Project 7: Data Warehousing with IBM Cloud Db2 Warehouse

Phase 5: Project Documentation & Submission.

PROJECT OBJECTIVE

Our project main ob to tackle the challenge of modernizing our data management by designing and implementing a robust data warehouse using IBM Cloud Db2 Warehouse. This project is driven by the need to harness the potential of the organization's data, coming from diverse sources, and to empower data architects with the tools needed for insightful data analysis and informed decision-making.

The project encompasses the following key objectives:

i. Data Warehouse Structure: We will define a flexible and scalable schema and structure for our data warehouse that can efficiently handle data from various sources.

ii. Data Integration: The project will involve identifying key data sources within our organization and devising a strategy for integrating this data seamlessly into the data warehouse.

iii. ETL (Extract, Transform, Load) Processes: To ensure data accuracy and relevance, we will develop robust ETL processes for extracting data from source systems, transforming it into a usable format, and loading it efficiently into the data warehouse.

iv. Data Exploration: We aim to create an intuitive and interactive environment that allows data architects to explore and analyze the data effectively.

v. Actionable Insights: Ultimately, our goal is to deliver actionable insights from the data analysis, enabling our organization to make data-driven decisions.

DESIGN THINKING

➢ Data Warehouse Structure

To define the data warehouse structure, we will take the following practical steps:

- Data Profiling: We will conduct thorough data profiling to gain insights into the data's characteristics, including data types, relationships, and quality.
- Entity-Relationship Diagram (ERD): We will create an ERD to visualize the data model, making it easier to design schemas and tables.

➢ Data Integration

For effective data integration, we will follow a realistic approach:

- Source Identification: We will identify all potential data sources within the organization, including databases, third-party APIs, and legacy systems.
- Data Extraction Strategy: We will determine data extraction methods and frequency, aligning them with our organization's data needs.
- Transformation Rules: We will establish clear data transformation rules and procedures to maintain data consistency and quality.

➢ ETL Processes

Pragmatic ETL processes will be designed and implemented as follows:

- ETL Tool Selection: We will select ETL tools or custom scripting languages based on our organization's resources and expertise.

- **Data Exploration**

To promote practical data exploration, we will adopt these strategies:

- User-Centric Design: Our user interface using flask will prioritize user-friendliness, making it easier for data architects to interact with the data warehouse.
- Query Building: We will provide user-friendly query-building capabilities to empower data architects to create custom queries.
- Data Visualization: Implementing data visualization components, such as interactive charts and graphs, will enhance the data exploration experience.

- **Actionable Insights**

Our approach to delivering actionable insights will be grounded in reality:

- Analysis Templates: We will create predefined analysis templates for common use cases to expedite insights generation.

TOOLS AND TECHNOLOGIES PLANNED TO USE

Data Modelling

Creating a data model that represents the structure of our data warehouse by involving tables, relationships, and attributes.

We are going to use the IBM Data Architect tool which can provide assistance in data modeling.

Setting Up IBM Cloud Db2 Warehouse

Configuring our database instance, including setting up security, scalability, and performance options.

Data Loading

Loading the transformed data into your **IBM Db2** Warehouse instance. We will use IBM DataStage for this purpose.

Ensuring data quality and consistency during the loading process

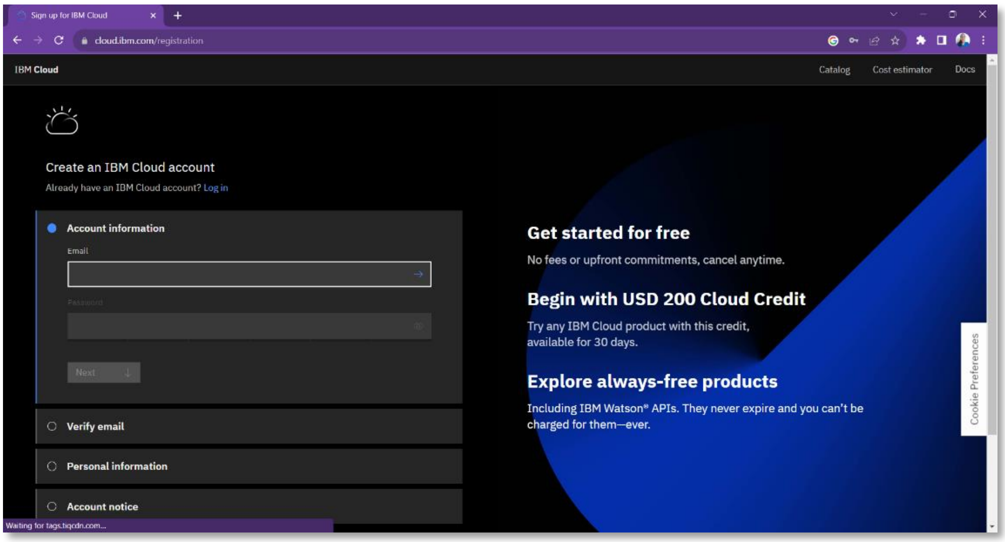
Data Access and Analysis

Analysis using **IBM Watson Studio** tool. By building dashboards, reports, and data analysis pipelines to empower your data architects and analysts.

Utilize the capabilities of Db2 Warehouse, such as its in-database analytics and machine learning features, to perform advanced analyses.

TECHNICAL IMPLEMENTATION

➢ **Step 1:** Creating IBM cloud account



Sign up for IBM Cloud

cloud.ibm.com/registration

IBM Cloud | Catalog | Cost estimator | Docs

Create an IBM Cloud account

Already have an IBM Cloud account? Log in

● Account information

Email

Password

Next

○ Verify email

○ Personal information

○ Account notice

Get started for free

No fees or upfront commitments, cancel anytime.

Begin with USD 200 Cloud Credit

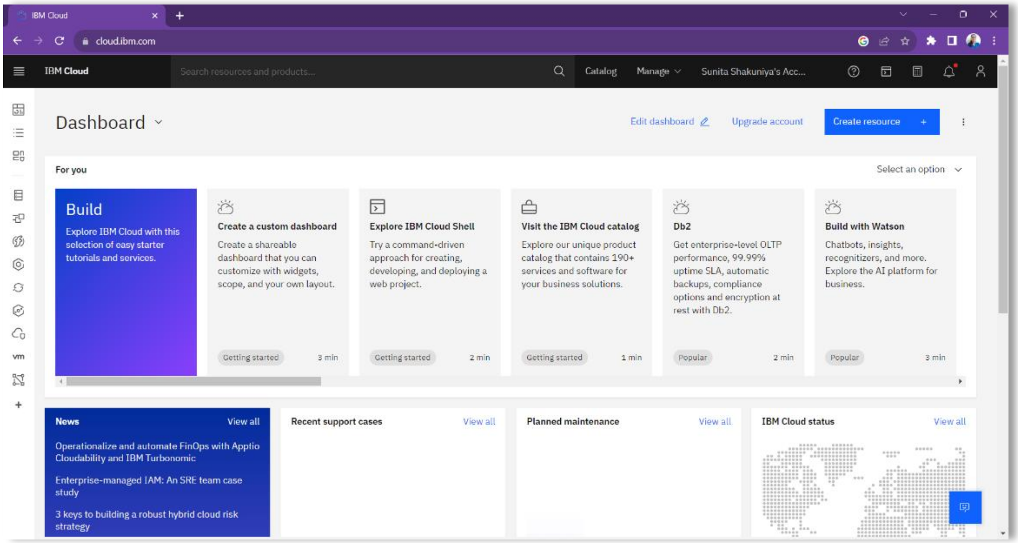
Try any IBM Cloud product with this credit, available for 30 days.

Explore always-free products

Including IBM Watson® APIs. They never expire and you can't be charged for them—ever.

Cookie Preferences

Waiting for tags.tiqcdn.com...



