



Cloud Pak for Data
Tutorial – Mortgage

Contents

1.	Prerequisites	5
2.	Setting up database and sample data.....	5
3.	Access Credentials	6
3.1.	Access credential for Db2 database.....	6
3.2.	Sign in to Cloud Pak for Data web console as Administrator.....	6
4.	Create Connection	7
4.2.	Navigate to Connections	7
4.3.	Add connection	7
5.	Discover Assets	8
5.1.	Navigate to discover assets	8
6.	Add users	9
7.	Implement Business Glossary.....	12
7.1.	Download Business Glossaries.....	12
7.2.	Import Categories	12
8.	Implement Policies and Rules.....	14
7.2.	Create a policy	15
7.3.	Create a rule	15
7.4.	Add rule to metadata.....	16
8.	Access data as a Data Scientist	17
8.1.	Create analytic project.....	18
8.2.	Assets from Glossary	18
10.	Navigate to data catalog.....	24
11.	Data Virtualization	26
11.1.	Adding a new data source for Db2	26
11.5.	Add virtual table to catalog	28
11.6.	Publish virtualized table.....	28
11.7.	Access information for virtual table	29
11.8.	Deliver Dataset	29
12.	Build Model	31
12.1.	Navigate to analytics project	32
12.2.	Create deployment space	32
12.3.	Create notebook	32

Cloud Pak for Data – Tutorial

12.4. Review and run notebook.....	33
12.5. Test the model	34

Cloud Pak for Data is a single end to end platform for data management, governance and data science analytics. It provides a one stop shop for data scientists, data engineer and data stewards to collaborate on the platform to acquire, govern and extract best insights from the data in the least amount of time.

In this demo, user will use a set of a fictitious mortgage data that available in Db2 database on IBM Bluemix Cloud. User will perform following tasks to predict if a prospective customer may default on their mortgage.

- Create connection from Cloud Pak for Data to Db2 database on cloud
- Discover Db2 assets from Cloud Pak for Data
- Transform the Db2 data on Cloud Pak for Data
- Use analytics dashboard to build visualizations
- Build a simple machine learning model from prediction

1. Prerequisites

- Access to an operational Cloud Pak for Data Instance
- Install Git on the machine that you will use for the tutorial.

2. Setting up database and sample data

Log in to the cluster where Cloud Pak for Data is deployed or log in to a Linux-based system (RedHat or Ubuntu) that can access the cluster over your network.

From your home directory, clone the tutorial sample files:

```
git clone git@github.com:IBM-ICP4D/ICP4DTutorial.git
```

Change to the tutorials directory:

```
cd ICP4XTutorial/tutorials/
```

The sample data-loading utility, `load_samples.sh`, provides an easy way to host a Db2 server and load it with sample data.

Run the following command to view the list of sample data that is provided in the `load_samples.sh` utility:

```
./load_samples.sh -l
```

Run the following command to load the sample data into a Db2 database:

```
./load_samples.sh -t mortgage-002
```

After the loading process completes, an instance of Db2 is hosted on your cluster as a Docker container.

3. Access Credentials

To work through the tutorial, you need access a Db2 database.

3.1. Access credential for Db2 database

For this tutorial you need JDBC connection to access to a Db2 database that hosted locally on Cloud Pak for Data. Following are JDBC connection credential for Db2:

JDBC Host name	<Same IP address as your web console>
Port number	50000
Database name	MORTGAGE
User ID	db2inst1
Password	password
Db2	Version 11.1
JDBC connection string	jdbc:db2://<same IP as Web Console>:50000/MORTGAGE

3.2. Sign in to Cloud Pak for Data web console as Administrator

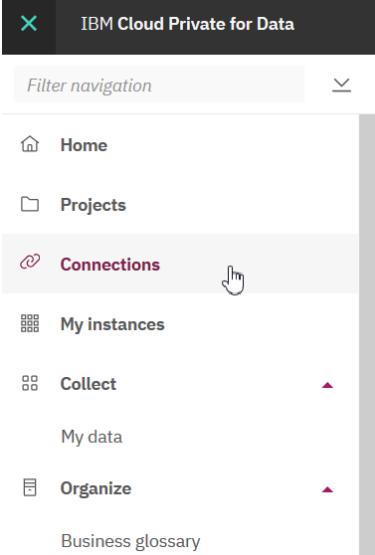
You should have an operational Cloud Pak for Data Instance. Use latest version of Firefox or Google Chrome browser to access the Cloud Pak for Data web console. Starting from here all instruction need to execute on Cloud Pak for Data web console only. You need to login as admin who has administrator privileges.

 Sign in Sign up	<p>Sigh in to the Cloud Pak for Data web console as user ‘admin’ and password is ‘password’.</p>
Username <input type="text" value="admin"/> PASSWORD <input type="password" value="*****"/>	
<input type="button" value="Sign In"/>	

4. Create Connection

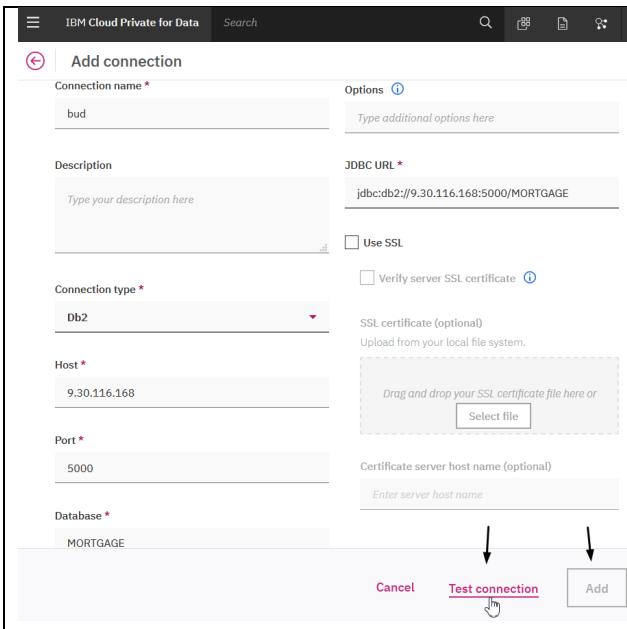
Create a connection to the data source for Db2 database.

4.2. Navigate to Connections



On the left pane choose **Connections**. Next, on the **Data Connections** window click on the  icon.

4.3. Add connection



Fill out the **Add Connection** information according to the information provided in step ‘2.1. Access credential for DB2. Credential used in following step is just an example.

1. For **Choose connection** use the drop-down menu and select ‘Db2’.
2. Use ‘Bud’ as the **Name**
3. **JDBC URL** is ‘`jdbc:db2://172.16.171.29:50000/MORTGAG E'`
4. **Username** is ‘`db2inst1`’ and **Password** is ‘`password`’.

Next click on **Test Connection**, once it successful click on **Save Connection**.



Success The test connection was successful. Click **Add** to save the connection information.

5. Discover Assets

Use the data source created above discover all data assets from Db2 database.

5.1. Navigate to discover assets

6.

From **Organize** option on the left pane, choose **Metadata Curation > Data discovery**.

To select discover job

Navigate to **New discover job > Quick scan**

To discover assets

- Click on Add a connection**
- Choose the connection named **bud** that you created previously, click Next**

Quick scan job

Connection * bud

Discovery root ⓘ schema[MORTGAGE|DB2INST1] [Browse](#)

Discovery options

- Analyze columns
- Analyze data quality
- Assign terms
- Use machine learning to assign terms
- Use data sampling

The maximum number of records included in the data set sample: 1000

Workspace * ⓘ Mortgage

Cancel Discover

3. Choose the connection named **bud** that you created previously.

4. Select **Discover root** as **MORTGAGE > DB2INST1**

5. Check necessary **Discover options**

6. Click on **Add a workspace** under Workspace and named it as **Mortgage**. Click **Create**.

7. Click on **Discover**

It may take few minutes to complete.

Click on **View results** or **View workspaces** to explore the discover assets.

