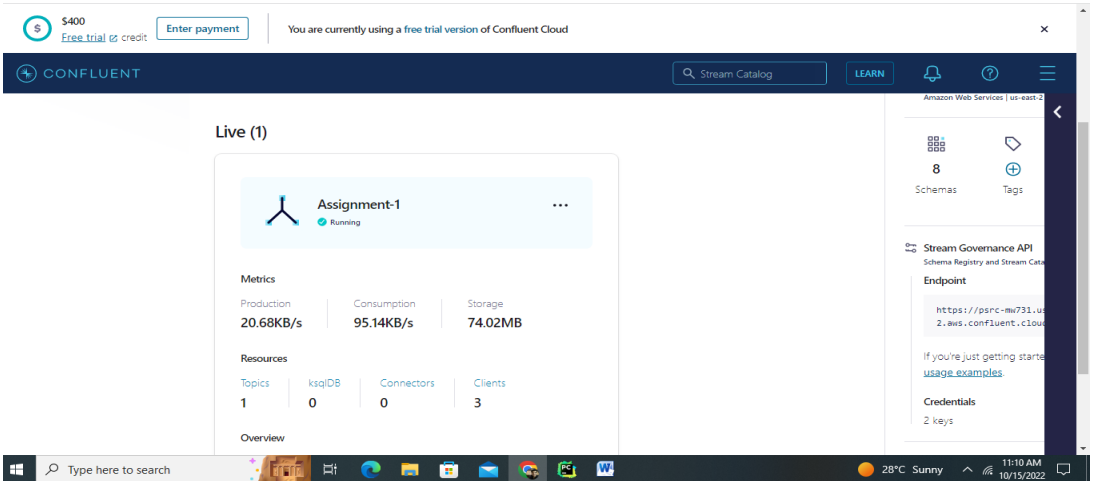
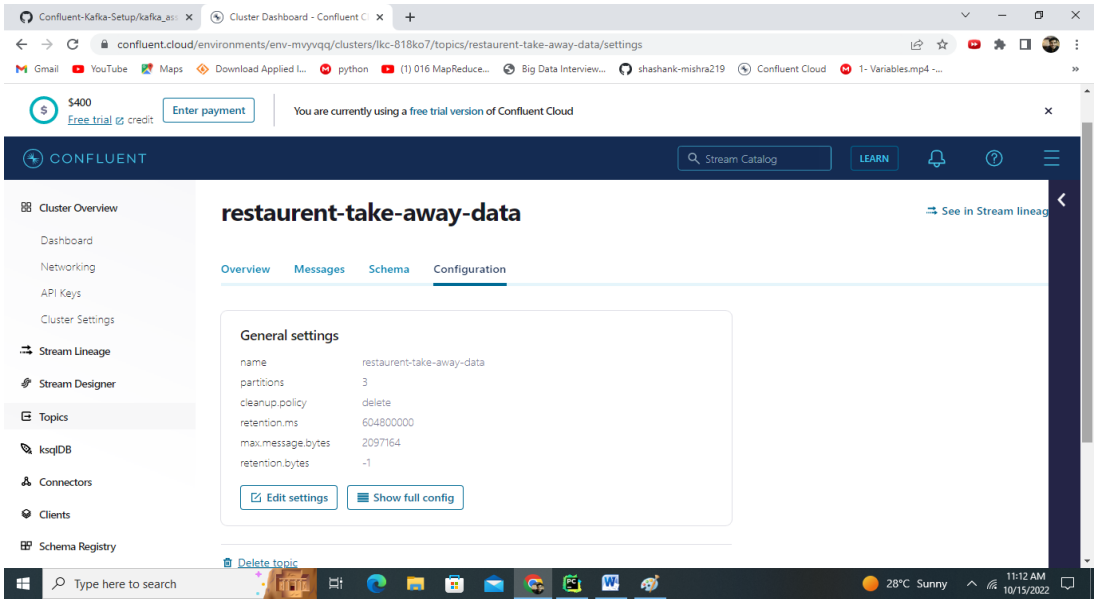
Conflutent Kafka

