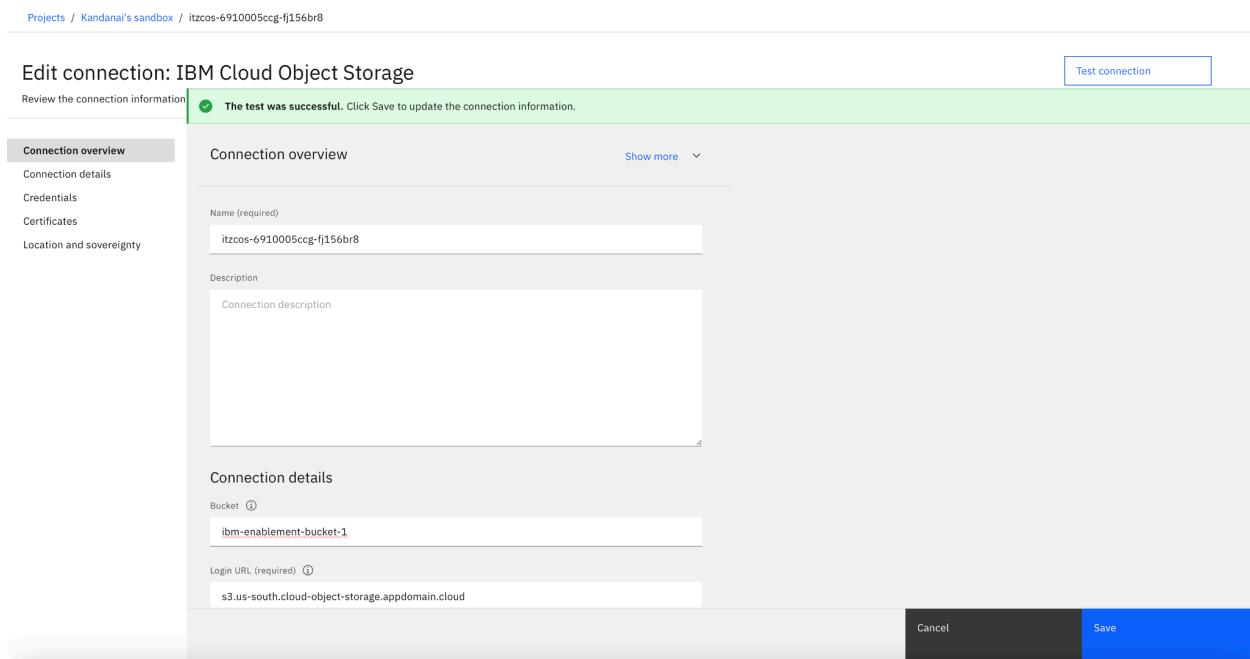
IBM Cloud Object Storage on IBM Cloud provides unstructured data storage for cloud applications. Cloud Object Storage offers S3 API and application binding with regional and cross-regional resiliency.

Compatible tools: Catalogs, DataStage, Data Product Hub, Data Virtualization, Metadata Enrichment, Metadata import, Watson Studio

Cancel Next



Step 6) Choose the corresponding connection and enter your bucket name. Please Test your connection. If the test is successful please save to create your connection.



The screenshot shows the DataBricks interface with the 'Assets' tab selected. On the left, there's a sidebar with '1 assets' and 'All assets'. The main area is titled 'All assets' and lists a single item: 'itzcos-6910005ccg-fj156br8 Connection'. The table has columns for 'Name' and 'Last modified'. A search bar at the top right says 'Find assets'.

Step 7) Now, let us repeat the process of visualizing the data, but this time via connections. Please create a new asset for the data refinery by selecting your uploaded data.

The screenshot shows the DataBricks interface with the 'What do you want to do?' section selected. It asks 'Select a task based on your goal. You'll use a tool to create an asset for that goal.' Below is a search bar with 'data' typed in. The 'Prepare data' section is expanded, showing three options: 'Connect to a data source' (with Connection), 'Prepare and visualize data' (with Data Refinery), and 'Generate synthetic tabular data' (with Synthetic Data Generator). The 'Work with models' section is also partially visible.

Prepare and visualize data

Select data from project

Selected assets (1 / 1)

All asset details must load before final selection.

dataset\_loan\_example.csv

Type: File

Bucket: ibm-enablement-bucket-1

Path: /ibm-enablement-bucket-1/dataset\_lo...

Fields (13)

Field	Type
Loan_ID	varchar
Gender	varchar
Married	varchar
Dependents	varchar
Education	varchar
Self_Employed	varchar
ApplicantIncome	varchar
CoplicantIncome	varchar
LoanAmount	varchar

Cancel Select

Categories (2)

Connections (1)

dataset\_loan\_example.csv

Selected assets (1 / 1)

All asset details must load before final selection.

dataset\_loan\_example.csv

Type: File

Bucket: ibm-enablement-bucket-1

Path: /ibm-enablement-bucket-1/dataset\_lo...

Fields (13)

Field	Type
Loan_ID	varchar
Gender	varchar
Married	varchar
Dependents	varchar
Education	varchar
Self_Employed	varchar
ApplicantIncome	varchar
CoplicantIncome	varchar
LoanAmount	varchar

Cancel Select

Step 8) Once established, create a visualization and view the data profile as before to perform basic EDA.

Prepare and visualize data

Define the details to create a Data Refinery flow asset and open it in the Data Refinery tool.