IBM Cloud

cloud.ibm.com

IBM Cloud | Search resources and products... | Catalog | Manage | Sunita Shakuniya's Acc...

Dashboard

Edit dashboard | Upgrade account | Create resource +

For you

Select an option

Build

Explore IBM Cloud with this selection of easy starter tutorials and services.

Create a custom dashboard

Create a shareable dashboard that you can customize with widgets, scope, and your own layout.

Getting started | 3 min

Explore IBM Cloud Shell

Try a command-driven approach for creating, developing, and deploying a web project.

Getting started | 2 min

Visit the IBM Cloud catalog

Explore our unique product catalog that contains 190+ services and software for your business solutions.

Getting started | 1 min

Db2

Get enterprise-level OLTP performance, 99.99% uptime SLA, automatic backups, compliance options and encryption at rest with Db2.

Popular | 2 min

Build with Watson

Chatbots, insights, recognitizers, and more. Explore the AI platform for business.

Popular | 3 min

News | View all

Operationalize and automate FinOps with Apptio Cloudability and IBM Turbonomic

Enterprise-managed IAM: An SRE team case study

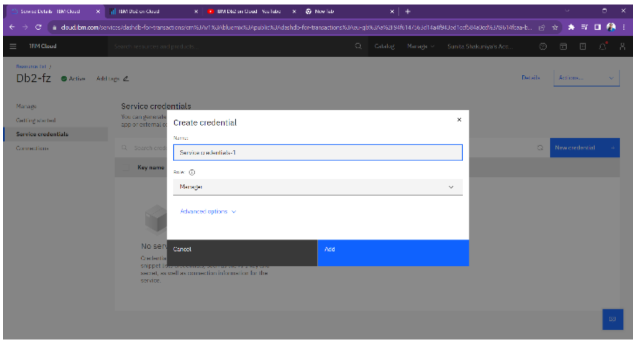
3 keys to building a robust hybrid cloud risk strategy

Recent support cases | View all

Planned maintenance | View all

IBM Cloud status | View all

➢ **Step 2:** Setting up the db2 instance and experimenting.



Service Details - IBM Cloud | IBM Db2 on Cloud | IBM Db2 on Cloud - YouTube | New tab

IBM Cloud | Search resources and products... | Catalog | Manage | Sunita Shakuniya's Acc...

Resource list /

Db2-fz ● Active | Add tags

Details | Actions...

Manage

Getting started

Service credentials

Connections

Service credentials

Create credential

Name:

Service credentials-1

Role: ⓘ

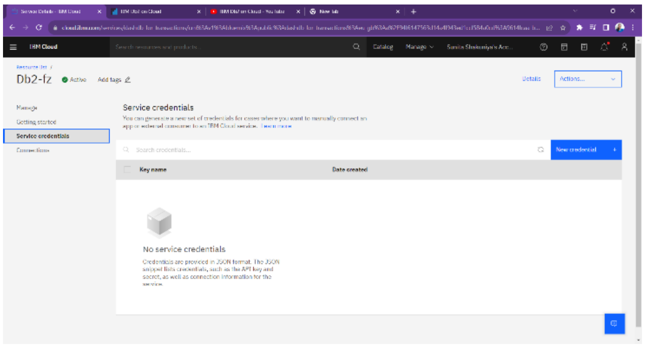
Manager

Advanced options

Cancel | Add

New credential +

Key name



Service Details - IBM Cloud | IBM Db2 on Cloud | IBM Db2 on Cloud - YouTube | New tab

IBM Cloud | Search resources and products... | Catalog | Manage | Sunita Shakuniya's Acc...

Resource list /

Db2-fz ● Active | Add tags

Details | Actions...

Manage

Getting started

Service credentials

Connections

Service credentials

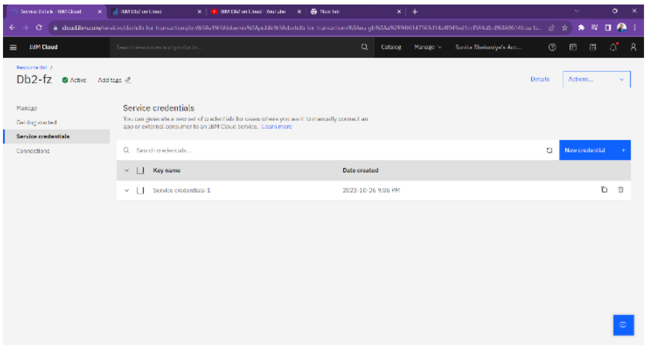
You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. Learn more

Search credentials... | New credential +

Key name | Date created

No service credentials

Credentials are provided in JSON format. The JSON snippet lists credentials, such as the API key and secret, as well as connection information for the service.

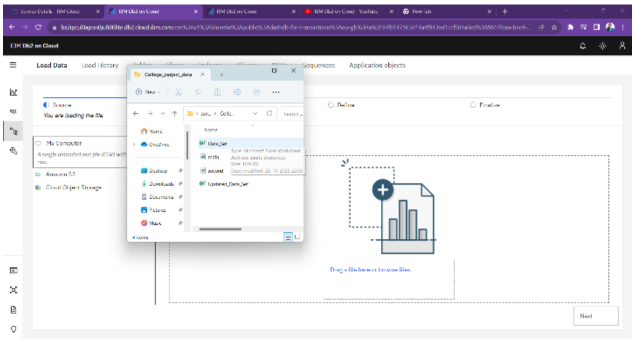


Db2-fz ● Active Add tags

Service credentials

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

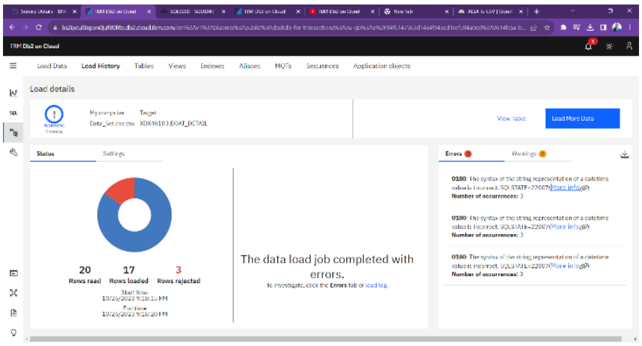
Key name	Date created
Service credentials-1	2023-10-26 9:06 PM



You are loading the file

A single delimited text file (CSV) with row.

Type: Microsoft Excel Worksheet
Authors: sunita shekuniya
Size: 10.9 KB
Date modified: 25-10-2023 22:53



Load details

My computer Target
Data_Set.csv.csv XDX46183.BOAT_DETAIL

20 Rows read 17 Rows loaded 3 Rows rejected

Start time
10/26/2023 9:18:15 PM
End time
10/26/2023 9:18:20 PM

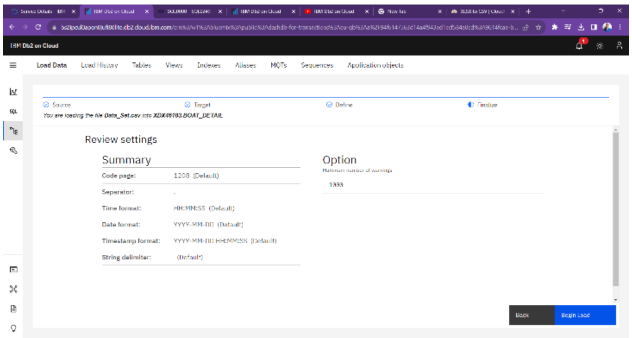
The data load job completed with errors.

To investigate, click the Errors tab or load log.