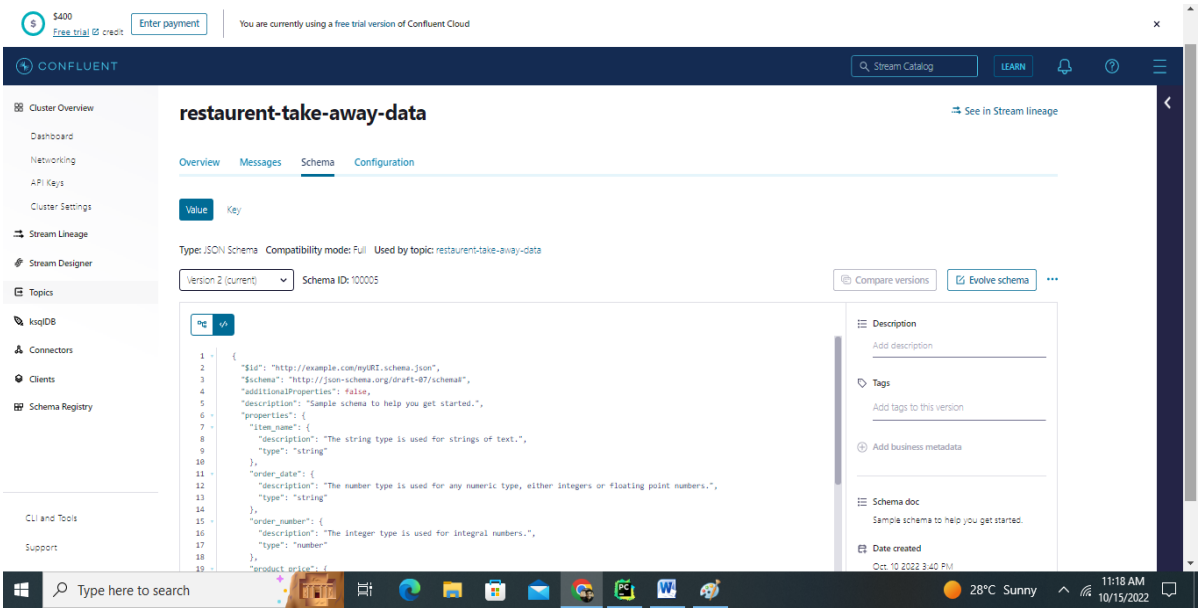
- **Cluster name -> Assignment-1**



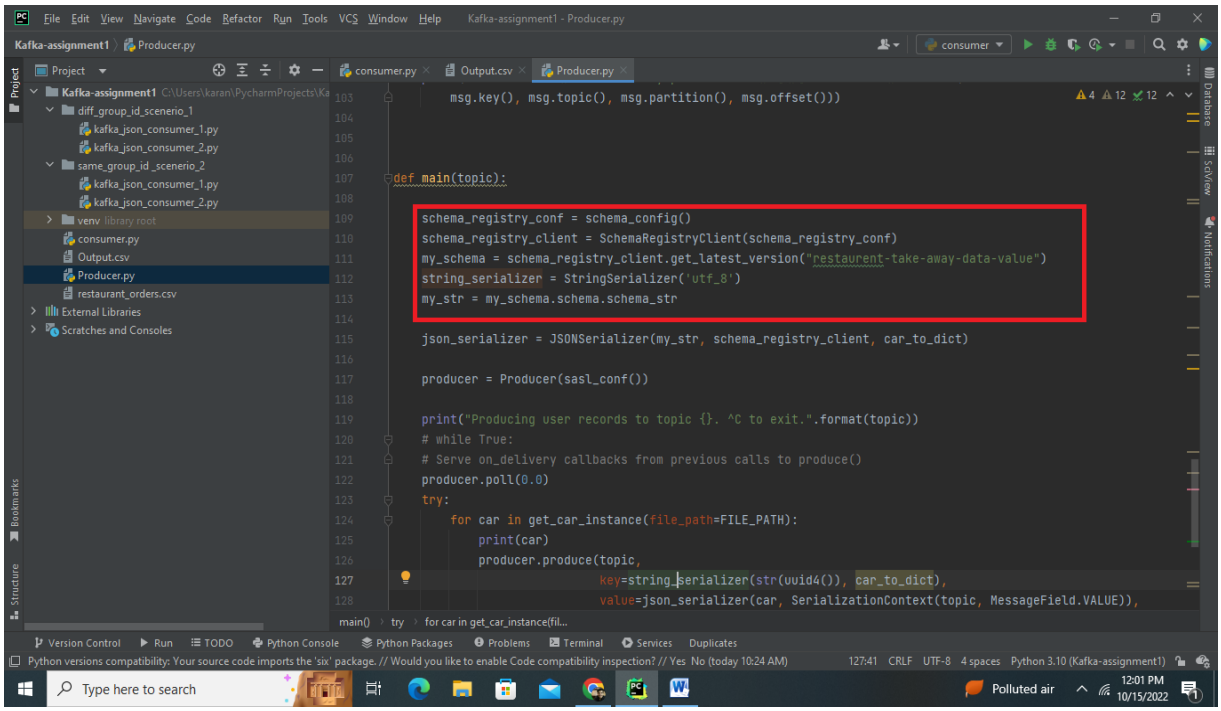
- **Created one kafka topic named as "restaurent-take-away-data" with 3 partitions**



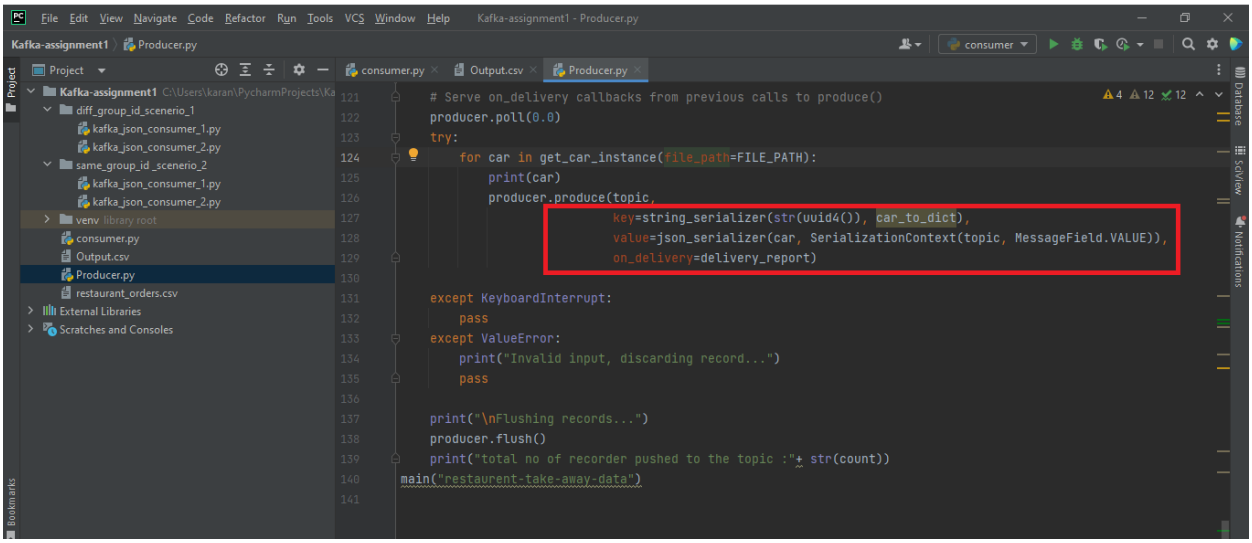
- **Setup key (string) & value (json) schema in the confluent schema registry**



- **Write a kafka producer program (python or any other language) to read data records from restaurent data csv file, make sure schema is not hardcoded in the producer code, read the latest version of schema and schema_str from schema registry and use it for data serialization**



- ***From producer code, publish data in Kafka Topic one by one and use dynamic key while publishing the records into the Kafka Topic***

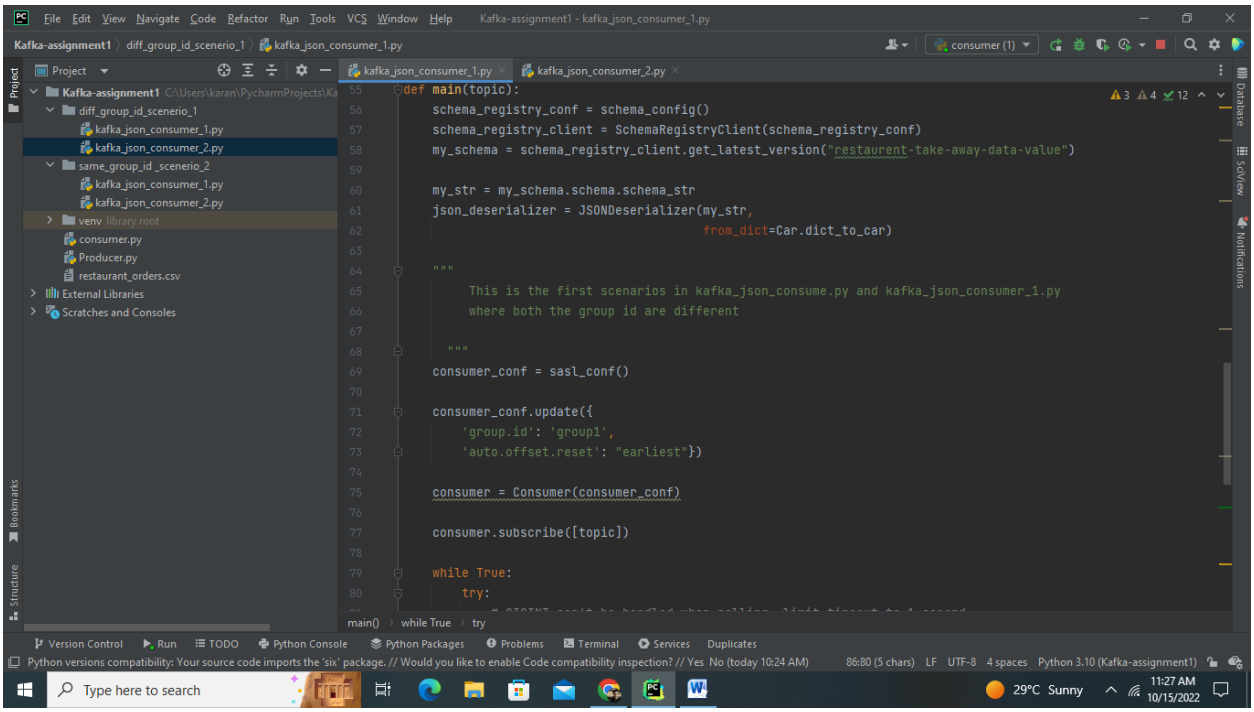


Qus) Write kafka consumer code and create two copies of same consumer code and save it with different names (kafka_consumer_1.py & kafka_consumer_2.py), again make sure lates schema version and schema_str is not hardcoded in the consumer code, read it automatically from the schema registry to desrialize the data. Now test two scenarios with your consumer code:

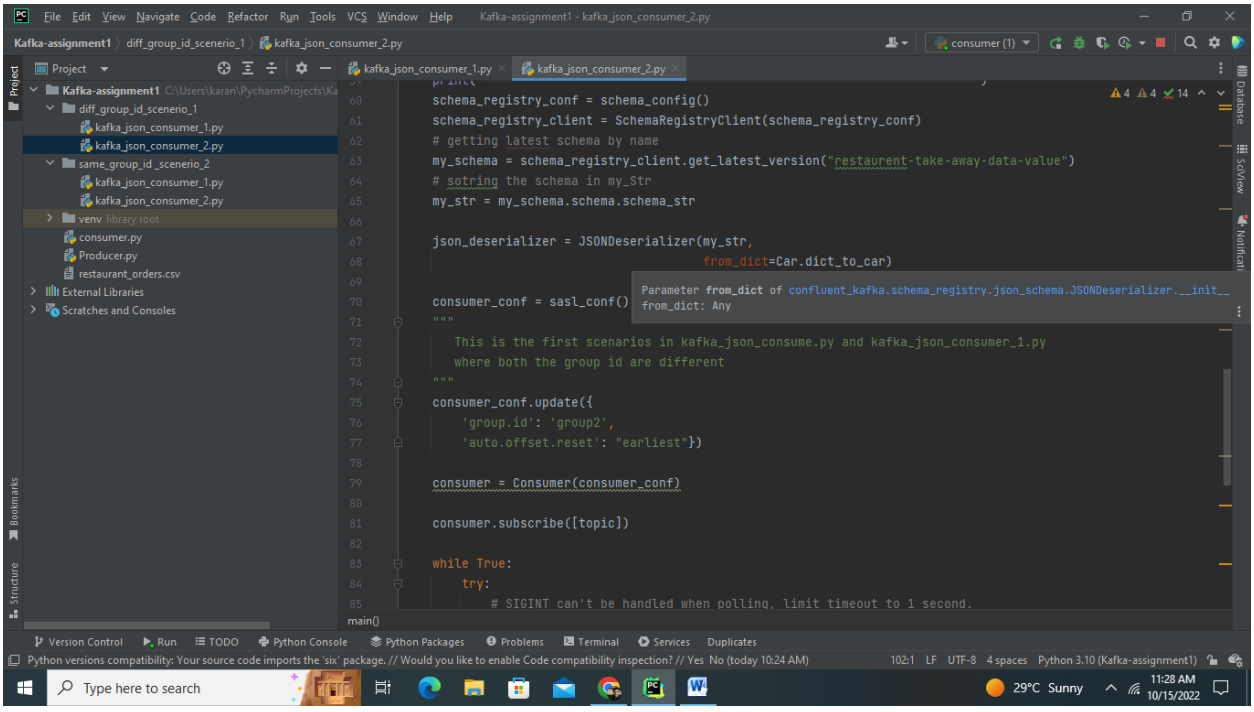
<u>Scenarios with your consumer code:</u>

Qus) Use "group.id" property in consumer config for both consumers and mention different group_ids in kafka_consumer_1.py & kafka_consumer_2.py,apply "earliest" offset property in both consumers and run these two consumers from two different terminals. Calculate how many records each consumer consumed and printed on the terminal

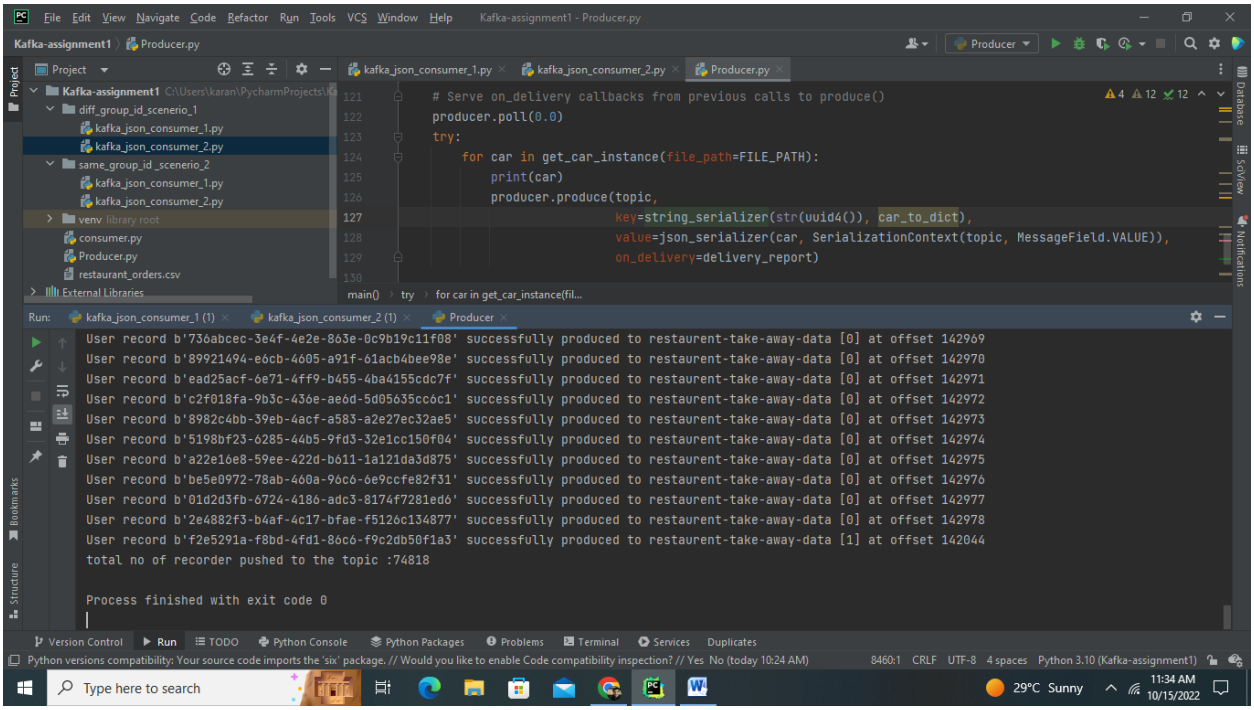
- **Kafka_json_consumer_1.py (group id:group1,auto.offset.reset:earliest)**



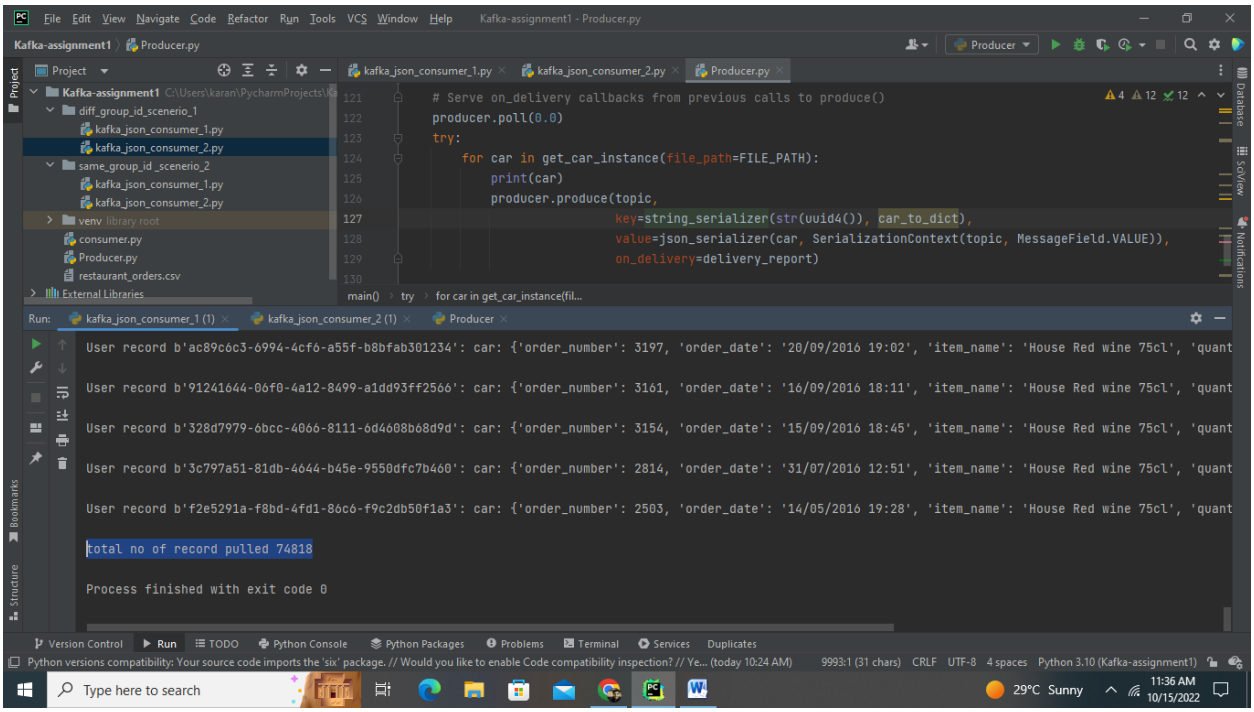
- **Kafka_json_consumer_2.py (group id:group1,auto.offset.reset:earliest)**