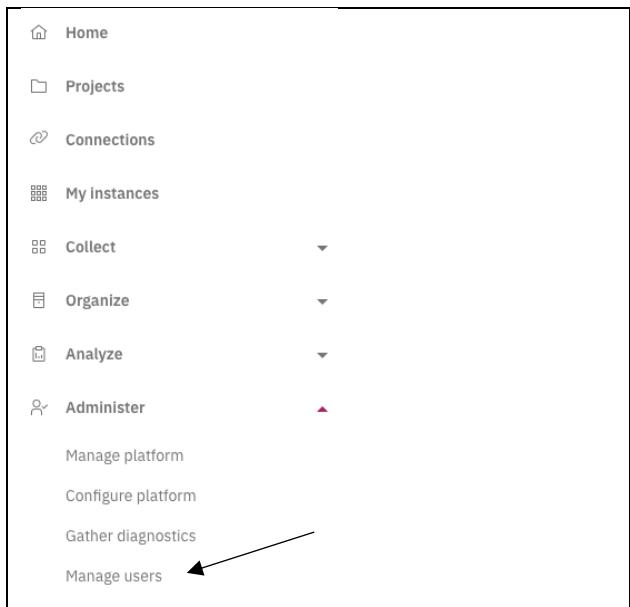
Quick scan results

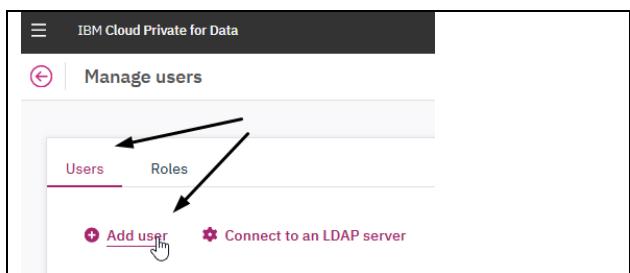
New discovery job ⓘ [View workspaces](#) [View automated discovery results](#)

Summary	Pending analysis	Action required	Reviewed
Status	<input checked="" type="radio"/> All jobs pending analysis <input type="radio"/> Analyzing <input type="radio"/> In queue for analysis	Pause View results	1 item selected (select up to 15) Cancel
		<input checked="" type="checkbox"/> Job ID <input checked="" type="checkbox"/> Data assets <input checked="" type="checkbox"/> Connection <input checked="" type="checkbox"/> Started by <input checked="" type="checkbox"/> Processing time <input checked="" type="checkbox"/> Status <input checked="" type="checkbox"/> Status updated	qs_1571071613091 - bud admin 2 minutes 15 seconds Analyzing -

6. Add users

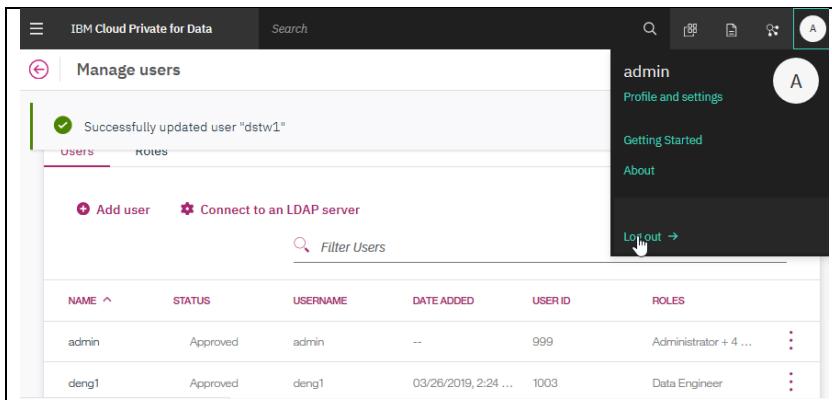
Create users with different roles.

 <ul style="list-style-type: none"> Home Projects Connections My instances Collect Organize Analyze Administer <ul style="list-style-type: none"> Manage platform Configure platform Gather diagnostics Manage users 	<p>From Administer option on the left pane, choose Manage users.</p>
--	--

 <p>IBM Cloud Private for Data</p> <p>Manage users</p> <p>Users Roles</p> <p>Add user Connect to an LDAP server</p>	<p>Switch tab to 'Users' and click on 'Add user'</p>
---	--

 <p>Add user</p> <p>Name: dst1</p> <p>Username: dst1</p> <p>Email: dst1@abc.com</p> <p>User roles:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Administrator <input type="checkbox"/> Business Analyst <input type="checkbox"/> Data Engineer <input checked="" type="checkbox"/> Data Scientist <input type="checkbox"/> Data Steward <p>Cancel Add</p>	<p>Fill out Add User information for a data scientist</p> <ol style="list-style-type: none"> 1. Name as dst1 2. Username is dst1 3. Use a valid email address 4. Set Password as dst1 5. Choose the user roles as Data Scientist <p>Click on Add to confirm the add user</p>
--	---

<p>Follow same steps in Add User section (above) and two more account. Create deng1 for Data Engineer and dstw1 a data steward.</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">User</th> <th style="text-align: center; padding: 5px;">Role</th> <th style="text-align: center; padding: 5px;">Password</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 5px;">• deng1</td> <td style="text-align: center; padding: 5px;">Data Engineer</td> <td style="text-align: center; padding: 5px;">deng1</td> </tr> <tr> <td style="text-align: center; padding: 5px;">• dstw1</td> <td style="text-align: center; padding: 5px;">Data Stewards</td> <td style="text-align: center; padding: 5px;">dstw1</td> </tr> </tbody> </table>	User	Role	Password	• deng1	Data Engineer	deng1	• dstw1	Data Stewards	dstw1
User	Role	Password							
• deng1	Data Engineer	deng1							
• dstw1	Data Stewards	dstw1							



The screenshot shows the 'Manage users' page of the IBM Cloud Private for Data interface. At the top, there is a success message: 'Successfully updated user "dstw1"'. Below this, there are two tabs: 'USERS' (which is selected) and 'ROLES'. There are also links for 'Add user' and 'Connect to an LDAP server'. A search bar labeled 'Filter Users' is present. On the right side, a sidebar for the user 'admin' includes links for 'Profile and settings', 'Getting Started', 'About', and 'Logout' (with a mouse cursor hovering over it). The main table lists two users:

NAME ^	STATUS	USERNAME	DATE ADDED	USER ID	ROLES
admin	Approved	admin	--	999	Administrator + 4 ...
deng1	Approved	deng1	03/26/2019, 2:24 ...	1003	Data Engineer

Log out from user **admin**

7. Implement Business Glossary

Cloud Pak for Data enables you to structure your enterprise information in a logical way, discover relationships between assets, and keep your data always up-to-date. You can import existing glossary with categories, terms, information governance policies and rules.

7.1. Download Business Glossaries

First download business glossaries from the GIT to your local machine.

Go to: <https://github.com/IBM-ICP4D/icp4d-tutorials/tree/master/assets/mortgage-002/BusinessGlossary>

Download all four CSV files and save them locally.