0180: The syntax of the string representation of a datetime value is incorrect. SQLSTATE=22007(More info)
Number of occurrences: 3

0180: The syntax of the string representation of a datetime value is incorrect. SQLSTATE=22007(More info)
Number of occurrences: 3

0180: The syntax of the string representation of a datetime value is incorrect. SQLSTATE=22007(More info)
Number of occurrences: 3



You are loading the file Data_Set.csv into XDX46183.BOAT_DETAIL

Review settings

Summary

Code page: 1208 (Default)

Separator: ,

Time format: HH:MM:SS (Default)

Date format: YYYY-MM-DD (Default)

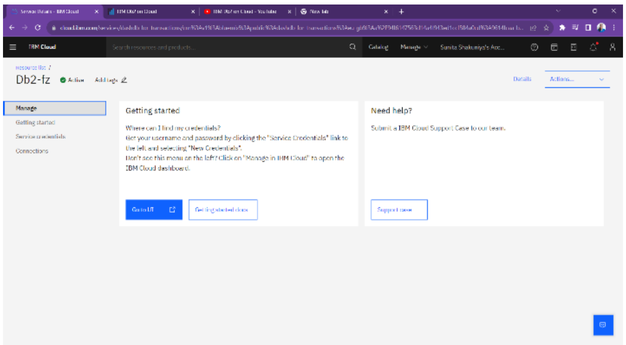
Timestamp format: YYYY-MM-DD HH:MM:SS (Default)

String delimiter: (Default)

Option

Maximum number of warnings

1000



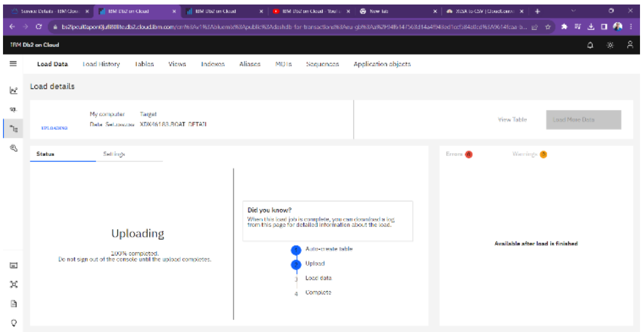
Db2-fz ● Active Add tags

Getting started

Where can I find my credentials?
Get your username and password by clicking the "Service Credentials" link to the left and selecting "New Credentials".
Don't see this menu on the left? Click on "Manage in IBM Cloud" to open the IBM Cloud dashboard.

Need help?

Submit a IBM Cloud Support Case to our team.



Load details

My computer Target
Data_Set.csv.csv XDX46183.BOAT_DETAIL

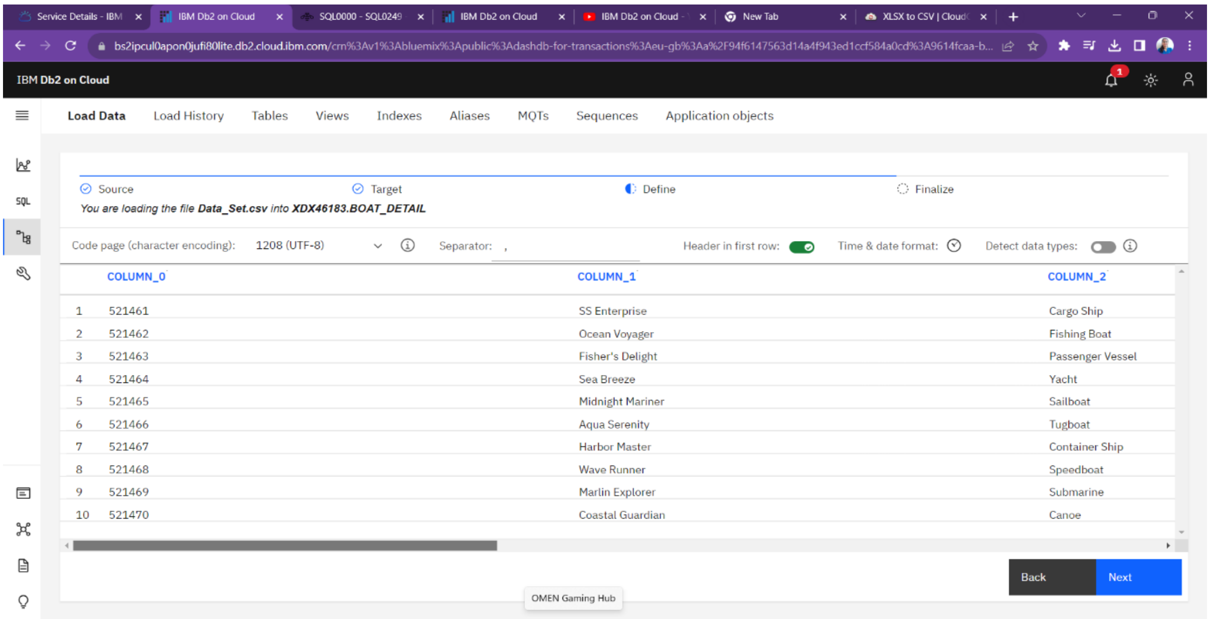
Uploading

100% completed.
Do not sign out of the console until the upload completes.

Did you know?
When this load job is complete, you can download a log from this page for detailed information about the load.

1 Auto-create table
2 Upload
3 Load data
4 Complete

Available after load is finished

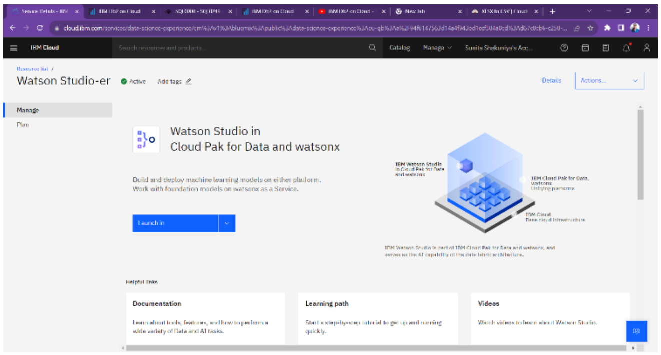


You are loading the file Data_Set.csv into XDX46183.BOAT_DETAIL

Code page (character encoding): 1208 (UTF-8) Separator: , Header in first row: ● Time & date format: Detect data types: ○

	COLUMN_0	COLUMN_1	COLUMN_2
1	521461	SS Enterprise	Cargo Ship
2	521462	Ocean Voyager	Fishing Boat
3	521463	Fisher's Delight	Passenger Vessel
4	521464	Sea Breeze	Yacht
5	521465	Midnight Mariner	Sailboat
6	521466	Aqua Serenity	Tugboat
7	521467	Harbor Master	Container Ship
8	521468	Wave Runner	Speedboat
9	521469	Marlin Explorer	Submarine
10	521470	Coastal Guardian	Canoe

- **Step 3:** Setting up the Watson Studio instance and experimenting.



Resource list /

Watson Studio-er ● Active Add tags

Manage

Plan

Watson Studio in Cloud Pak for Data and watsonx

Build and deploy machine learning models on either platform. Work with foundation models on watsonx as a Service.

Launch in

IBM Watson Studio is part of IBM Cloud Pak for Data and watsonx, and serves as the AI capability of the data fabric architecture.

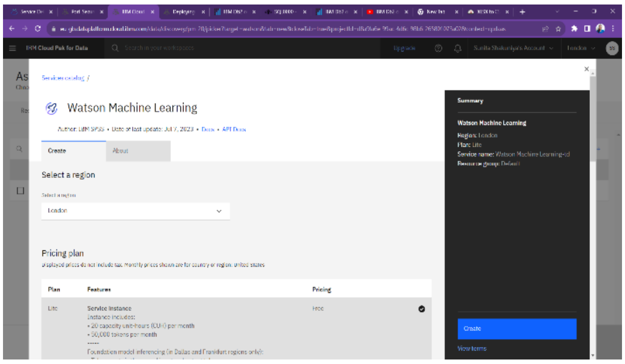
Helpful links

Documentation: Learn about tools, features, and how to perform a wide variety of Data and AI tasks.