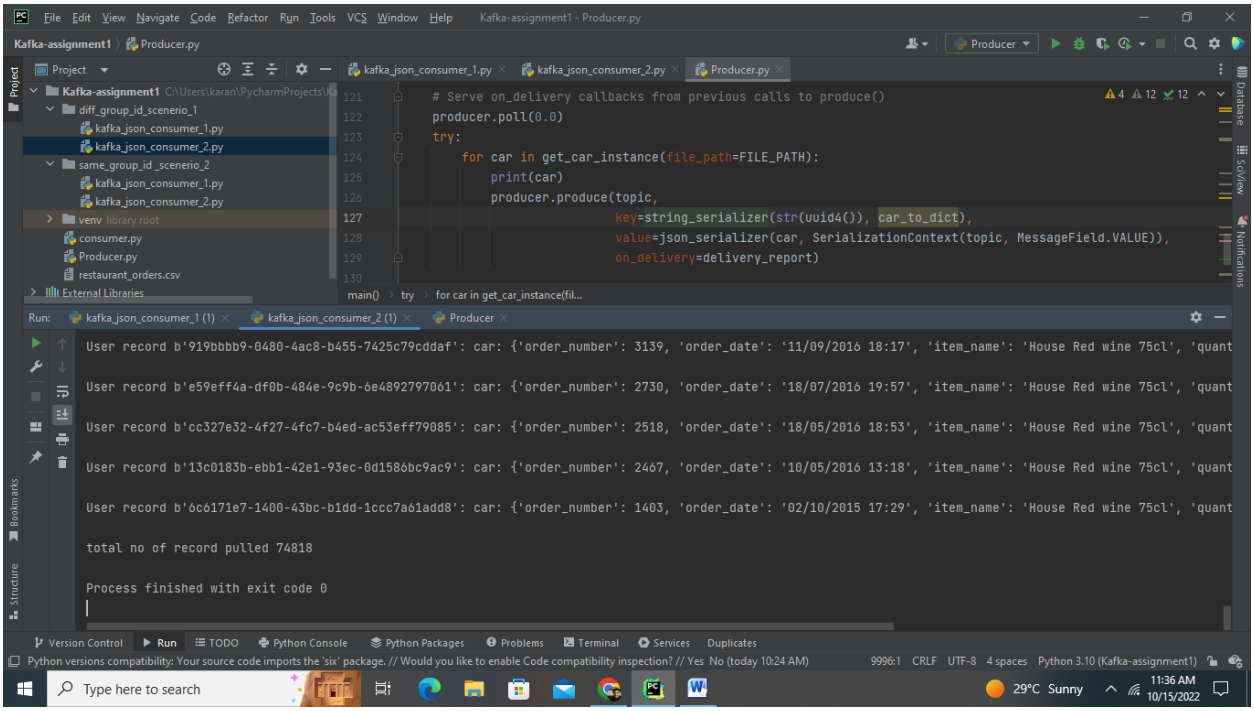
- **Producer.py Producer successfully publishes 74818 records**



Kafka_json_consumer_1 terminal: (total no of record consumed from the topic : 74818)



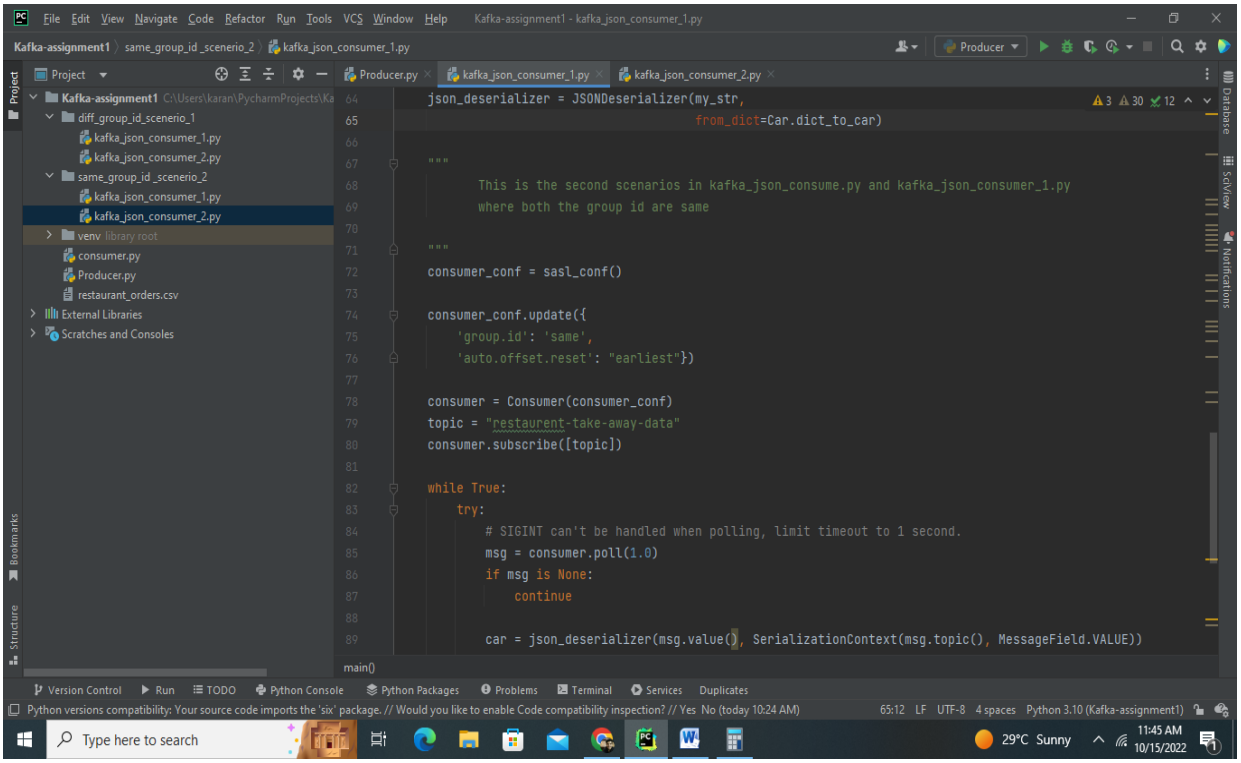
Kafka_json_consumer_2 terminal: (total no of record consumed from the topic : 74818)