The screenshot shows a GitHub repository interface. At the top, there's a navigation bar with 'Code' selected. Below it, a branch dropdown shows 'master'. The main area displays a folder named 'BusinessGlossary' containing four CSV files:

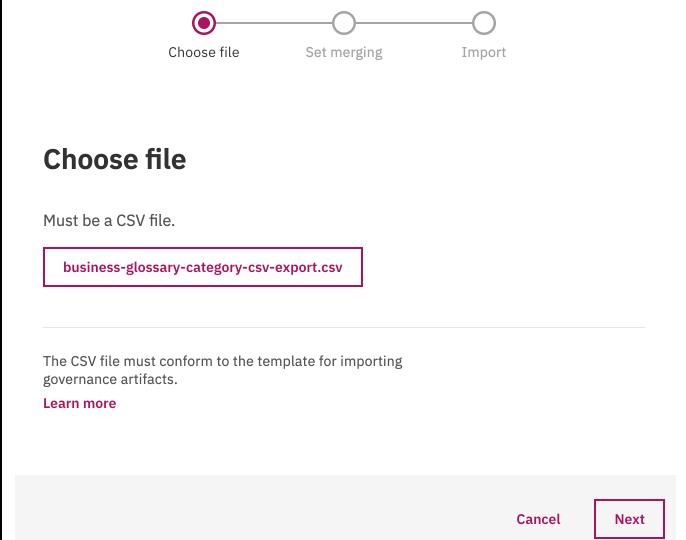
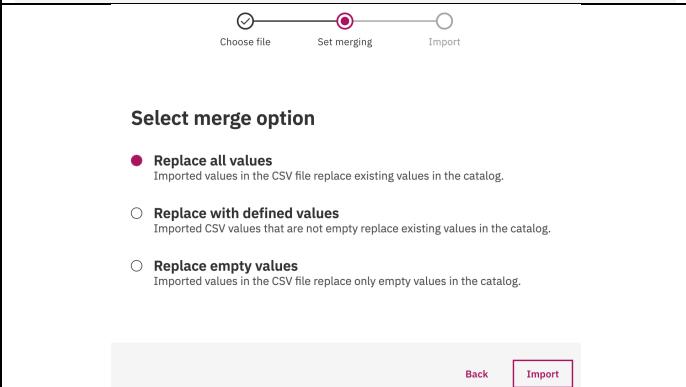
- business-glossary-category-csv-export.csv**: Add files via upload
- business-glossary-policy-csv-export.csv**: Add files via upload
- business-glossary-rule-csv-export.csv**: Add files via upload
- business-glossary-term-csv-export.csv**: Add files via upload

7.2. Import Categories

Sequence is important when importing business glossaries. Make sure import categories before do the terks.

Choose **Organize > Data and AI governance > Categories** from the left pane.

	<p>Click on Import to import the CSV file contains category information that you downloaded from Git.</p>
--	--

	<p>Choose the CSV file location</p> <p>Click Next</p>
	<p>Select merge option as Replace all values</p> <p>Click Import</p>

7.2. Import Terms

<p>Choose Organize > Data and AI governance > Business terms from the left pane.</p> 	<p>Click on Import to import the CSV file contains category information that you downloaded from Git.</p>
---	--

	<p>Choose the CSV file location</p> <p>Click Next</p>
	<p>Select merge option as Replace all values</p> <p>Click Import</p>

8. Implement Policies and Rules

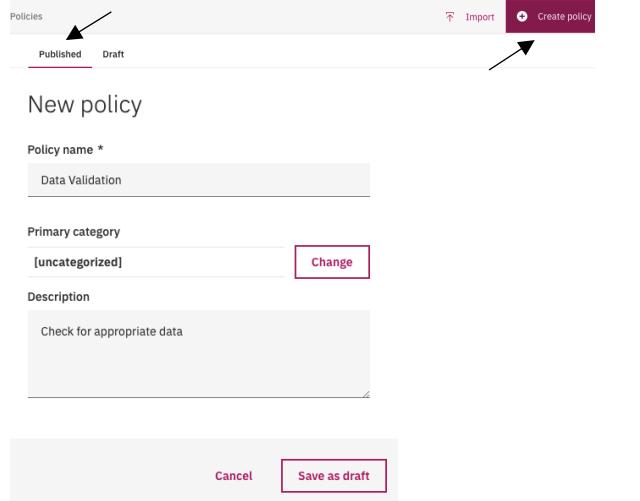
Create governance policies and rules for the entire organization to ensure clarity and compatibility among departments, projects, or products.

	<p>Sigh in to the Cloud Pak for Data web console as user ‘dstw1’ and password is ‘dstw1’ that you created earlier.</p>
--	--

7.2. Create a policy

Choose **Organize > Data and AI governance > Policy** from the left pane

Select **Published** tab and click on **Create Policy**

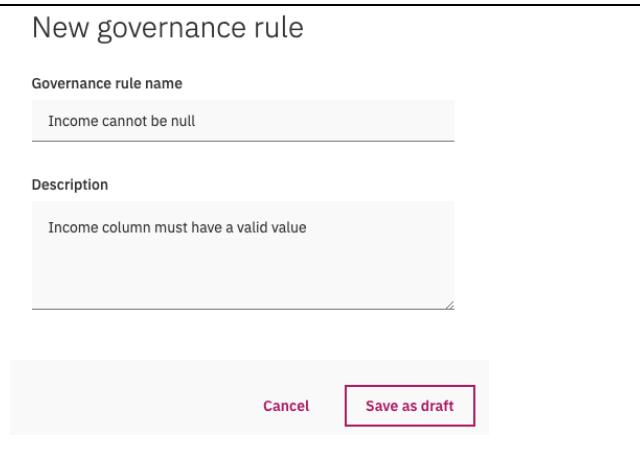
	<p>On the New policy window create a policy with following information and click on Save as draft:</p> <p>Name: Data Validation Description: Check for appropriate data</p> <p>It will take few minutes to appear under list of available policies.</p>
---	--

7.3. Create a rule

Choose **Organize > Data and AI governance > Rule** from the left pane

Select **Published** tab and click on **Create Rule**

Choose **Governance rule**

	<p>On the New governance rule window create a rule with following information and click on Save as draft:</p> <p>Name: Income cannot be null Referencing policies: Data Validation Short Description: Income column must have a valid value</p> <p>It will take few minutes to appear under list of available rules.</p>
---	---

Income cannot be null DRAFT Not started Delete draft Publish

Overview Related content

No secondary category added yet.

▼ Parent policies Add policy

Click on **Add policy** under **Parent policies** to assign the rule to it.

7.4. Add rule to metadata

IBM Cloud Private for Data mortgage_customer

Search Results Relationship Graph Transform Associate to project

MORTGAGE_CUSTOMER 95% RELEVANCY TABLE

MORTGAGE_CUSTOMER 88% RELEVANCY TABLE

T3_MORTGAGE_CUSTOMER 59% RELEVANCY VIEW

MORTGAGE_CUSTOMER
Created by: admin admin

TYPE database_table GROUP Databases CLASS ASCLM0

Click on the enterprise search, Search for 'mortgage_customer' and hit enter
From the search results select table 'mortgage_customer'

Click on **Details** tab at the top

Database Table details MORTGAGE_CUSTOMER

Governance Context: jdbc:db2://10.208.125.50000/MORTGAGE > db2 > DB2INST1

Database Columns (10)

Created by: admin admin

Created on: 04 June 2019, 11:28:49 am

Modified by: InformationServerSystemUser

Modified on: 04 June 2019, 11:28:49 am

APPLIED_ONLINE

CARD_DEBT

CURRENT_LOANS

ID

INCOME

On Database Table Details window choose **Database Columns** from left

Select INCOME column

Next click on icon (right top corner) and choose **Edit**

Scroll down to **Implement Rules** section

Search and select the rule **Income cannot be null** that you created earlier.

Click on **Save**

Database Column details
INCOME

Header (1)

- General Information
- Quality Analysis
- Suggested Term Assignments
- Notes

Assigned to Terms

Add to list Remove all

You haven't added any item yet

Implements Rules

inco Remove all

Income cannot be null

dstw1

Profile and settings

Getting Started

About

Log out →

Log out from user 'dstw1'

8. Access data as a Data Scientist

Explore the data require for build a model

Sign in Sign up

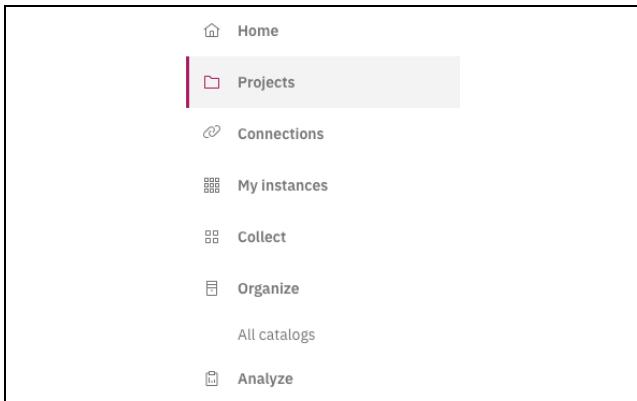
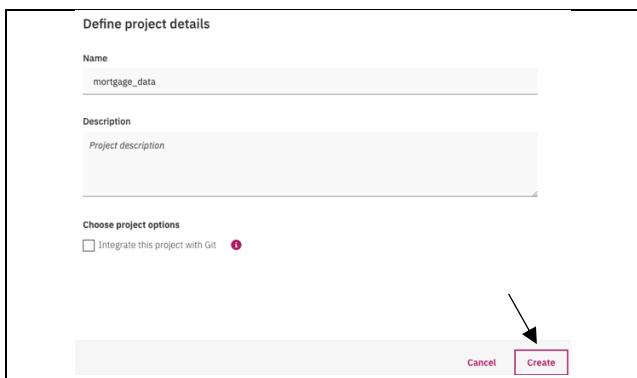
Username dst1

PASSWORD ****

Sign In

Sigh in to the Cloud Pak for Data web console as user 'dst1' and password is 'dst1' that you created earlier.