Learning path: Start a step-by-step tutorial to get up and running quickly.

Videos: Watch videos to learn about Watson Studio.

IBM Watson Studio in Cloud Pak for Data and watsonx
IBM Cloud Pak for Data, watsonx
Unifying platforms
IBM Cloud
Base cloud infrastructure



Services catalog /

Watson Machine Learning

Author: IBM SPSS • Date of last update: Jul 7, 2023 • Docs • API Docs

Create | About

Select a region

Select a region: London

Pricing plan

Displayed prices do not include tax. Monthly prices shown are for country or region: United States

Plan	Features	Pricing
Lite	Service Instance Instance includes: • 20 capacity unit-hours (CUH) per month • 50,000 tokens per month ----- Foundation model inferencing (in Dallas and Frankfurt regions only):	Free

Summary

Watson Machine Learning

Region: London

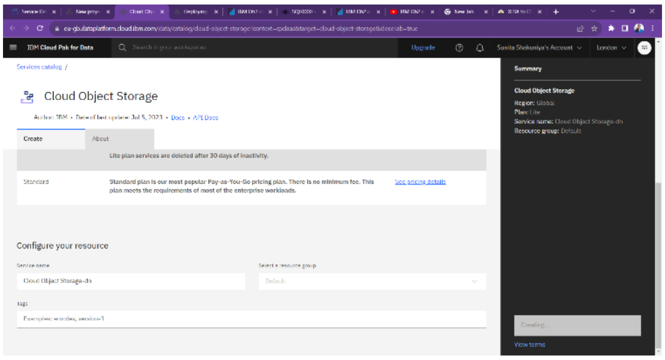
Plan: Lite

Service name: Watson Machine Learning-rd

Resource group: Default

Create

View terms



Services catalog /

Cloud Object Storage

Author: IBM • Date of last update: Jul 5, 2023 • Docs • API Docs

Create | About

Lite plan services are deleted after 30 days of inactivity.

Standard: Standard plan is our most popular Pay-as-You-Go pricing plan. There is no minimum fee. This plan meets the requirements of most of the enterprise workloads. See pricing details

Configure your resource

Service name: Cloud Object Storage-dn

Select a resource group: Default

Tags: Examples: env:dev, version-1

Summary

Cloud Object Storage

Region: Global

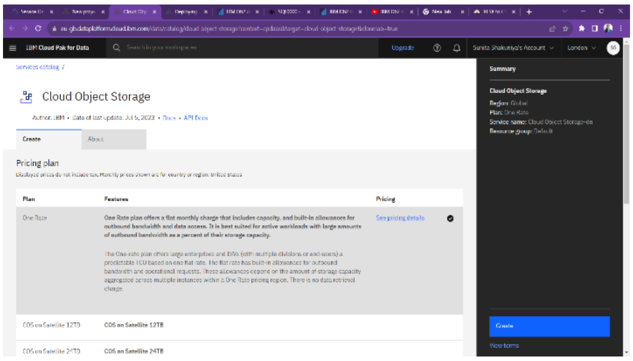
Plan: Lite

Service name: Cloud Object Storage-dn

Resource group: Default

Creating...

View terms



Services catalog /

Cloud Object Storage

Author: IBM • Date of last update: Jul 5, 2023 • Docs • API Docs

Create | About

Pricing plan

Displayed prices do not include tax. Monthly prices shown are for country or region: United States

Plan	Features	Pricing
One Rate	One Rate plan offers a flat monthly charge that includes capacity, and built-in allowances for outbound bandwidth and data access. It is best suited for active workloads with large amounts of outbound bandwidth as a percent of their storage capacity. The One-rate plan offers large enterprises and ISVs (with multiple divisions or end-users) a predictable TCO based on one flat rate. The flat rate has built-in allowances for outbound bandwidth and operational requests. These allowances depend on the amount of storage capacity aggregated across multiple instances within a One Rate pricing region. There is no data retrieval charge.	See pricing details
COS on Satellite 12TB	COS on Satellite 12TB	
COS on Satellite 24TB	COS on Satellite 24TB	

Summary

Cloud Object Storage

Region: Global

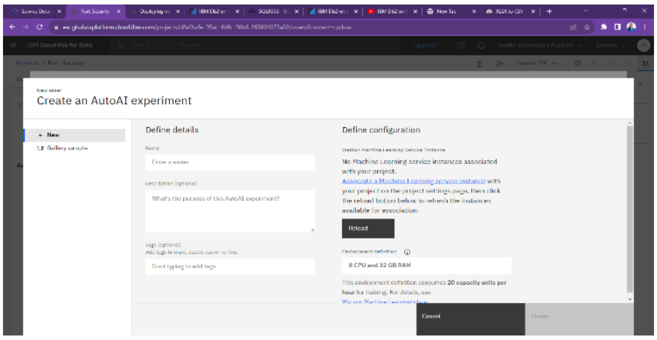
Plan: One Rate

Service name: Cloud Object Storage-dn

Resource group: Default

Create

View terms



Projects / Port_Security

New asset

Create an AutoAI experiment

+ New

Gallery sample

Define details

Name: Enter a name

Description (optional): What's the purpose of this AutoAI experiment?

Tags (optional): Add tags to make assets easier to find. Start typing to add tags

Define configuration

Watson Machine Learning Service Instance

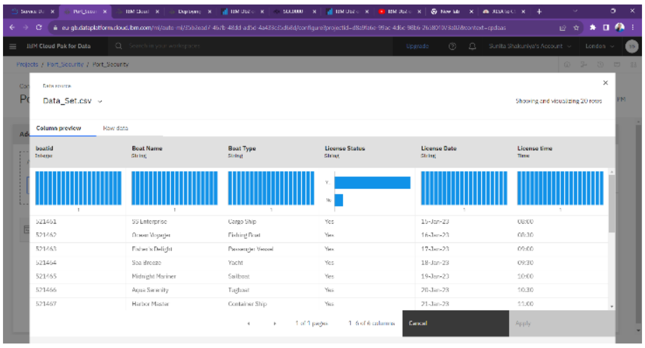
No Machine Learning service instances associated with your project. Associate a Machine Learning service instance with your project on the project settings page, then click the reload button below to refresh the instances available for association.

Reload

Environment definition: 8 CPU and 32 GB RAM

This environment definition consumes 20 capacity units per hour for training. For details, see Watson Machine Learning plans

Cancel | Create



Projects / Port_Security / Port_Security

Data source

Data_Set.csv

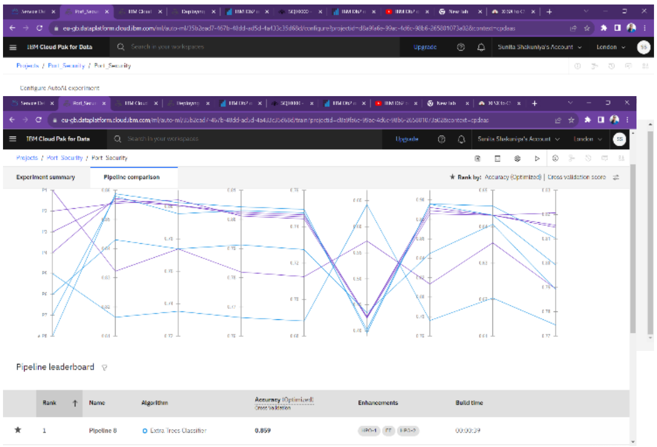
Showing and visualizing 20 rows

Column preview | Raw data

boatid Integer	Boat Name String	Boat Type String	License Status String	License Date String	License time Time
521461	SS Enterprise	Cargo Ship	Yes	15-Jan-23	08:00
521462	Ocean Voyager	Fishing Boat	Yes	16-Jan-23	08:30
521463	Fisher's Delight	Passenger Vessel	Yes	17-Jan-23	09:00
521464	Sea Breeze	Yacht	Yes	18-Jan-23	09:30
521465	Midnight Mariner	Sailboat	Yes	19-Jan-23	10:00
521466	Aqua Serenity	Tugboat	Yes	20-Jan-23	10:30
521467	Harbor Master	Container Ship	Yes	21-Jan-23	11:00