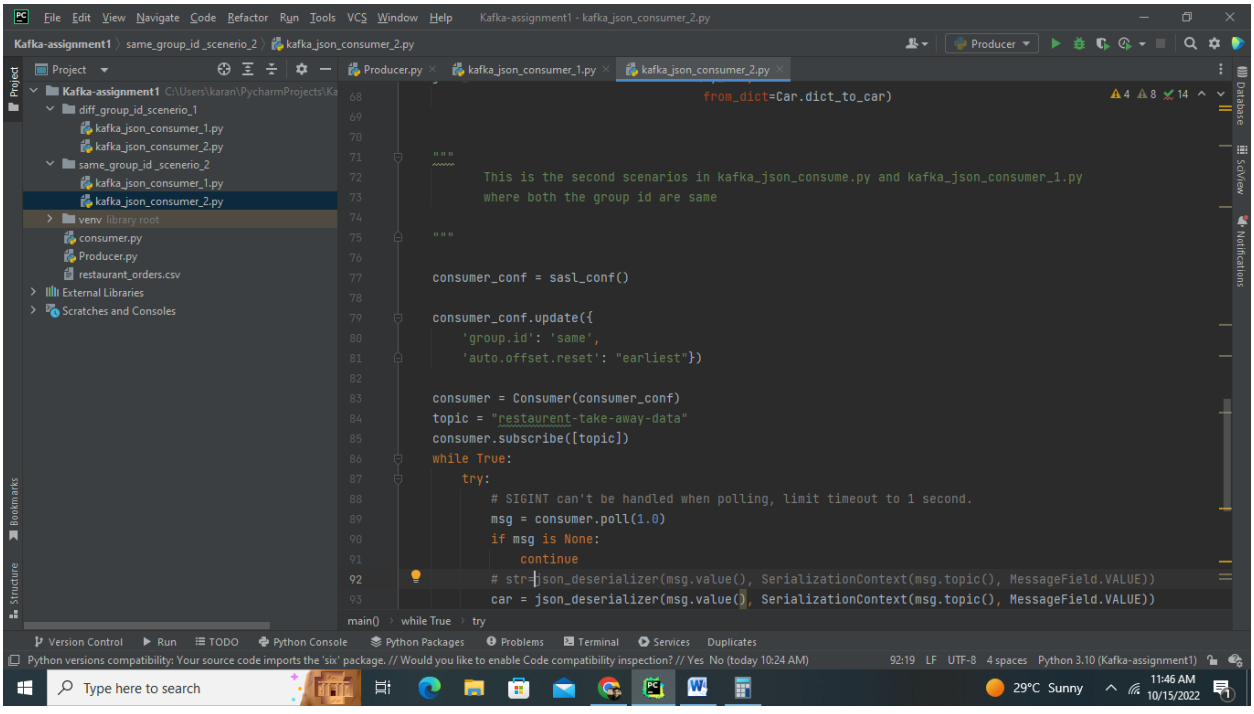
Qus) Use "group.id" property in consumer config for both consumers and mention same group_ids in kafka_consumer_1.py & kafka_consumer_2.py, apply "earliest" offset property in both consumers and run these two consumers from two different terminals. Calculate how many records each consumer consumed and printed on the terminal

***(Previous record are consumed and the terminal cleared and started producer and both consumer again so that count should be start from zero)**

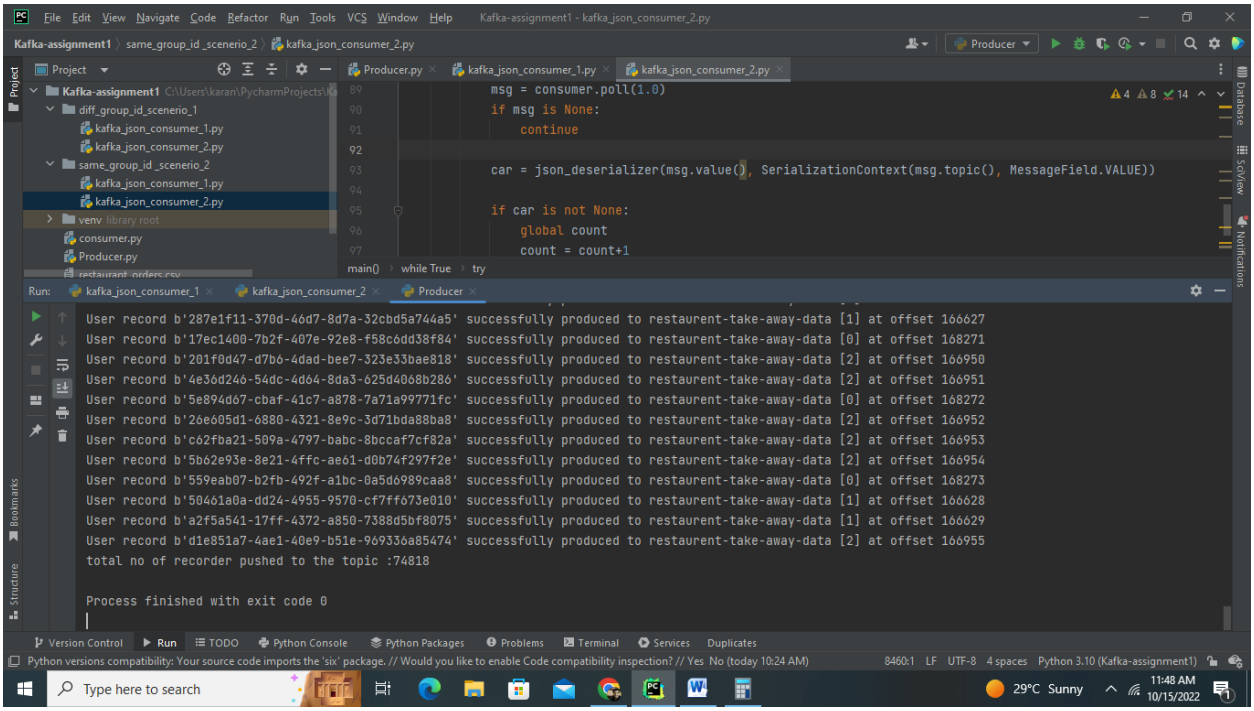
Kafka_json_consumer_1.py (group.id: ‘same’ , ‘auto.offset.reset:earliest’)