8.1. Create analytic project

	<p>Create a new analytical project by 'Projects' from right pane.</p> <p>Click on the  New project icon</p> <p>Select Create an empty project</p>
	<p>Provide a project name and click Create</p>

8.2. Assets from Glossary

Let's look for mortgage related terms in glossary to get an idea about different data assets available on the system.

Choose **Organize** from the left pane, select **Data Catalog -> Queries -> Glossary Categories and Terms**.

You should have all mortgage related information as follows. Click on each **ASSET NAME, TERMS** for additional information. The TERM DESCRIPTION provides a basic information about each term.

The screenshot shows a web-based interface for 'IBM Cloud Private for Data'. At the top, there's a navigation bar with a menu icon, the text 'IBM Cloud Private for Data', a search bar labeled 'Search', and three icons for search, refresh, and file operations. Below the header, a breadcrumb path 'Query Details: Glossary Categories and Terms' is displayed. The main content area is titled 'Query Results: 10' and contains a table with four columns: 'Category', 'ASSET NAME', 'CATEGORY DESCRIPTION', 'TERMS', and 'TERM DESCRIPTION'. The table has three rows of data.

Category	ASSET NAME	CATEGORY DESCRIPTION	TERMS	TERM DESCRIPTION
	Address Information	Location related glossary for a JK insurance customer	Customer Zipcode Continuity Of Address Segment Address part 1 Customer City Address part 2 Customer Street Suffix Customer Street Name Customer State Customer House Label Country Of Residence	Current zip code for customer's address Customer City Current suffix for street for customer's address Current street name for customer's address Current state of residence for a customer House number with optional suffix
	Crown Jewels	All data that is sensitive customer info per regulatory obligations	Sensitive Personal Data	Any data deemed to be sensitive personal info for a customer
	Insurance Customer Details	Category for individual insurance customers	Gender Market Segment Summary	Customer's gender, if known Customer Market Segment Summary information about a JKLV insured customer

For example, click on ASSET NAME **Customer**

a. Check Asset Details

Go through each item related to mortgage in glossary to have better idea about data you need for your project.

The asset **Customer** shows different terms associated with it.

Check each **Terms** for additional information.

b. Enterprise search

Click on the enterprise search

Search for 'mortgage' and hit enter

MORTGAGE_PROPERTY

MORTGAGE_PROPERTY
Database Table
idbc:db2://9.30.116.168:50000
/MORTGAGE
db2 >> DB2INST1

★★★★★ 0 Ratings None Quality score

Description

Select your rating:

New Comment:

All Comments (0)

Choose the **mortgage_property** table and click on **Relationship Graph** to see details about the table.

Click on the '+' next to **Database Column** to expand list of columns in the table.

Same way you can view other mortgage related tables.

39 Search Results Filter

mortgage_property	67% RELEVANCY	TABLE
mortgage_default	67% RELEVANCY	TABLE
mortgage_customer	67% RELEVANCY	TABLE
mortgage customer	67% RELEVANCY	TERM
mortgage_join	65% RELEVANCY	TABLE
mortgage_default	64% RELEVANCY	COLUMN

Go back to the enterprise **Search Result**

The enterprise search will return all objects that mentioned word mortgage but as a data scientist you don't have access to any of those objects.

Click on the **New Data Request** on top right corner for request access to mortgage related datasets.

Fill up the **New Data Request** form with detail information as much possible, so a data engineer can provide accurate dataset. Click Confirm and then Submit request.

At this point you need to wait for data engineer to address the data request.

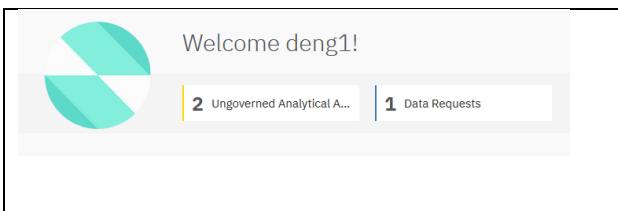
You can go to the home page by clicking on icon from left pane and check the status of the data request.

Sign out from user **dst1**

9. Review data request



Sigh in to the Cloud Pak for Data web console as user ‘deng1’ and password is ‘deng1’ that you created earlier.

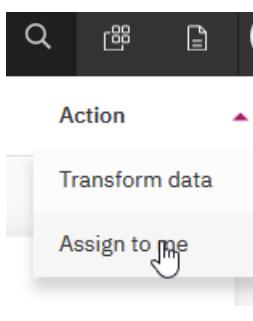


After sing in Click on [Go to your home page](#)

Check the **Data Request** tab on the home page.

Click on the new data request that submitted by data scientist earlier for review.
After reviewing the request click on Action in top right corner and select assign to me.

ID	Name	Status	Requested by	Assigned to	Priority	Last updated
1	Mortgage_Data_Access	Claimed	dst1	deng1	Medium	3 Jun 2019, 8:15 PM
2	Mortgage_Data_Access_Request	Claimed	dst1	deng1	Medium	3 Jun 2019, 8:41 PM
3	Mortgage_Data_request1	Claimed	dst1	deng1	High	3 Jun 2019, 8:39 PM
4	CustData	New	admin	Unassigned	High	4 Jun 2019, 9:03 AM