1 of 1 pages | 1–6 of 6 columns

Cancel | Apply



Projects / Port_Security / Port_Security

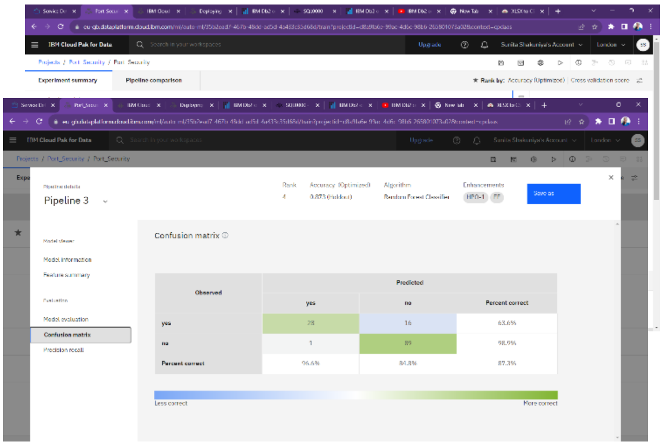
Configure AutoAI experiment

Experiment summary | Pipeline comparison

★ Rank by: Accuracy (Optimized) | Cross validation score

Pipeline leaderboard

	Rank	Name	Algorithm	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 8	Extra Trees Classifier	0.859	HPO-1 FE HPO-2	00:00:39



Projects / Port_Security / Port_Security

Experiment summary | Pipeline comparison

★ Rank by: Accuracy (Optimized) | Cross validation score

Pipeline details

Pipeline 3

Rank: 4

Accuracy (Optimized): 0.873 (Holdout)

Algorithm: Random Forest Classifier

Enhancements: HPO-1 FE

Save as

Model viewer: Model information, Feature summary

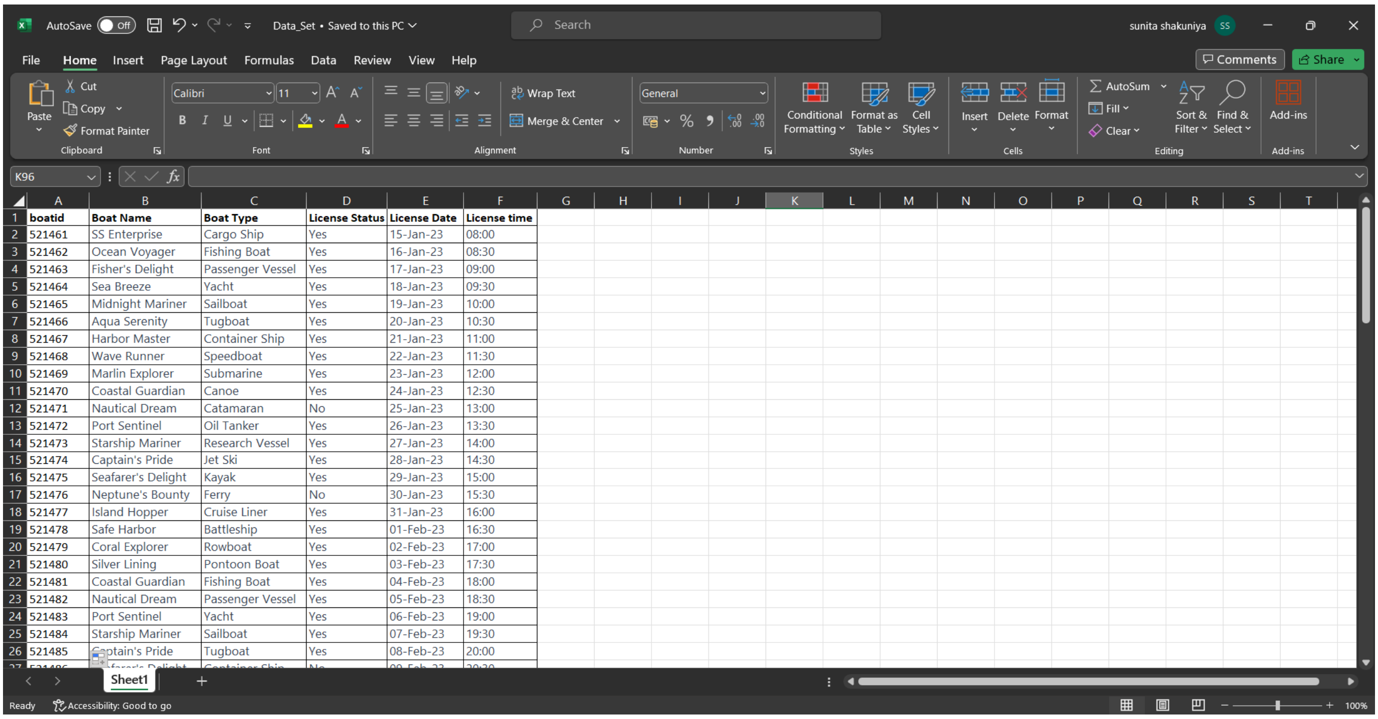
Evaluation: Model evaluation, Confusion matrix, Precision recall

Confusion matrix

Observed	Predicted yes	Predicted no	Percent correct
yes	28	16	63.6%
no	1	89	98.9%
Percent correct	96.6%	84.8%	87.3%

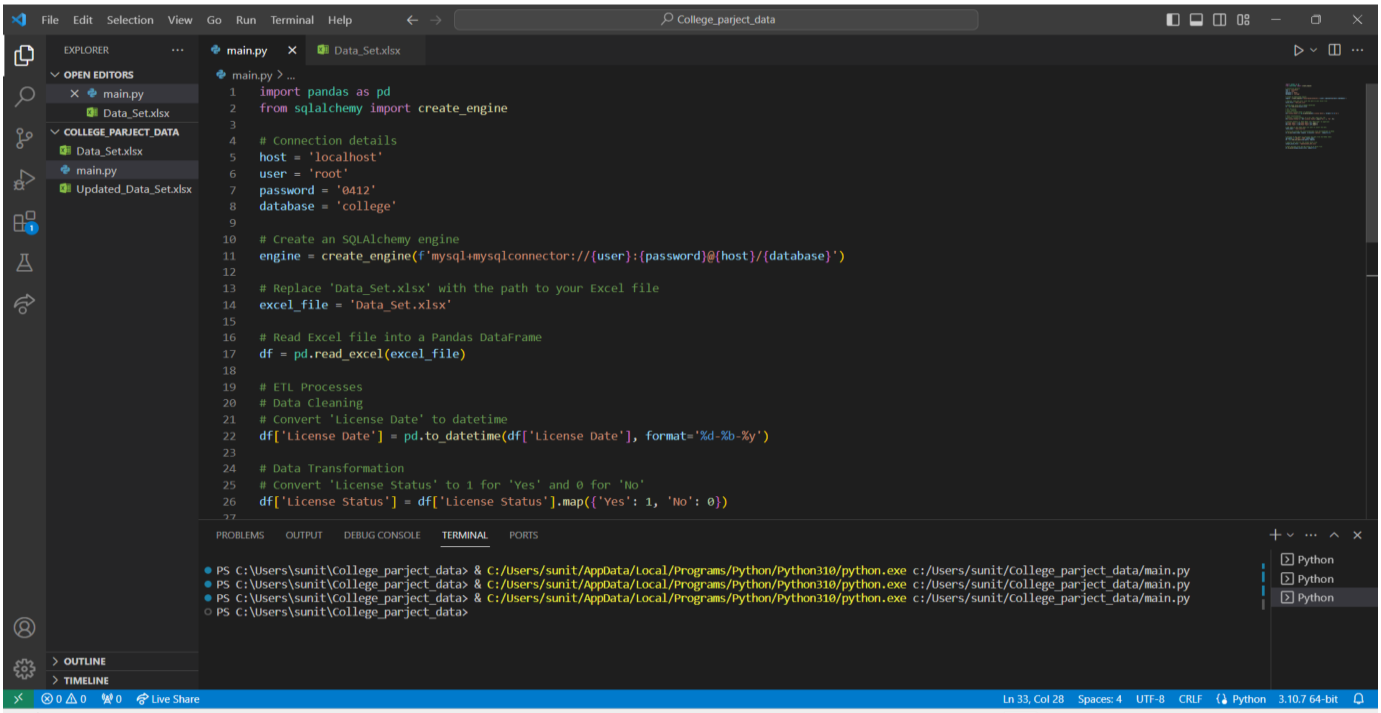
➢ **Step 4:** As we can perform ETL process over Watson studio so we further proceed with panadas library and MySql data base.