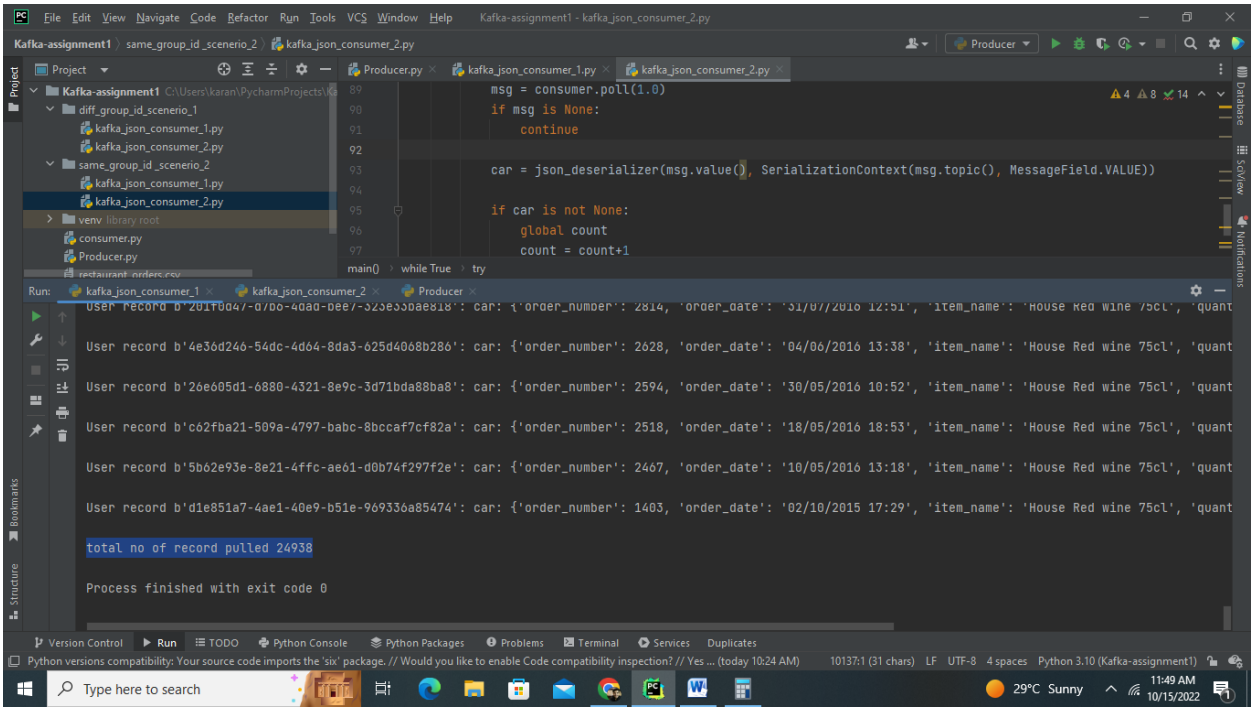
Kafka_json_consumer_2.py (group.id: ‘same’ , ‘auto.offset.reset:earliest’)



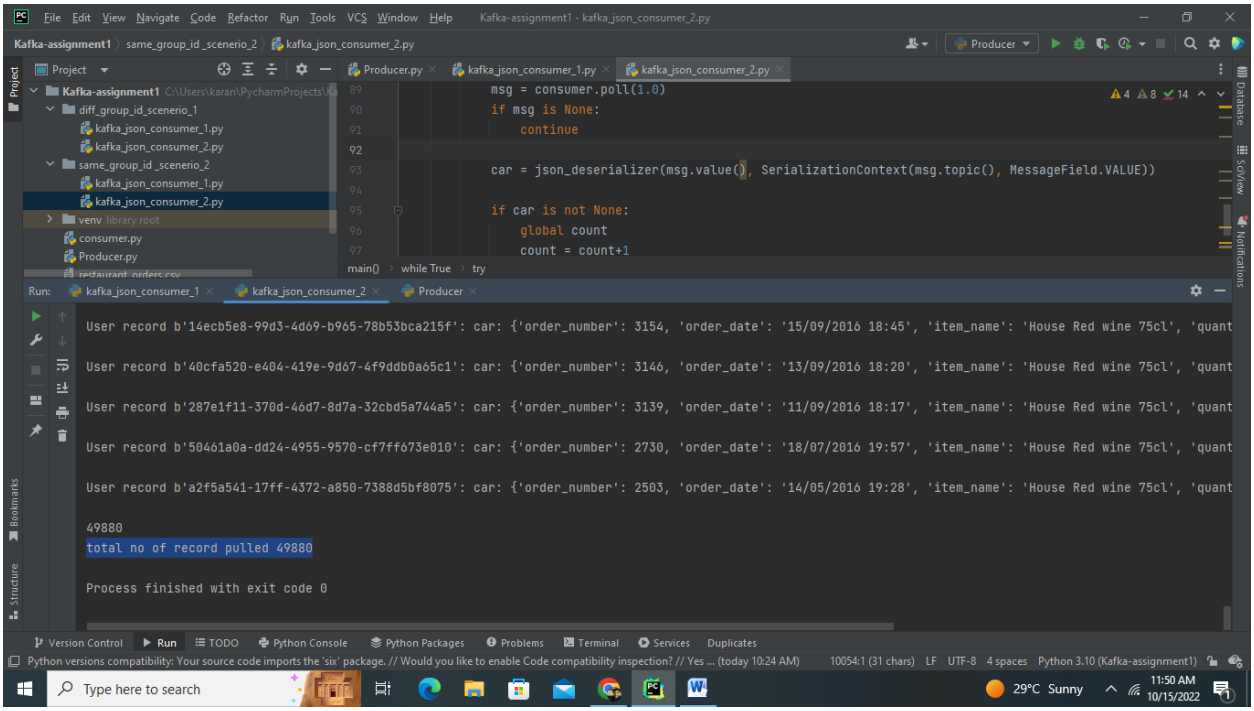
Producer terminal : (Record published : 74818)



Kafka_json_consumer_1.py terminal (total record consumed : 24938)



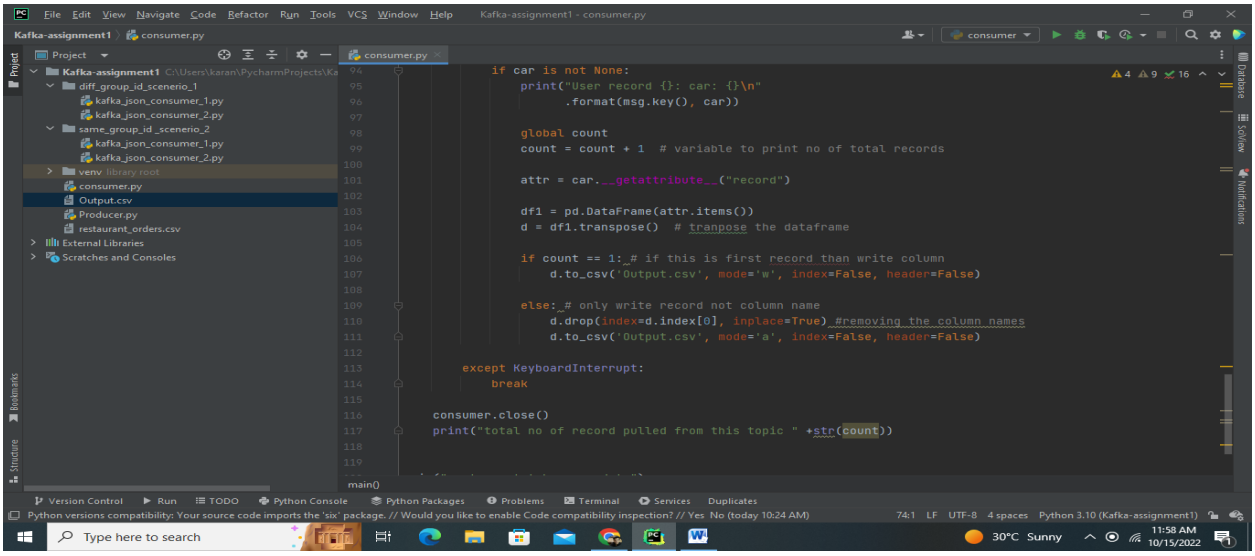
Kafka_json_consumer_2.py terminal (total record consumed :49880)