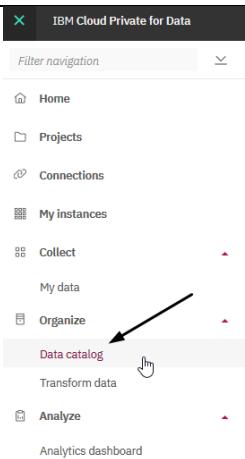
Action ▾

Transform data

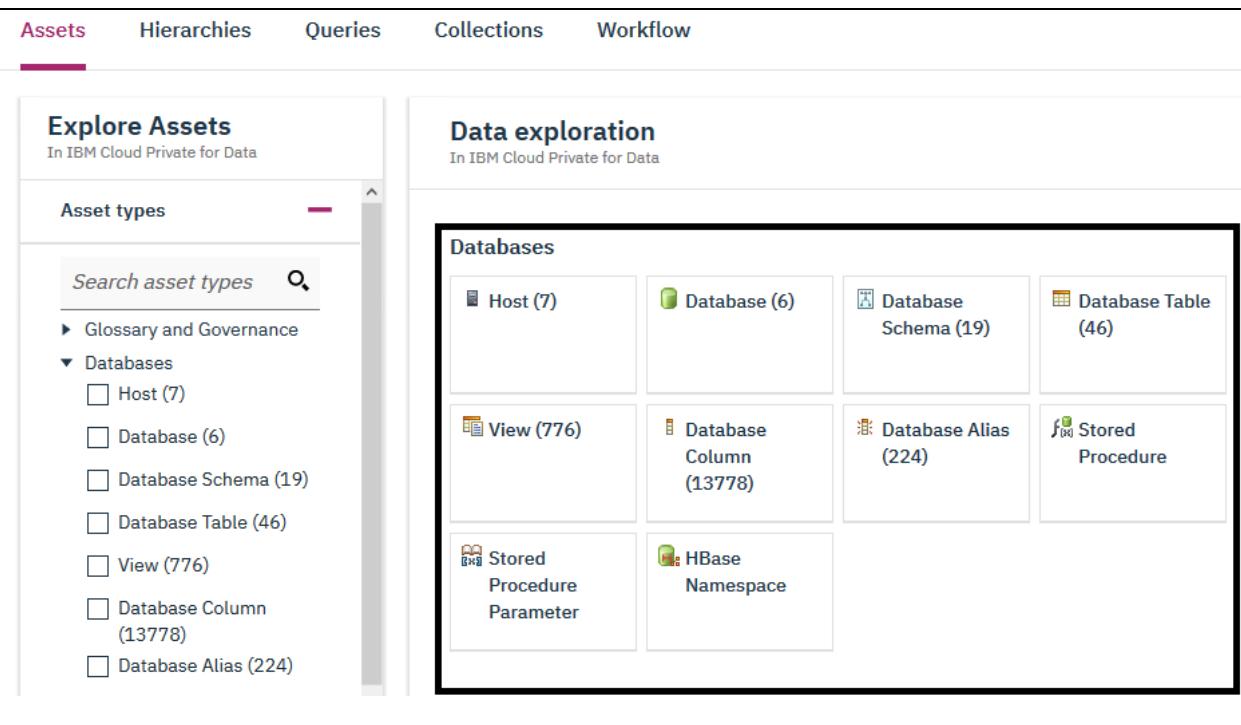
Assign to me

10. Navigate to data catalog

Once discover assets process completed. All database objects automatically cataloged in Cloud Pak for Data. You can review those database object in the catalog.

	<p>Next go back to Organize option on the left pane and choose Data catalog.</p>
---	--

At this point Cloud Pak for Data should displays all the database objects. You can click each individual object under **Databases** to explore the catalog generated from discover asset previously. Click on the **Database Table** to check tables discovered from Db2. Take a look into the database named **mortgage**.



Under the **Database Tables** you can see ‘MORTGAGE_CUSTOMER’, ‘MORTGAGE_DEFAULT’ and ‘MORTGAGE_PROPERTY’ tables, cataloged from Db2 database.

Filter results

[Clear all filters](#)

Asset types (1)

Search asset types

- ▶ Glossary and Governance
- ▼ Databases (1)
 - Host
 - Database
 - Database Schema
 - Database Table (46)
 - View
 - Database Column
 - Database Alias
 - Stored Procedure
 - Stored Procedure Parameter
 - HBase Namespace
- Data Files

All results

46 results

No items selected

<input type="checkbox"/>	/MORTGAGE db2 » SYSTOOLS	Modified by InformationServerSystemUser on Jun 3, 2019, 6:52 PM
<input type="checkbox"/>	HMON_COLLECTION	Modified by InformationServerSystemUser on Jun 4, 2019, 11:30 AM
<input type="checkbox"/>	idbc:db2://10.208.125.125:50000 /MORTGAGE db2 » SYSTOOLS	
<input type="checkbox"/>	MONGO_MORTGAGE_DEFAULT	Created by admin on Jun 3, 2019, 6:34 PM
<input type="checkbox"/>	idbc:db2://dv-server.zen.svc.cluster.local:32051 /bisqal db2 » USER999	Modified by InformationServerSystemUser on Jun 3, 2019, 6:34 PM
<input type="checkbox"/>	MONGO_MORTGAGE_PROPERTY	
<input type="checkbox"/>	idbc:db2://dv-server.zen.svc.cluster.local:32051 /bisqal db2 » USER999	Created by admin on Jun 4, 2019, 11:28 AM
<input type="checkbox"/>	MORTGAGE_CUSTOMER	
<input type="checkbox"/>	idbc:db2://10.208.125.125:50000 /MORTGAGE db2 » DB2INST1	

11. Data Virtualization

Context: Data virtualization (DV) integrates data sources across multiple types and locations and turns it into one logical data view. In this case, you have data across three different tables. Creating a virtual table you can quickly view data from different tables.

11.1. Adding a new data source for Db2

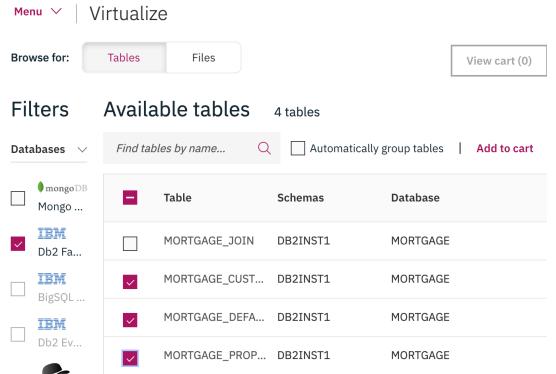
Context: DV supports many relational and non-relational data sources (as well as files that reside on a local disk or network file system) that you can add to your data source ecosystem. After a data source has been added, any user that has virtualize permission can create virtual tables. DV agents connect to relational data sources using JDBC protocol. In this tutorial you will add a data source for Db2 database.

Define a data connection to Db2. Use your existing Db2 database connection for Db2 data source.

1. Go to **Collect > Virtualized data > Menu > Data sources**
2. Click **Add > New data source > Add connection**
3. Select **Db2** that you created earlier and click **Next**

11.3. Select tables for virtualization

Context: the most common mechanism for virtualizing data is to create a "view" or virtual table. Virtual tables can be full or segment of data from one or more tables. You can then run queries against the resulting virtual table.

<ul style="list-style-type: none"> • Click Collect > Virtualized data > Menu > Virtualize • Select tables MORTGAGE_CUSTOMER, MORTGAGE_PROPERTY and MORTGAGE_DEFAULT from MORTGAGE database, then click Add to cart • Click View cart • Click Next 	 <p>The screenshot shows the 'Virtualize' interface. At the top, there's a 'Menu' dropdown with 'Virtualize' selected. Below it, a 'Browse for:' section with 'Tables' and 'Files' buttons, and a 'View cart (0)' button. The main area is titled 'Available tables' with '4 tables'. A 'Filters' section includes a 'Databases' dropdown set to 'IBM' and a search bar. A table lists the available tables:</p> <table border="1"> <thead> <tr> <th></th> <th>Table</th> <th>Schemas</th> <th>Database</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>MORTGAGE_JOIN</td> <td>DB2INST1</td> <td>MORTGAGE</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>MORTGAGE_CUST...</td> <td>DB2INST1</td> <td>MORTGAGE</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>MORTGAGE_DEFA...</td> <td>DB2INST1</td> <td>MORTGAGE</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>MORTGAGE_PROP...</td> <td>DB2INST1</td> <td>MORTGAGE</td> </tr> </tbody> </table>		Table	Schemas	Database	<input type="checkbox"/>	MORTGAGE_JOIN	DB2INST1	MORTGAGE	<input checked="" type="checkbox"/>	MORTGAGE_CUST...	DB2INST1	MORTGAGE	<input checked="" type="checkbox"/>	MORTGAGE_DEFA...	DB2INST1	MORTGAGE	<input checked="" type="checkbox"/>	MORTGAGE_PROP...	DB2INST1	MORTGAGE
	Table	Schemas	Database																		
<input type="checkbox"/>	MORTGAGE_JOIN	DB2INST1	MORTGAGE																		
<input checked="" type="checkbox"/>	MORTGAGE_CUST...	DB2INST1	MORTGAGE																		
<input checked="" type="checkbox"/>	MORTGAGE_DEFA...	DB2INST1	MORTGAGE																		
<input checked="" type="checkbox"/>	MORTGAGE_PROP...	DB2INST1	MORTGAGE																		

- Select **Neither**
- Uncheck the box for **Submit to catalog**
- Click **Virtualize** to complete the process

Table	Schema	Source schema	Host/Database	Grouped tables
MORTGAGE_CUSTOMER	USER999	X ▾	DB2INST1	169.46.33.180:MORTGAGE
MORTGAGE_DEFAULT	USER999	X ▾	DB2INST1	169.46.33.180:MORTGAGE
MORTGAGE_PROPERTY	USER999	X ▾	DB2INST1	169.46.33.180:MORTGAGE

11.4. Creating virtual table

You can create a new virtual table based on existing tables under **My data** section. You can use “drag and drop” or write your own SQL to create the view.