- Creating Excel data set



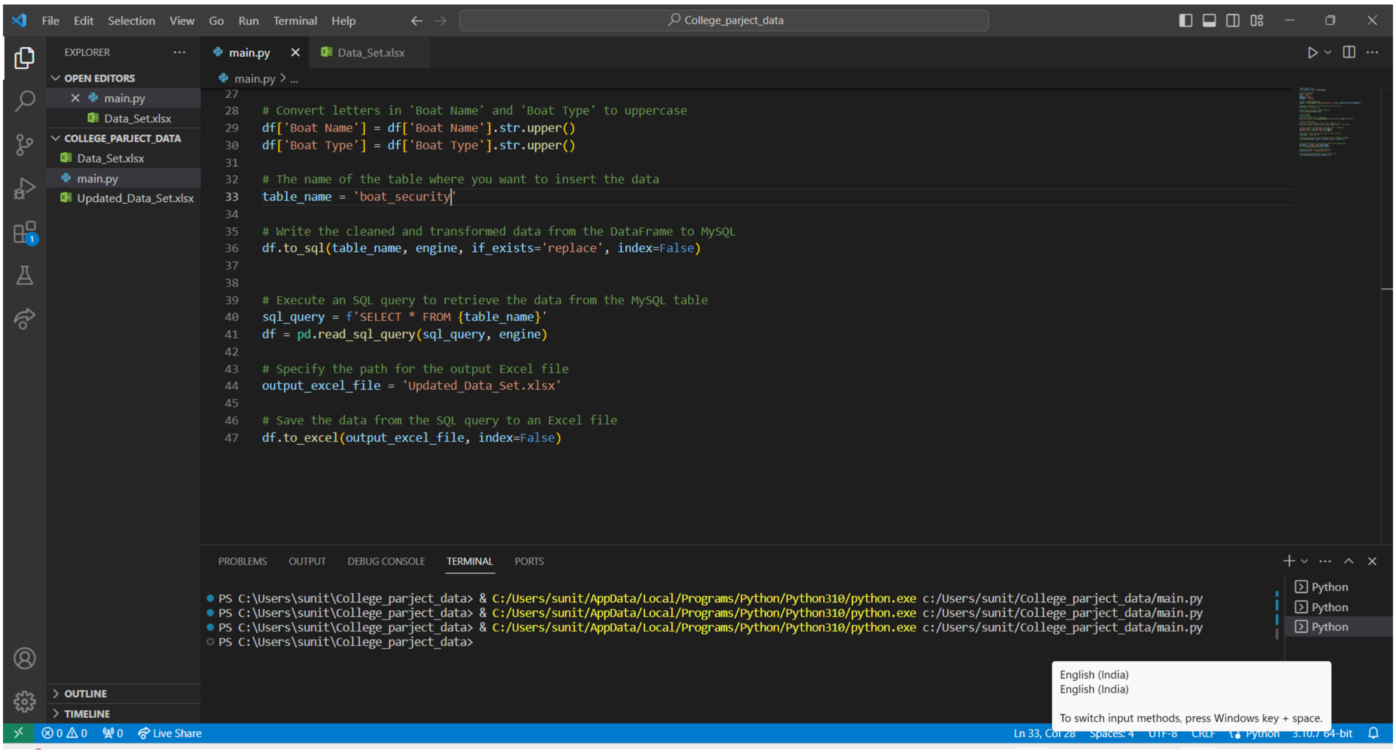
boatid	Boat Name	Boat Type	License Status	License Date	License time
521461	SS Enterprise	Cargo Ship	Yes	15-Jan-23	08:00
521462	Ocean Voyager	Fishing Boat	Yes	16-Jan-23	08:30
521463	Fisher's Delight	Passenger Vessel	Yes	17-Jan-23	09:00
521464	Sea Breeze	Yacht	Yes	18-Jan-23	09:30
521465	Midnight Mariner	Sailboat	Yes	19-Jan-23	10:00
521466	Aqua Serenity	Tugboat	Yes	20-Jan-23	10:30
521467	Harbor Master	Container Ship	Yes	21-Jan-23	11:00
521468	Wave Runner	Speedboat	Yes	22-Jan-23	11:30
521469	Marlin Explorer	Submarine	Yes	23-Jan-23	12:00
521470	Coastal Guardian	Canoe	Yes	24-Jan-23	12:30
521471	Nautical Dream	Catamaran	No	25-Jan-23	13:00
521472	Port Sentinel	Oil Tanker	Yes	26-Jan-23	13:30
521473	Starship Mariner	Research Vessel	Yes	27-Jan-23	14:00
521474	Captain's Pride	Jet Ski	Yes	28-Jan-23	14:30
521475	Seafarer's Delight	Kayak	Yes	29-Jan-23	15:00
521476	Neptune's Bounty	Ferry	No	30-Jan-23	15:30
521477	Island Hopper	Cruise Liner	Yes	31-Jan-23	16:00
521478	Safe Harbor	Battleship	Yes	01-Feb-23	16:30
521479	Coral Explorer	Rowboat	Yes	02-Feb-23	17:00
521480	Silver Lining	Pontoon Boat	Yes	03-Feb-23	17:30
521481	Coastal Guardian	Fishing Boat	Yes	04-Feb-23	18:00
521482	Nautical Dream	Passenger Vessel	Yes	05-Feb-23	18:30
521483	Port Sentinel	Yacht	Yes	06-Feb-23	19:00
521484	Starship Mariner	Sailboat	Yes	07-Feb-23	19:30
521485	Captain's Pride	Tugboat	Yes	08-Feb-23	20:00

- Coding implementation over Vscode



```python
import pandas as pd
from sqlalchemy import create_engine

# Connection details
host = 'localhost'
user = 'root'
password = '0412'
database = 'college'

# Create an SQLAlchemy engine
engine = create_engine(f'mysql+mysqlconnector://{user}:{password}@{host}/{database}')

# Replace 'Data_Set.xlsx' with the path to your Excel file
excel_file = 'Data_Set.xlsx'

# Read Excel file into a Pandas DataFrame
df = pd.read_excel(excel_file)

# ETL Processes
# Data Cleaning
# Convert 'License Date' to datetime
df['License Date'] = pd.to_datetime(df['License Date'], format='%d-%b-%y')

# Data Transformation
# Convert 'License Status' to 1 for 'Yes' and 0 for 'No'
df['License Status'] = df['License Status'].map({'Yes': 1, 'No': 0})
```

