# Code inspired from Confluent Cloud official examples library

# https://github.com/confluentinc/examples/blob/7.1.1-post/clients/cloud/python/producer.py

from confluent\_kafka import Producer

import json

import ccloud\_lib # Library not installed with pip but imported from ccloud\_lib.py

import numpy as np

import time

# Initialize configurations from "python.config" file

CONF = ccloud\_lib.read\_ccloud\_config("python.config")

TOPIC = "my\_first\_topic"

# Create Producer instance

producer\_conf = ccloud\_lib.pop\_schema\_registry\_params\_from\_config(CONF)

producer = Producer(producer\_conf)

# Create topic if it doesn't already exist

ccloud\_lib.create\_topic(CONF, TOPIC)

try:

# Starts an infinite while loop that produces random current temperatures

while True:

record\_key = "weather"

record\_value = json.dumps(

{

"degrees\_in\_celsion": np.random.randint(10, 40)

}

)

print("Producing record: {}\t{}".format(record\_key, record\_value))

# This will actually send data to your topic

producer.produce(

TOPIC,

key=record\_key,

value=record\_value,

)

time.sleep(0.5)

# Interrupt infinite loop when hitting CTRL+C

except KeyboardInterrupt:

pass

finally:

producer.flush() # Finish producing the latest event before stopping the whole script