

SPRINT – 3

TEAM ID:PNT2022TMID47006

OUTPUT

The screenshot displays the IBM Watson IoT Platform interface. The main view shows a list of recent events for a device. The events are listed in a table with columns: Event, Value, Format, and Last. The events are generated by a script that sends random data points.

Event	Value	Format	Last
eventbatch11	{"randomNumber":0,"level":47,"weight":947}	json	a fe
eventbatch11	{"randomNumber":37,"level":6,"weight":273}	json	a m
eventbatch11	{"randomNumber":90,"level":35,"weight":605}	json	2 m
eventbatch11	{"randomNumber":54,"level":34,"weight":224}	json	3 m
eventbatch11	{"randomNumber":45,"level":54,"weight":407}	json	4 m

A modal window titled "Device Type: abcd" is open, showing the "Events" tab. It allows creating a new event type. The "Event type name" is "eventbatch11". The "Schedule" is set to "Every Minute". The "Payload" is a JSON object with three fields: "randomNumber", "level", and "weight", each with a random value generator.

```
{
  "randomNumber": random(0, 100)
  "level": random(0, 100)
  "weight": random(0, 1000)
}
```

The modal also includes a "Send" button, a "New event type" button, and a "Cancel" button. There is also an "Upload a CSV file" button.

```
ibmiotpublishsubscribe (1).py - C:\Users\naave\Dropbox\PC\Downloads\ibmiotpublishsubscribe (1).py (3.7.0)
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\naave\Dropbox\PC\Downloads\ibmiotpublishsubscribe (1).py =
2022-11-13 11:52:44,654 ibmiotf.device.Client INFO Connected successfully: d:cbsejl:abc
d:1234
Published level = 82 C weight = 64 % to IBM Watson
Full LED ON
Published level = 5 C weight = 2 % to IBM Watson
Published level = 22 C weight = 57 % to IBM Watson
Published level = 83 C weight = 60 % to IBM Watson
Full LED ON
Published level = 16 C weight = 12 % to IBM Watson
Published level = 19 C weight = 91 % to IBM Watson
Published level = 35 C weight = 77 % to IBM Watson
Published level = 22 C weight = 46 % to IBM Watson
Published level = 85 C weight = 68 % to IBM Watson
Full LED ON
Published level = 36 C weight = 88 % to IBM Watson
Published level = 69 C weight = 72 % to IBM Watson
Published level = 14 C weight = 3 % to IBM Watson
Published level = 99 C weight = 0 % to IBM Watson

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is off")
    #print(cmd)

try:
    deviceOptions = {"org": organization, "type": de
```