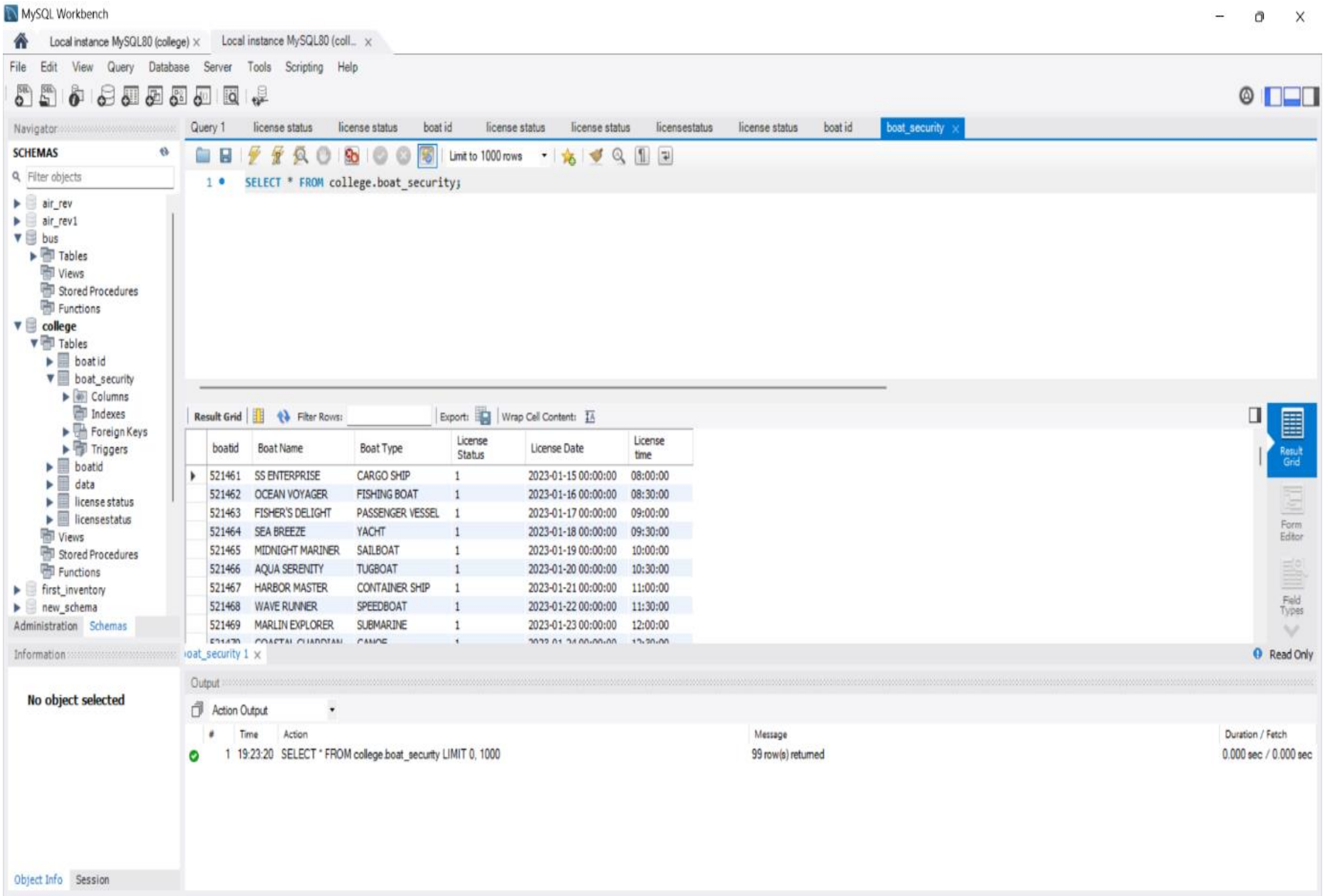
```
PS C:\Users\sunit\College_parject_data> & C:/Users/sunit/AppData/Local/Programs/Python/Python310/python.exe c:/Users/sunit/College_parject_data/main.py
PS C:\Users\sunit\College_parject_data> & C:/Users/sunit/AppData/Local/Programs/Python/Python310/python.exe c:/Users/sunit/College_parject_data/main.py
PS C:\Users\sunit\College_parject_data> & C:/Users/sunit/AppData/Local/Programs/Python/Python310/python.exe c:/Users/sunit/College_parject_data/main.py
PS C:\Users\sunit\College_parject_data>
```



```python
# Convert letters in 'Boat Name' and 'Boat Type' to uppercase
df['Boat Name'] = df['Boat Name'].str.upper()
df['Boat Type'] = df['Boat Type'].str.upper()

# The name of the table where you want to insert the data
table_name = 'boat_security'

# Write the cleaned and transformed data from the DataFrame to MySQL
df.to_sql(table_name, engine, if_exists='replace', index=False)


# Execute an SQL query to retrieve the data from the MySQL table
sql_query = f'SELECT * FROM {table_name}'
df = pd.read_sql_query(sql_query, engine)

# Specify the path for the output Excel file
output_excel_file = 'Updated_Data_Set.xlsx'

# Save the data from the SQL query to an Excel file
df.to_excel(output_excel_file, index=False)
```

```
PS C:\Users\sunit\College_parject_data> & C:/Users/sunit/AppData/Local/Programs/Python/Python310/python.exe c:/Users/sunit/College_parject_data/main.py
PS C:\Users\sunit\College_parject_data> & C:/Users/sunit/AppData/Local/Programs/Python/Python310/python.exe c:/Users/sunit/College_parject_data/main.py
PS C:\Users\sunit\College_parject_data> & C:/Users/sunit/AppData/Local/Programs/Python/Python310/python.exe c:/Users/sunit/College_parject_data/main.py
PS C:\Users\sunit\College_parject_data>
```

- MySQL Workbench

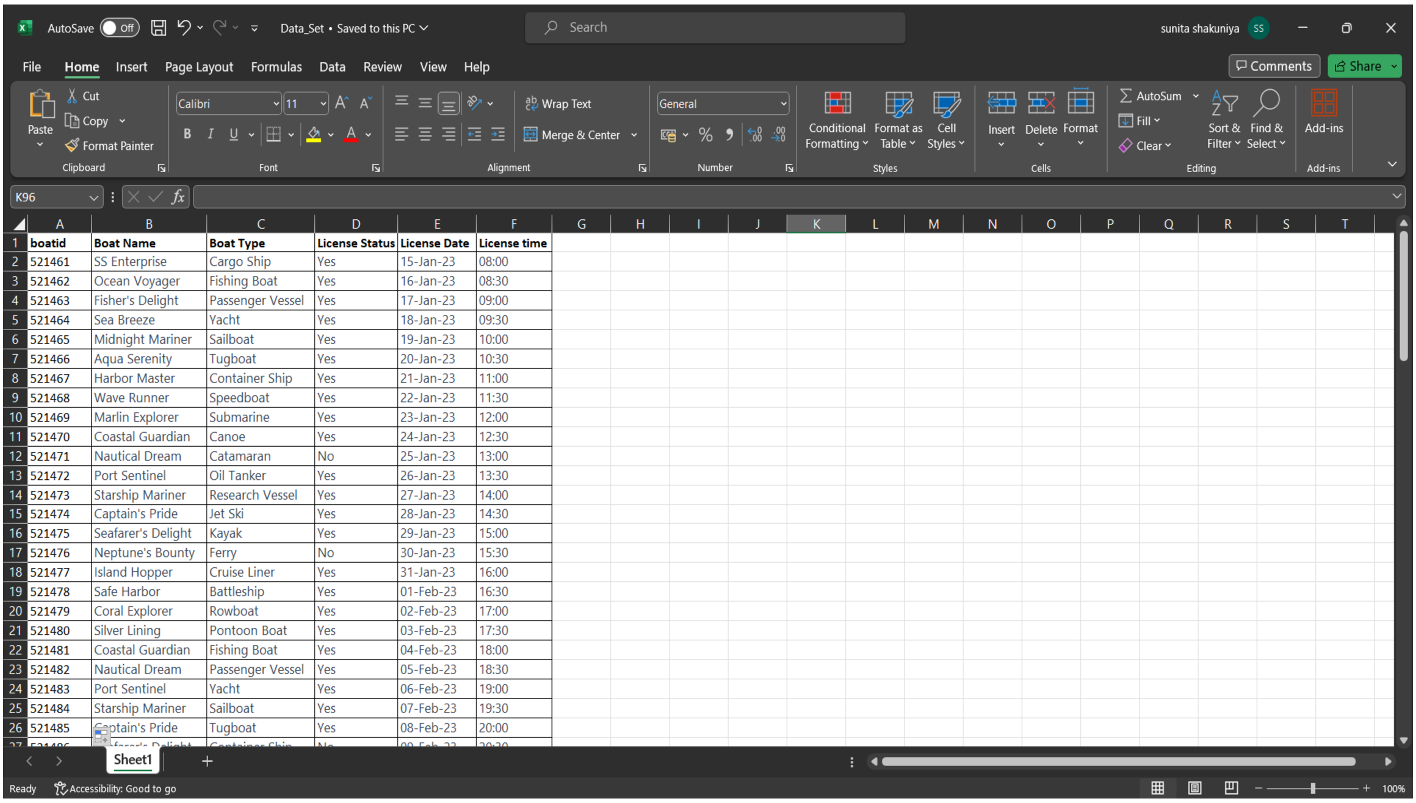


```sql
SELECT * FROM college.boat_security;
```