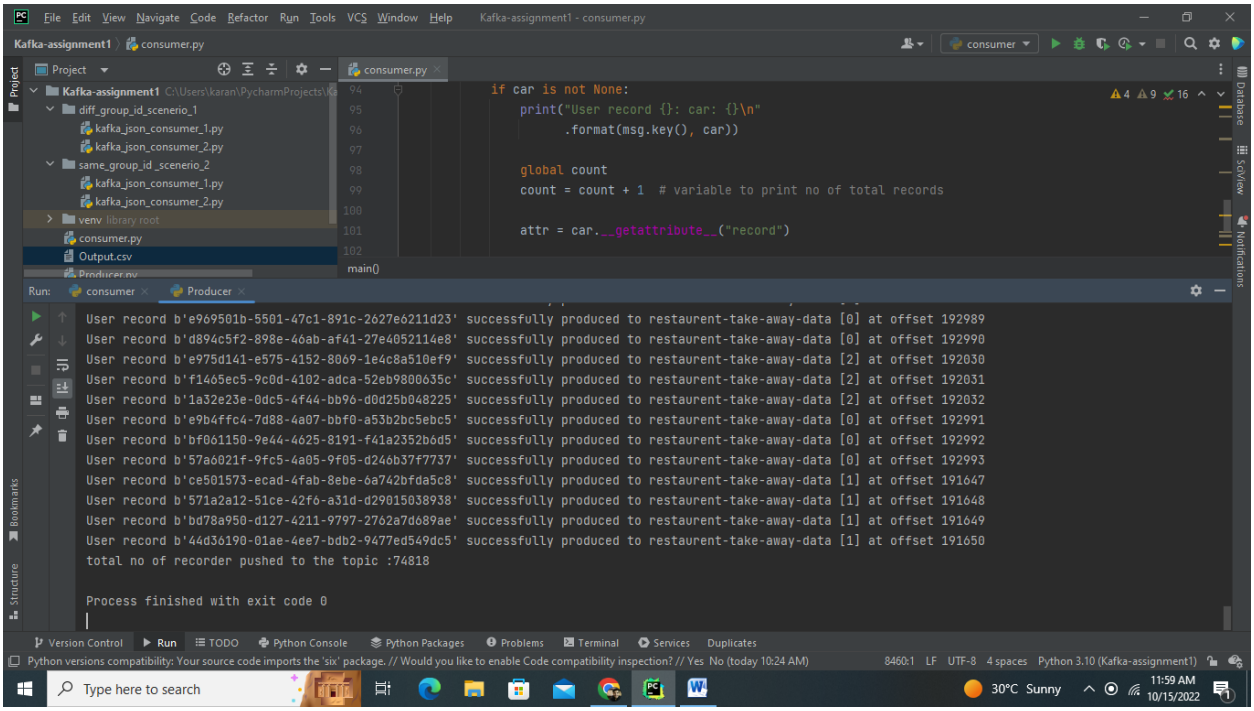
Qus) Write another kafka consumer to read data from kafka topic and from the consumer code create one csv file "output.csv" and append consumed records output.csv file