- Click **Collect > Virtualized data > Menu > SQL editor** to access the editor.
- Copy the following SQL statement and paste it on the editor
- Click on **Run all**

```
CREATE VIEW MORTGAGE_JOIN_VIEW
AS
SELECT A.ID, INCOME, APPLIED_ONLINE, RESIDENCE, YRS_CURRENT_ADD,
       YRS_CURRENT_EMP, NO_OF_CARDS, CARD_DEBT, CURRENT_LOANS,
       LOAN_AMOUNT, SALE_PRICE, LOCATION, MORTGAGE_DEFAULT
FROM   MORTGAGE_CUSTOMER A,
       MORTGAGE_PROPERTY B,
       MORTGAGE_DEFAULT C
WHERE  A.ID = B.ID
AND    A.ID = C.ID;
```

Menu ▾ | SQL editor

* Untitled - 1

```
1
2
3 CREATE VIEW MORTGAGE_JOIN_VIEW
4 AS
5 SELECT A.ID, INCOME, APPLIED_ONLINE, RESIDENCE, YRS_CURRENT_
6       YRS_CURRENT_EMP, NO_OF_CARDS, CARD_DEBT, CURRENT_LOAN
7       LOAN_AMOUNT, SALE_PRICE, LOCATION, MORTGAGE_DEFAULT
8 FROM   MORTGAGE_CUSTOMER A,
9       MORTGAGE_PROPERTY B,
10      MORTGAGE_DEFAULT C
11 WHERE  A.ID = B.ID
12 AND    A.ID = C.ID;
13
```

- Click **Collect > Virtualized data > Menu > My virtualized data** to access the virtual table **MORTGAGE_JOIN_VIEW**
- Check the box associated with **MORTGAGE_JOIN_VIEW**
- Click on the table actions menu

- Select **Manage access** option
- On grant access window select All data virtualization users
- Click **Continue**

Grant access to

All data virtualization users [?](#) Specific users [?](#)

Users Roles

Search [?](#) [Revoke](#) [?](#) [Grant access](#)

<input type="checkbox"/>	Name	Username	Role	User ID	Access level
<input type="checkbox"/>					

11.5. Add virtual table to catalog

Once you create a virtual table, you can add it to the catalog, making it easily searchable.

- Click **Collect > Virtualized data > Menu > My data** to find the virtual table just created.
- Mark the checkbox associated with virtual table
- Choose **Submit to catalog** from table action
- Click on **Confirm**

Menu [?](#) | My virtualized data

Find [?](#)

Assign Join view [Add tab](#)

Total tables: 23 [?](#) Access to some tables is restricted by policies. [?](#)

<input type="checkbox"/>	Table	Schema	Created on
<input checked="" type="checkbox"/>	MORTGAGE_JOIN_VIEW	USER999	18 Oct 2019 20:17:28
<input type="checkbox"/>	V2	USER999	18 Oct 2019 20:09:27
<input type="checkbox"/>	V1	USER999	18 Oct 2019 20:08:40
<input type="checkbox"/>	MORTGAGE_PROPERTY	USER999	18 Oct 2019 20:08:40

11.6. Publish virtualized table

A data steward needs approve the published request before the asset is added to the enterprise data catalog. You signed in as user 'admin', it should allow to publish the virtual table.

Pending Publish to Catalog Requests

Search [?](#)

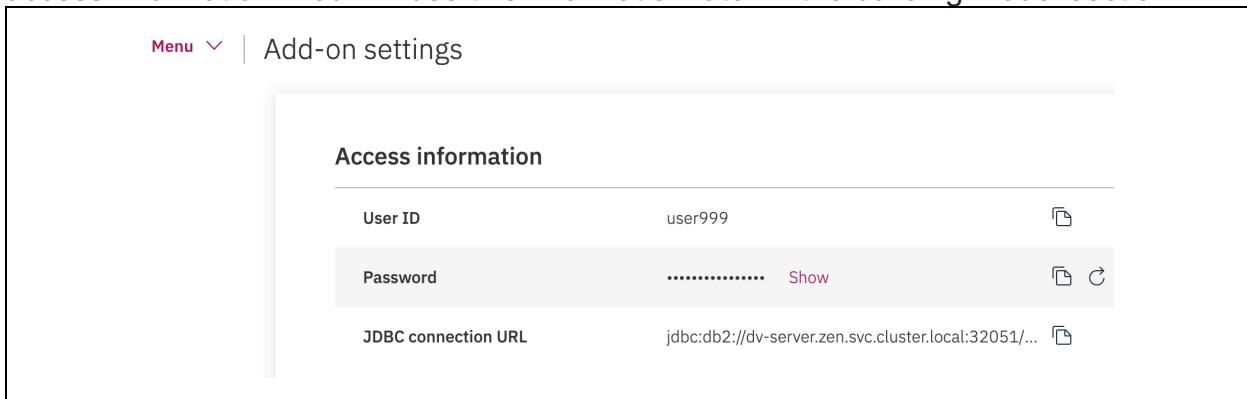
Name	Type	Project	Owner	Date Updated	Status
> USER999.MORTGAGE_JOIN_VIEW	view	-	admin	21 October 2019, 2:49PM	Pending
> USER999.Currency USER999.Country	table	-	admin	17 October 2019, 8:40AM	Pending

• Click on [Home](#) access the **Home** page
 • Click on **Pending Publish to Catalog Requests**

	<ul style="list-style-type: none"> • Click on  icon on left for virtual table MORTGAGE_JOIN_VIEW that you created • Click on Approve
--	---

11.7. Access information for virtual table

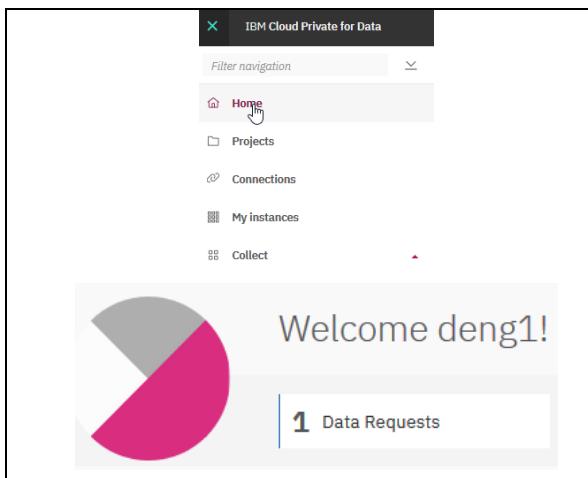
To access virtual table from external application, you need the JDBC connection information. Click on **Collect > Virtualized data > Menu > Add-on settings** to find out access information. You will use this information later in the building model section.



The screenshot shows the 'Add-on settings' page under the 'Menu' dropdown. It displays the following access information:

Access information	
User ID	user999
Password Show
JDBC connection URL	jdbc:db2://dv-server.zen.svc.cluster.local:32051/...

11.8. Deliver Dataset

	<p>Go to the home page by clicking on  icon from left pane and check the data request tab.</p>
---	---

Click on the data request for update that submitted by data scientist earlier.

Data requests + Add new data request

	Name	ID	Status	Last Updated
1	Mortgage_Data_Access	2	New	27 Mar 2019, 11:15 AM

Click on the **Source** and fill out all the necessary information. This information will be picked up by the data scientist later.

Add the **remote data** set information that you created during data transformation. In this case remote data set is MORTGAGE_JOIN_VIEW. Use the **Access information** from the **Add-on settings** information from DV.

New data request

Overview Columns **Source**

Source

Data source name
mortgage_join

DB2

Username
db2inst1

Password
.....