	boatid	Boat Name	Boat Type	License Status	License Date	License time
▶	521461	SS ENTERPRISE	CARGO SHIP	1	2023-01-15 00:00:00	08:00:00
	521462	OCEAN VOYAGER	FISHING BOAT	1	2023-01-16 00:00:00	08:30:00
	521463	FISHER'S DELIGHT	PASSENGER VESSEL	1	2023-01-17 00:00:00	09:00:00
	521464	SEA BREEZE	YACHT	1	2023-01-18 00:00:00	09:30:00
	521465	MIDNIGHT MARINER	SAILBOAT	1	2023-01-19 00:00:00	10:00:00
	521466	AQUA SERENITY	TUGBOAT	1	2023-01-20 00:00:00	10:30:00
	521467	HARBOR MASTER	CONTAINER SHIP	1	2023-01-21 00:00:00	11:00:00
	521468	WAVE RUNNER	SPEEDBOAT	1	2023-01-22 00:00:00	11:30:00
	521469	MARLIN EXPLORER	SUBMARINE	1	2023-01-23 00:00:00	12:00:00
	521470	COASTAL GUARDIAN	CANOE	1	2023-01-24 00:00:00	12:30:00

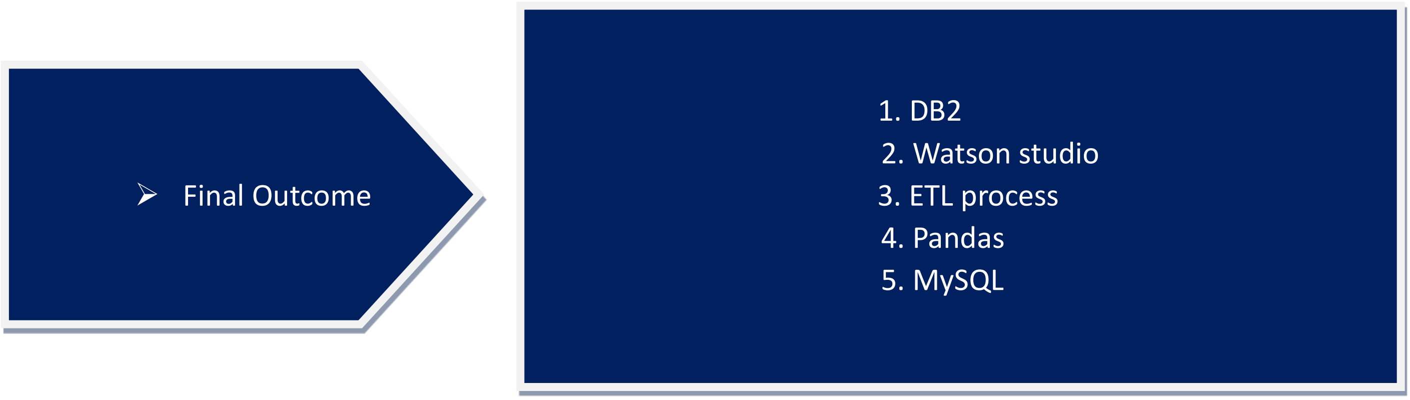
	#	Time	Action	Message	Duration / Fetch
✓	1	19:23:20	SELECT * FROM college.boat_security LIMIT 0, 1000	99 row(s) returned	0.000 sec / 0.000 sec

- Updated excel sheet



	A	B	C	D	E	F
1	boatid	Boat Name	Boat Type	License Status	License Date	License time
2	521461	SS Enterprise	Cargo Ship	Yes	15-Jan-23	08:00
3	521462	Ocean Voyager	Fishing Boat	Yes	16-Jan-23	08:30
4	521463	Fisher's Delight	Passenger Vessel	Yes	17-Jan-23	09:00
5	521464	Sea Breeze	Yacht	Yes	18-Jan-23	09:30
6	521465	Midnight Mariner	Sailboat	Yes	19-Jan-23	10:00
7	521466	Aqua Serenity	Tugboat	Yes	20-Jan-23	10:30
8	521467	Harbor Master	Container Ship	Yes	21-Jan-23	11:00
9	521468	Wave Runner	Speedboat	Yes	22-Jan-23	11:30
10	521469	Marlin Explorer	Submarine	Yes	23-Jan-23	12:00
11	521470	Coastal Guardian	Canoe	Yes	24-Jan-23	12:30
12	521471	Nautical Dream	Catamaran	No	25-Jan-23	13:00
13	521472	Port Sentinel	Oil Tanker	Yes	26-Jan-23	13:30
14	521473	Starship Mariner	Research Vessel	Yes	27-Jan-23	14:00
15	521474	Captain's Pride	Jet Ski	Yes	28-Jan-23	14:30
16	521475	Seafarer's Delight	Kayak	Yes	29-Jan-23	15:00
17	521476	Neptune's Bounty	Ferry	No	30-Jan-23	15:30
18	521477	Island Hopper	Cruise Liner	Yes	31-Jan-23	16:00
19	521478	Safe Harbor	Battleship	Yes	01-Feb-23	16:30
20	521479	Coral Explorer	Rowboat	Yes	02-Feb-23	17:00
21	521480	Silver Lining	Pontoon Boat	Yes	03-Feb-23	17:30
22	521481	Coastal Guardian	Fishing Boat	Yes	04-Feb-23	18:00
23	521482	Nautical Dream	Passenger Vessel	Yes	05-Feb-23	18:30
24	521483	Port Sentinel	Yacht	Yes	06-Feb-23	19:00
25	521484	Starship Mariner	Sailboat	Yes	07-Feb-23	19:30
26	521485	Captain's Pride	Tugboat	Yes	08-Feb-23	20:00
27	521486	Seafarer's Delight	Container Ship	No	09-Feb-23	20:30

- **Step 5:** Conclusion, we finally performed the ETL process using the pandas' library over vscode and Excel data and after running the code for applying ETL queries we got the updated data back in the same format.



- Final Outcome

1. DB2
2. Watson studio
3. ETL process
4. Pandas
5. MySQL