** All previous data from the topic is consumed and cleared

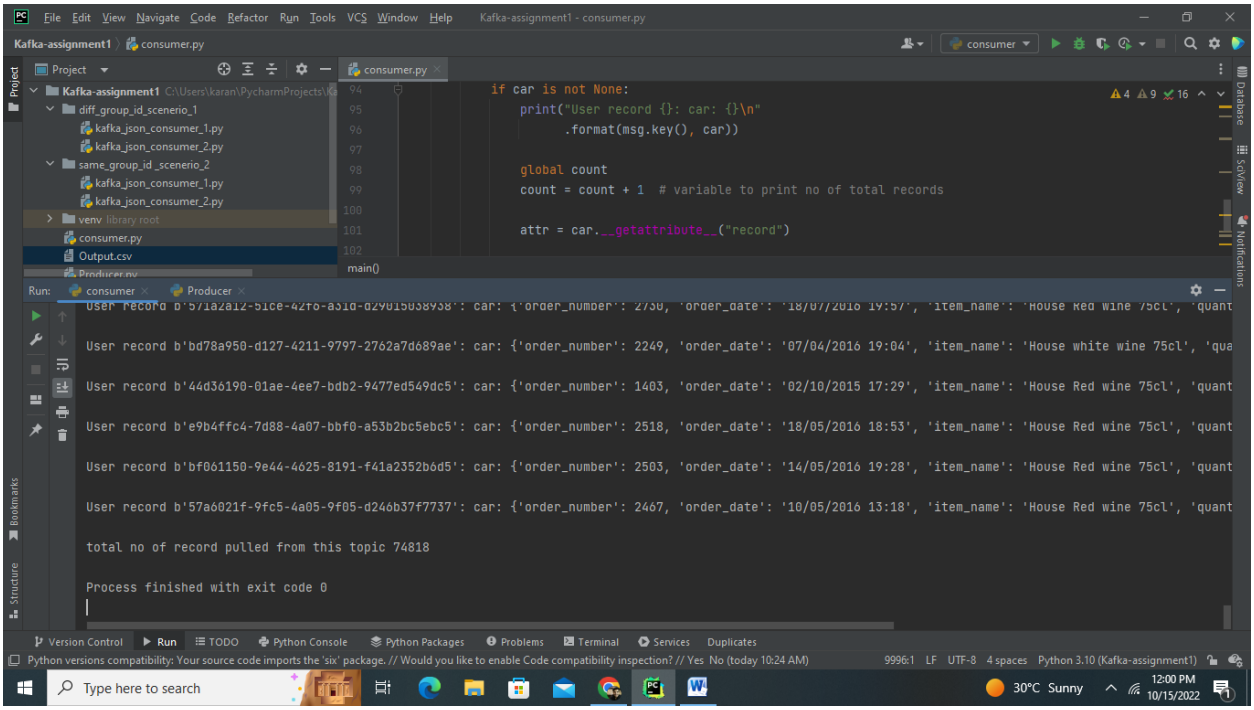
Consumer.py



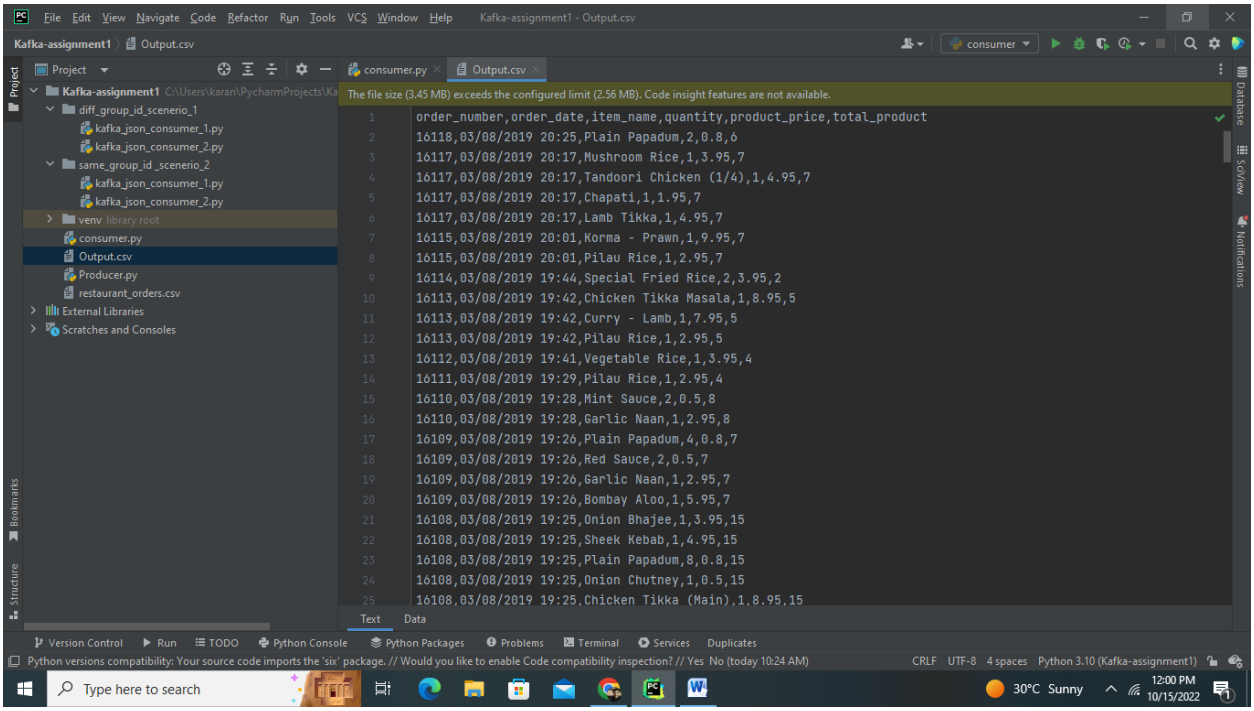
Producer terminal

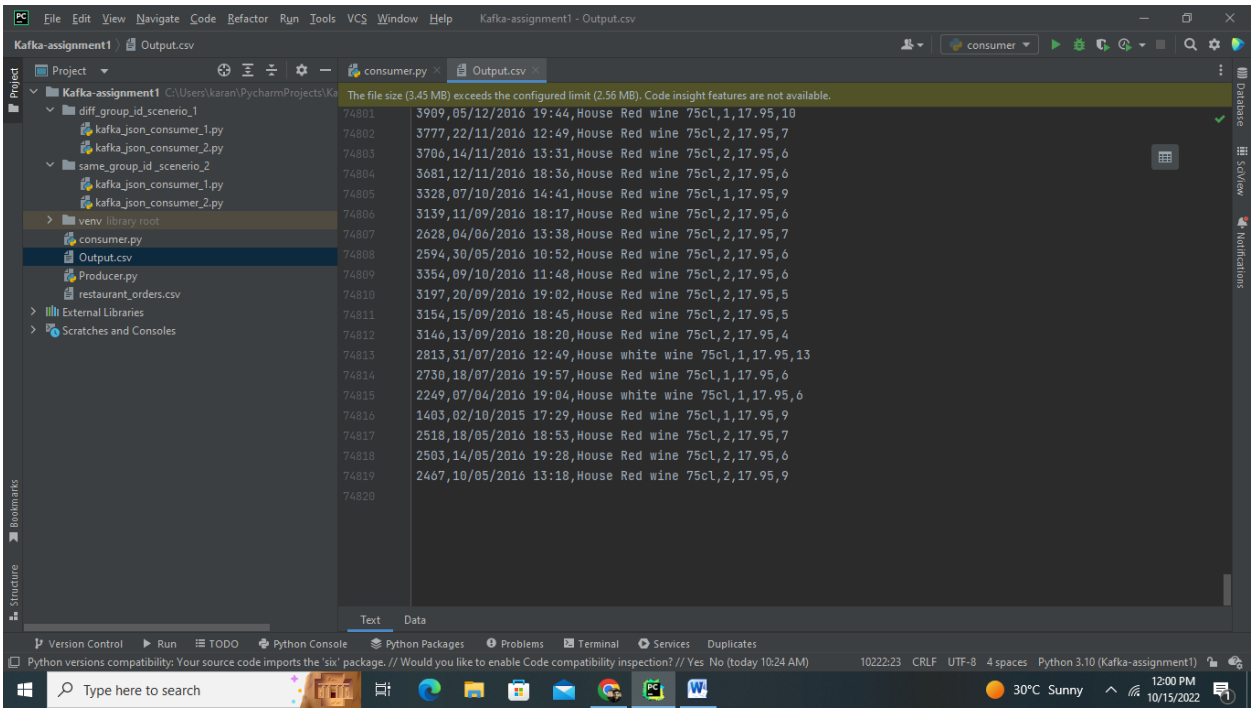


Consumer terminal



Output.csv


Kafka-assignment1 - Output.csv
Kafka-assignment1 › Output.csv
consumer.py
Output.csv
The file size (3.45 MB) exceeds the configured limit (2.56 MB). Code insight features are not available.
order_number,order_date,item_name,quantity,product_price,total_product
16118,03/08/2019 20:25,Plain Papadum,2,0.8,6
16117,03/08/2019 20:17,Mushroom Rice,1,3.95,7
16117,03/08/2019 20:17,Tandoori Chicken (1/4),1,4.95,7
16117,03/08/2019 20:17,Chapati,1,1.95,7
16117,03/08/2019 20:17,Lamb Tikka,1,4.95,7
16115,03/08/2019 20:01,Korma - Prawn,1,9.95,7
16115,03/08/2019 20:01,Pilau Rice,1,2.95,7
16114,03/08/2019 19:44,Special Fried Rice,2,3.95,2
16113,03/08/2019 19:42,Chicken Tikka Masala,1,8.95,5
16113,03/08/2019 19:42,Curry - Lamb,1,7.95,5
16113,03/08/2019 19:42,Pilau Rice,1,2.95,5
16112,03/08/2019 19:41,Vegetable Rice,1,3.95,4
16111,03/08/2019 19:29,Pilau Rice,1,2.95,4
16110,03/08/2019 19:28,Mint Sauce,2,0.5,8
16110,03/08/2019 19:28,Garlic Naan,1,2.95,8
16109,03/08/2019 19:26,Plain Papadum,4,0.8,7
16109,03/08/2019 19:26,Red Sauce,2,0.5,7
16109,03/08/2019 19:26,Garlic Naan,1,2.95,7
16109,03/08/2019 19:26,Bombay Aloo,1,5.95,7
16108,03/08/2019 19:25,Onion Bhajee,1,3.95,15
16108,03/08/2019 19:25,Sheek Kebab,1,4.95,15
16108,03/08/2019 19:25,Plain Papadum,8,0.8,15
16108,03/08/2019 19:25,Onion Chutney,1,0.5,15
16108,03/08/2019 19:25,Chicken Tikka (Main),1,8.95,15
Text
Data
Type here to search
30°C Sunny
12:00 PM
10/15/2022


Kafka-assignment1 - Output.csv
Kafka-assignment1 › Output.csv
consumer.py
Output.csv
The file size (3.45 MB) exceeds the configured limit (2.56 MB). Code insight features are not available.
3909,05/12/2016 19:44,House Red wine 75cl,1,17.95,10
3777,22/11/2016 12:49,House Red wine 75cl,2,17.95,7
3706,14/11/2016 13:31,House Red wine 75cl,2,17.95,6
3681,12/11/2016 18:36,House Red wine 75cl,2,17.95,6
3328,07/10/2016 14:41,House Red wine 75cl,1,17.95,9
3139,11/09/2016 18:17,House Red wine 75cl,2,17.95,6
2628,04/06/2016 13:38,House Red wine 75cl,2,17.95,7
2594,30/05/2016 10:52,House Red wine 75cl,2,17.95,6
3354,09/10/2016 11:48,House Red wine 75cl,2,17.95,6
3197,20/09/2016 19:02,House Red wine 75cl,2,17.95,5
3154,15/09/2016 18:45,House Red wine 75cl,2,17.95,5
3146,13/09/2016 18:20,House Red wine 75cl,2,17.95,4
2813,31/07/2016 12:49,House white wine 75cl,1,17.95,13
2730,18/07/2016 19:57,House Red wine 75cl,1,17.95,6
2249,07/04/2016 19:04,House white wine 75cl,1,17.95,6
1403,02/10/2015 17:29,House Red wine 75cl,1,17.95,9
2518,18/05/2016 18:53,House Red wine 75cl,2,17.95,7
2503,14/05/2016 19:28,House Red wine 75cl,2,17.95,6
2467,10/05/2016 13:18,House Red wine 75cl,2,17.95,9
Text
Data
Type here to search
30°C Sunny
12:00 PM
10/15/2022