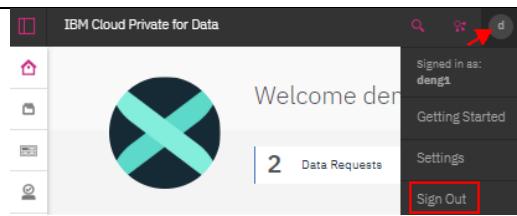
JDBC URL
169.45.83.218

+ Add new dataset

	Remote data set name	Description	Schema	Table
1	mortgage_join		db2inst1	mortage_join

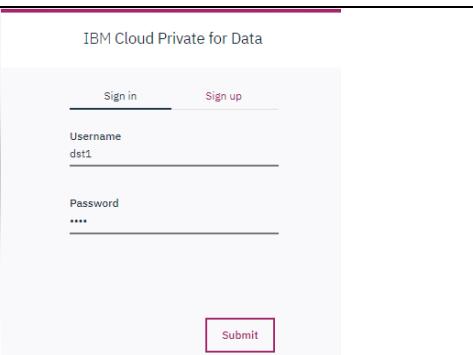
Click on the data request and change the status to **Deliver**.

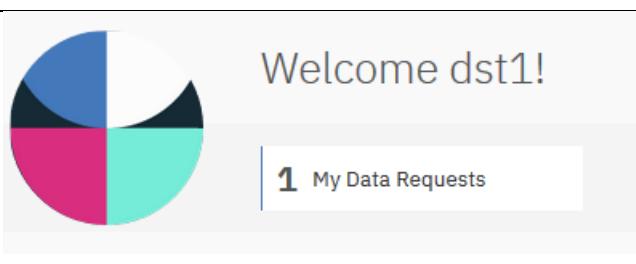
NAME	ID	STATUS	REQUESTED BY	ACCEPTED BY	LAST UPDATED	ACTIONS
mortgagedata1	7	Delivered	dst1	deng1	6 Aug 2018, 12:55 PM	
Mortgage_Data_Access	9	Accepted	dst1	deng1	15 Aug 2018, 1:17 AM	


Sign out from user **deng1**

12. Build Model

With Cloud Pak for Data, you can collaborate with other team members on analytic projects to create visualizations and machine learning models with data from your enterprise. In this step you will build a simple model to predict the possibilities of mortgage default by customer. The object of this model is to show the functionality of Cloud Pak for Data, not the prediction accuracy. One can use lot more data and build a complex algorithm to get better accuracy.


Sign in to the Cloud Pak for Data web console as user ‘dst1’ and password is ‘dst1’ that you created earlier.



At this point data engineer deliver the data set for the data you requested.
You can go to the home page by clicking on icon from left pane

12.1. Navigate to analytics project

Select **Projects** option from the left pane and click on the analytics project ‘mortgage_data’ that you created earlier.

12.2. Create deployment space

Create a separate deployment space for your project ‘mortgage_data’.

Choose : My Projects > **mortgage_data** > Settings > Associate a deployment space > New

Connect to a deployment space

New Existing

Name

MortgageDeploymentSpace

Description (Optional)

Description of deployment space

Cancel Associate

Name new deployment space as
‘MortgageDeploymentSpace’

Click on **Associate**

12.3. Create notebook

Create a notebook from a predefined Jupyter notebook that available on Github.

- Go to : My Projects > **mortgage_data** > Add to project
- Choose asset type as Notebook
- The new notebook needs to create from URL
- Name the notebook as **MortgageNotebook**
- Use notebook URL as <https://github.com/IBM-ICP4D/icp4d-tutorials/blob/master/assets/mortgage-002/MortgageNotebook.V25.jupyter-py36.ipynb>
- Click on **Create Notebook**

My Projects > mortgage_data > Add Notebook

New notebook

Blank From file From URL

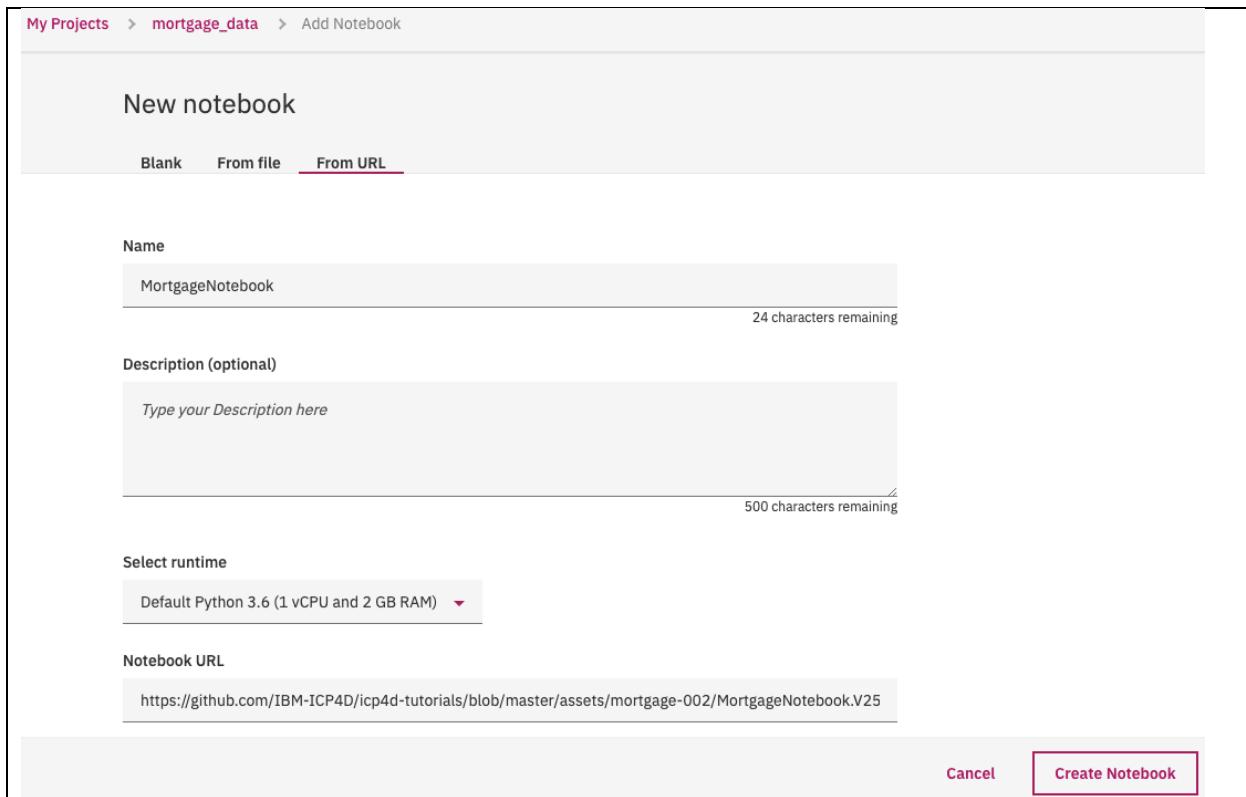
Name
MortgageNotebook 24 characters remaining

Description (optional)
Type your Description here 500 characters remaining

Select runtime
Default Python 3.6 (1 vCPU and 2 GB RAM) ▾

Notebook URL
<https://github.com/IBM-ICP4D/icp4d-tutorials/blob/master/assets/mortgage-002/MortgageNotebook.V25>

[Cancel](#) [Create Notebook](#)



12.4. Review and run notebook

The majority of the code in the notebook is standard open source code that's used for various steps in the predictive analytics process.

Switch to edit mode by clicking on  icon from top of the screen.

Do not run all cells at once. Follow the instruction below to run the notebook.

Run the **Step 1: Intall** section first. Once all package installed make sure resttart the Python kernel before move on next step.

The screenshot shows a Jupyter Notebook interface. At the top, there's a navigation bar with 'My Projects' → 'mortgage_data' → 'MortgageNotebook'. Below it is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. An arrow points to the 'Kernel' menu item. A dropdown menu is open under 'Kernel' with the following options: 'Interrupt', 'Restart', 'Restart & Clear Output' (which is highlighted with a red box), 'Restart & Run All', 'Reconnect', 'Change kernel' (with a submenu 'client-v4'), and 'Learning client' with 'g-client -y'. The main notebook area has two code cells. The first cell (In []:) contains Python code to uninstall and install WML, verify its version, and install findspark and pyspark. The second cell (In []:) also contains similar setup code.