Select data

Selected asset: dataset\_loan\_example.csv

Define details

Name: visualization

Description (optional): Enter a description

Cancel Back Create

Selected asset: dataset\_loan\_example.csv

Name: visualization

Description (optional): Enter a description

Cancel Back Create

IBM Watsonx

Projects / Kandana's sandbox / Data Refinery

**Steps (1)**

Data source: itzcos-6910005ccg-fj156br8/bm-enablement-bucket-1/dataset\_loan\_example.csv

1. Convert column type

Automatically converted one or more columns to inferred data types. Strings that are converted to decimal use a dot (.) for the decimal symbol.

Auto-generated

**Data**    Profile    Visualizations

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome
1	LP001002	Male	No	0	Graduate	No	584
2	LP001003	Male	Yes	1	Graduate	No	458
3	LP001005	Male	Yes	0	Graduate	Yes	300
4	LP001006	Male	Yes	0	Not Graduate	No	258
5	LP001008	Male	No	0	Graduate	No	600
6	LP001011	Male	Yes	2	Graduate	Yes	541
7	LP001013	Male	Yes	0	Not Graduate	No	233
8	LP001014	Male	Yes	3+	Graduate	No	303
9	LP001018	Male	Yes	2	Graduate	No	400
10	LP001020	Male	Yes	1	Graduate	No	128
11	LP001024	Male	Yes	2	Graduate	No	320
12	LP001027	Male	Yes	2	Graduate	--	250
13	LP001028	Male	Yes	2	Graduate	No	307
14	LP001029	Male	No	0	Graduate	No	185
15	LP001030	Male	Yes	2	Graduate	No	129
16	LP001032	Male	No	0	Graduate	No	495
17	LP001034	Male	No	1	Not Graduate	No	359
18	LP001036	Female	No	0	Graduate	No	351
19	LP001038	Male	Yes	0	Not Graduate	No	488
20	LP001041	Male	Yes	0	Graduate	--	260

New step +

Configure Viewing: 614 rows, 13 columns

Full data set: 614 rows, 13 columns

About this asset

Name: visualization  
Data Refinery flow

Description: What is the purpose of this Data Refinery flow?

Asset details: Steps: 1

Associated assets: Source: itzcos-6910005ccg-fj156br... Target: itzcos-6910005ccg-fj156br...

Last modified: Not yet saved  
Created on: Not yet saved

Projects / Kandana's sandbox / Data Refinery

**Steps (1)**

Data source: itzcos-6910005ccg-fj156br8/bm-enablement-bucket-1/dataset\_loan\_example.csv

1. Convert column type

Automatically converted one or more columns to inferred data types. Strings that are converted to decimal use a dot (.) for the decimal symbol.

Auto-generated

**Data**    **Profile**    Visualizations

	Property_Area	Loan_Status
	Semisuburb	FREQUENCY
	Urban	FREQUENCY
	Rural	FREQUENCY

200 300 400

0 50 100 150 200

0 100 200 300 400

range 0 Maximum length 9 Maximum length 1

0 Minimum length 5 Minimum length 1

1 Mean length 6.517915309446254 Mean length 1

1 Unique 3 Unique 2

0.36487831923640485

New step +

About this asset

Name: visualization  
Data Refinery flow

Description: What is the purpose of this Data Refinery flow?

Asset details: Steps: 1

Associated assets: Source: itzcos-6910005ccg-fj156br... Target: itzcos-6910005ccg-fj156br...

Last modified: Not yet saved  
Created on: Not yet saved

Step 9) Lastly, as we have done before, we can import the data frame through connections. And import the data frame in from IBM Cloud Object Storage.

Field	Type
Loan_ID	varchar
Gender	varchar
Married	varchar
Dependents	varchar
Education	varchar
Self_Employed	varchar
ApplicantIncome	varchar
CoapplicantIncome	varchar
LoanAmount	varchar
Loan_Amount_Term	varchar

jects / Kandanai's sandbox / testing

Edit View Run Kernel Help Trusted Memory:227 / 8192 MB Python 3.11

```
[2]: import types
import pandas as pd
import ibm_boto3
from botocore.client import Config

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share your notebook.

itzcos_6910005ccg_fj156br8_client = ibm_boto3.client(
    service_name='s3',
    ibm_api_key_id='xLbTR3EnbC-fPFd2ivddrz5rYYfQ9ZzP8054_t-nZaj',
    ibm_service_instance_id='crn:v1:bluemix:public:cloud-object-storage:global:a:98ff78eb326f477f8447f94142661697:05ef5cf0-21a9-4183-a307-793f612a0f01:',
    ibm_auth_endpoint='https://iam.cloud.ibm.com/identity/token',
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.us-south.cloud-object-storage.appdomain.cloud'
)
bucket = 'ibm-enablement-bucket-1'
object_key = 'dataset_loan_example.csv'

body = itzcos_6910005ccg_fj156br8_client.get_object(Bucket=bucket, Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType(__iter__, body)

df_0 = pd.read_csv(body)
df_0.head(10)
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoaapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Approved
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	Approved
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Approved
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Approved
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Approved
5	LP001011	Male	Yes	2	Graduate	Yes	5417	4196.0	267.0	360.0	1.0	Urban	Approved

Selected data  
Connection itzcos-6910005ccg-fj156br8  
Path /ibm-enablement-bucket-1/dataset\_loan\_example.csv  
Load as pandas DataFrame

Read data  
Generate a code snippet to load data from a data asset or connection into your notebook.

Insert code to cell