

Publish data to IBM cloud

Team ID	PNT2022TMID02419
Project Name	Real-Time River Water Quality Monitoring and Control System

You can publish your virtual data to catalogs in Watson™ Knowledge Catalog. An administrator can configure Data Virtualization to automatically publish all virtual objects that are created in the user interface to a configured primary catalog.

Data Virtualization / cqwwu-devstable-0621 ▾

Service settings ▾

General

Governance

Scaling settings

Scaling history

Access >

Enforcement
Turning policy and publishing enforcement on or off will impact where your objects can be published to and whether they will be governed. If you choose to enforce publishing, you must select a governed catalog as the primary catalog.

Enforce policies within Data Virtualization ⓘ ☒ On

Enforce publishing to a governed catalog ☒ On

Select a governed catalog as your primary catalog
All virtual objects will be published to the selected catalog and will be governed through the Watson Knowledge Catalog service. You can choose one of the following governed catalogs that you owned.

🔍 Find catalog by Name, Owner, Description

Catalog name	Owner	Date created	Description
<input checked="" type="radio"/> wkc0	IBMid-6610020D12	Jun 11, 2021	
<input type="radio"/> wkc1	IBMid-6610020D12	Jun 11, 2021	

Review cart and virtualize tables

Empty cart

Back

Virtualize

Assign to

☐ Project☐ My virtualized data

Also publish to

☒ Publish to catalog

Select a catalog

wkc0

Governed

DevCatalog

Not governed

TestCatalog

Not governed

wkc1

Governed

Objects to be virtualized

Table	Schema	Source schema	Virtualized tables
CONTEXT	DV_IBM... x v	AUDIT	1 :
SYSADMIN	DV_IBM... x v	AUDIT	1 :
VALIDATE	DV_IBM... x v	AUDIT	1 :

Python output:

```

File Edit Shell Debug Options Window Help
except Published Temperature = 0 C ph = 1 % turbidity = 58 NTU to IBM Watson
# Conn Published Temperature = 27 C ph = 11 % turbidity = 4 NTU to IBM Watson
device Published Temperature = 43 C ph = 14 % turbidity = 33 NTU to IBM Watson
while Published Temperature = 8 C ph = 13 % turbidity = 32 NTU to IBM Watson
Published Temperature = 68 C ph = 3 % turbidity = 77 NTU to IBM Watson
Published Temperature = 80 C ph = 14 % turbidity = 62 NTU to IBM Watson
Published Temperature = 51 C ph = 0 % turbidity = 15 NTU to IBM Watson
Published Temperature = 52 C ph = 5 % turbidity = 86 NTU to IBM Watson
Published Temperature = 87 C ph = 0 % turbidity = 75 NTU to IBM Watson
Published Temperature = 64 C ph = 9 % turbidity = 24 NTU to IBM Watson
Published Temperature = 89 C ph = 13 % turbidity = 11 NTU to IBM Watson
Published Temperature = 42 C ph = 10 % turbidity = 45 NTU to IBM Watson
Published Temperature = 41 C ph = 3 % turbidity = 10 NTU to IBM Watson
Published Temperature = 50 C ph = 11 % turbidity = 46 NTU to IBM Watson
Published Temperature = 100 C ph = 5 % turbidity = 53 NTU to IBM Watson
Published Temperature = 34 C ph = 10 % turbidity = 48 NTU to IBM Watson
Published Temperature = 38 C ph = 11 % turbidity = 1 NTU to IBM Watson
Published Temperature = 68 C ph = 3 % turbidity = 63 NTU to IBM Watson
Published Temperature = 38 C ph = 14 % turbidity = 44 NTU to IBM Watson
Published Temperature = 88 C ph = 2 % turbidity = 66 NTU to IBM Watson
Published Temperature = 9 C ph = 13 % turbidity = 22 NTU to IBM Watson
Published Temperature = 40 C ph = 2 % turbidity = 8 NTU to IBM Watson
Published Temperature = 42 C ph = 10 % turbidity = 19 NTU to IBM Watson
Published Temperature = 12 C ph = 5 % turbidity = 21 NTU to IBM Watson
Published Temperature = 100 C ph = 0 % turbidity = 21 NTU to IBM Watson
Published Temperature = 54 C ph = 2 % turbidity = 8 NTU to IBM Watson
# Disc Published Temperature = 26 C ph = 10 % turbidity = 78 NTU to IBM Watson
device Published Temperature = 41 C ph = 4 % turbidity = 9 NTU to IBM Watson
Published Temperature = 78 C ph = 3 % turbidity = 15 NTU to IBM Watson

```

Python code connected to IBM Watson:

The screenshot displays the IBM Watson IoT Platform dashboard and a Python 3.7.0 Shell window. The dashboard shows a device named 'ESP32' with status 'Connected' and type 'NodeMCU'. Below this, a table lists recent events with columns for Event, Value, and Format.

Event	Value	Format
Alert1	{"alert1": "Temperature(14.88) is high, Intake is p..."}	json
Flowrate sen...	{"Water_rate": 81.53}	json
TOC sensor	{"Orgcarbon_level": 710.638}	json
optod sensor	{"Oxygen_level": 2.9}	json
Turbidity sen...	{"Turbidity_level": 571.03}	json

The Python shell window shows the output of a script, including a successful connection message and a series of published data points to IBM Watson:

```
2022-11-11 15:47:48,477 ibmiotf.device.Client INFO Connected successfully: d:39hari:NodeMCU:ESP32
.....publish ok.....
Published Temperature = 37.74 C to IBM Watson
Published PHLevel = 4.869 to IBM Watson
Published Turbidity_level = 862.47 to IBM Watson
Published Oxygen_level = 1.0 to IBM Watson
Published Orgcarbon_level = 1686.9192 to IBM Watson
Published Water_rate = 350.297 C3 to IBM Watson

Intake to Filtration is Closed
Published alert1 : Temperature(37.74) is high, Intake to plant is closed to IBM Watson

Impurities too high diversion to Industrial use
Published alert3 : Turbidity_level(862.47) the filtration process to IBM Watson

low level of dissolved oxygen in river water
Published alert4 : Oxygen_level(1.0) Freshwater creatures may be dying required help to IBM Watson
alert is send marine department
high amount of organic carbon
Published alert1 : Orgcarbon_level(1686.9192) Authorities alerted prevented loss in lives to IBM Watson

sludge gates are opened
Published alert6 : water_rate(350.297) water rate is high so it indicates rain or reservoir release of water into the stream to IBM Watson

.....publish ok.....
Published Temperature = 14.88 C to IBM Watson
Published PHLevel = 11.67 to IBM Watson
Published Turbidity_level = 571.03 to IBM Watson
Published Oxygen_level = 2.9 to IBM Watson
Published Orgcarbon_level = 710.638 to IBM Watson
Published Water_rate = 81.53 C3 to IBM Watson
```

0 Simulations running