Action: restart the kernel!

Go the **Step 2: Authenticate** section and update the **url**, **username** and **password** fields with your CPD UI console details and access credential.

Step 2: Authenticate

```
[ ]: WML_CREDENTIALS = {
    "instance_id": "openshift",
    "url" : "https://zen-cpd-zen.apps.testcluster.demo.ibmcloud.com",
    "username": "admin",
    "password": "passw0rd",
    "version": "2.5.0"
}
```

In the next notebook cell, update the **dsn_url**, **dsn_uid** and **dsn_pwd** values with the information available from **Collect > Virtualized data > Menu > Add-on settings**.

```
[ ]: #Enter the values for your database connection found under data virtualization
dsn_url = "jdbc:db2://dv-server.zen.svc.cluster.local:32051/biggsql" # e.g.
dsn_uid = "user1022" # e.g.
dsn_pwd = "sw?#@LT_674MfPI5" # e.g.
```

Run all cells between step 2 and 6.

On **Step 7: Set default space**, run the first cell and find out the **GUID** for space name **MortgageDeploymentSpace**.

On the next cell replaced the **GUID** with one that you found above.

```
In [ ]: # Example: client.set.default_space('b49e13e8-ec68-408d-84a1-957e28c154b1')
client.set.default_space('GUID')
```

Run through remaining cells, so that it generates and deploys the model.

Before exit, save the notebook .

12.5. Test the model

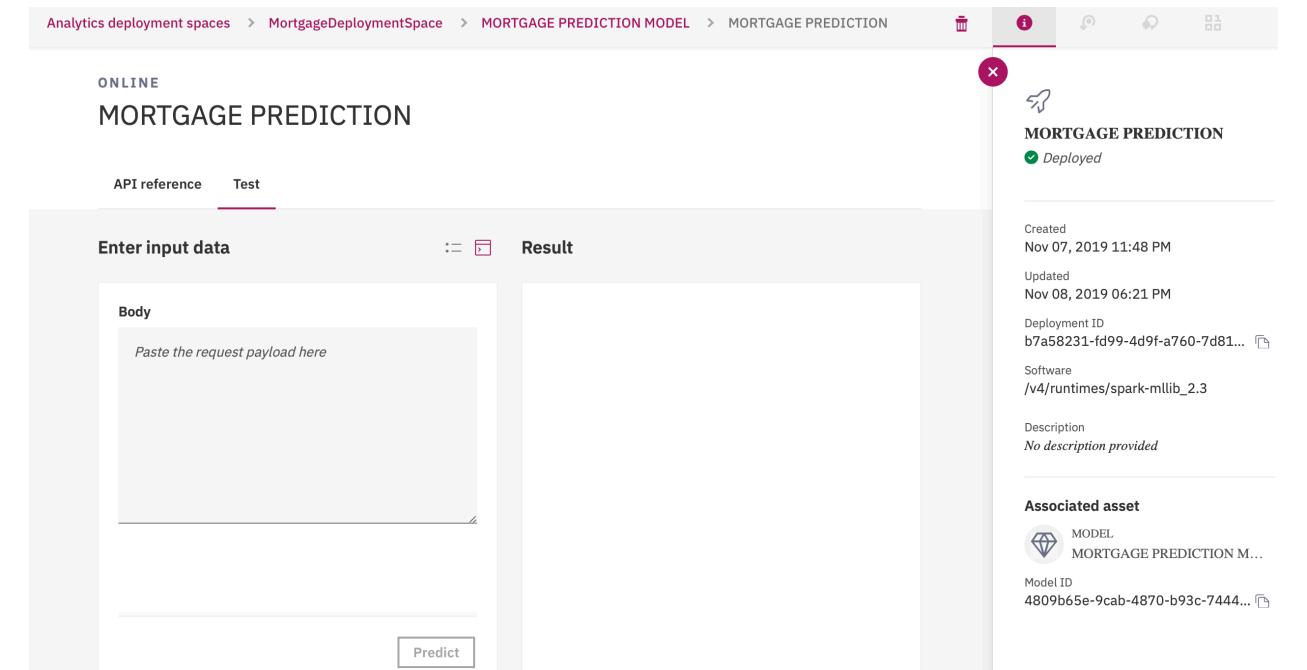
Go to: Analyze > Analytics deployment to access deployed model

Select the **MortgageDeploymentSpace** from the list of analytic deployment space

Click on the **MORTGAGE PREDICTION MODEL**

Choose the **MORTGAGE PREDICTION** model

Click on **Test** tab



The screenshot shows the 'MORTGAGE PREDICTION' model page in the 'Test' tab. On the left, there's a 'Enter input data' section with a 'Body' field containing placeholder text 'Paste the request payload here'. On the right, there's a 'Result' section which is currently empty. Below these are 'Associated asset' details: Model ID b7a58231-fd99-4d9f-a760-7d81..., Software /v4/runtimes/spark-mllib_2.3, and Description No description provided. At the bottom, there's a 'Predict' button.

<pre>{ "input_data": [{ "fields": ["INCOME", "APPLIED_ONLINE", "RESIDENCE", "YRS_CURRENT_ADD", "YRS_CURRENT_EMP", "NO_OF_CARDS", "CARD_DEBT", "CURRENT_LOANS", "LOAN_AMOUNT", "SALE_PRICE", "LOCATION"], "values": [[43151, "N", "P", 6, 9, 1, 750, ...]] }] }</pre>	<p>Copy this sample data and paste it on the Enter input data box.</p> <p>Click on Predict</p>
--	--

```

1,
8600,
320000,
110
]
}
]
}
}

```

According on input values, model will predict and displays the result.

The screenshot shows the Cloud Pak for Data interface for the Mortgage Prediction Model. The top navigation bar includes 'Analytics deployment spaces' > 'MortgageDeploymentSpace' > 'MORTGAGE PREDICTION MODEL' > 'MORTGAGE PREDICTION'. The main area is titled 'ONLINE MORTGAGE PREDICTION' with tabs for 'API reference' and 'Test'. The 'Test' tab is selected, showing an 'Enter input data' section with a 'Body' field containing JSON input and a 'Predict' button. To the right, the 'Result' section displays the predicted output. On the far right, there is a detailed view of the 'MORTGAGE PREDICTION' model, including its status ('Deployed'), creation date ('Nov 07, 2019 11:48 PM'), update date ('Nov 08, 2019 06:32 PM'), deployment ID ('b7a58231-fd99-4d9f-a760-7d81...'), software ('/v4/runtimes/spark-mllib_2.3'), and description ('No description provided'). Below this is an 'Associated asset' section with a diamond icon and the text 'MODEL MORTGAGE PREDICTION M...', followed by a model ID ('4809b65e-9cab-4870-b93c-7444...').

```

{
  "input_data": [
    {
      "fields": [
        "INCOME",
        "APPLIED_ONLINE",
        "RESIDENCE",
        "YRS_CURRENT_ADD",
        "YRS_CURRENT_EMP",
        "NO_OF_CARDS",
        "CARD_DEBT",
        "CURRENT_LOANS",
        "LOAN_AMOUNT",
        "SALE_PRICE",
        "LOCATION",
        "MORTGAGE_DEFAULT",
        "AppliedOnlineEncoded"
      ]
    }
  ],
  "predictions": [
    {
      "fields": [
        "INCOME",
        "APPLIED_ONLINE",
        "RESIDENCE",
        "YRS_CURRENT_ADD",
        "YRS_CURRENT_EMP",
        "NO_OF_CARDS",
        "CARD_DEBT",
        "CURRENT_LOANS",
        "LOAN_AMOUNT",
        "SALE_PRICE",
        "LOCATION",
        "MORTGAGE_DEFAULT",
        "AppliedOnlineEncoded"
      ]
    }
  ]
}

```