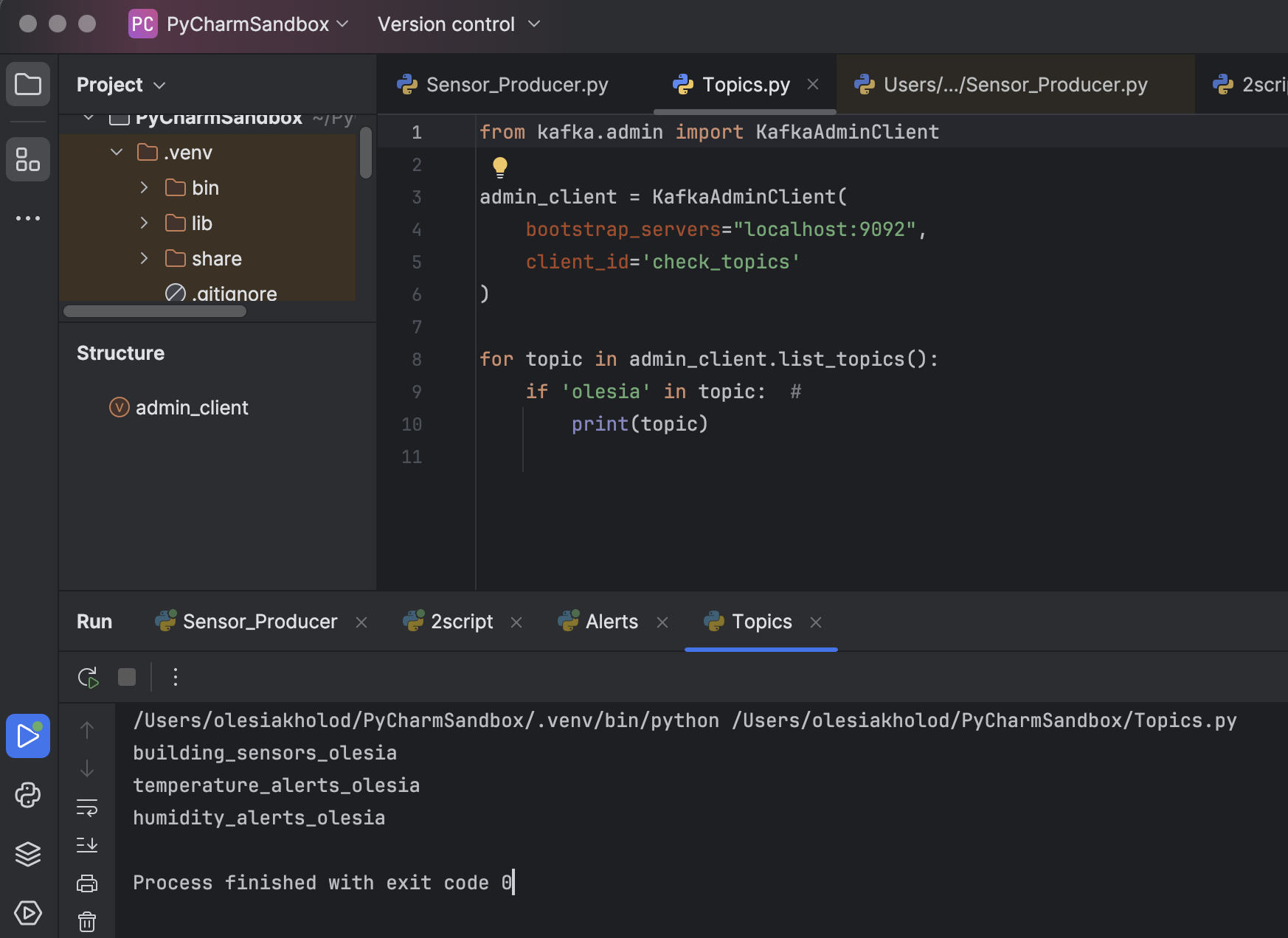
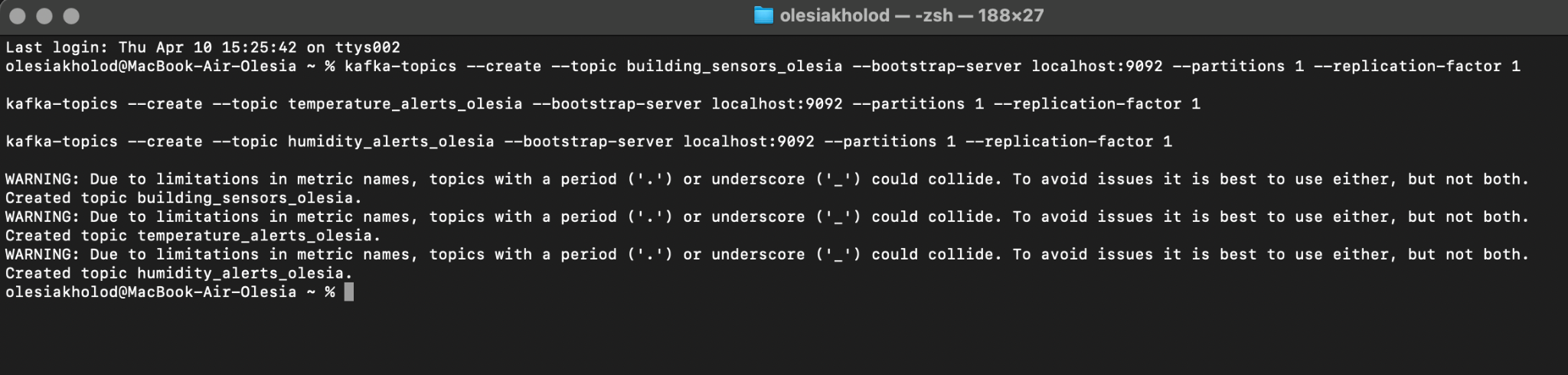
### **📌 1. Створення топіків в Kafka**

Створено 3 топіки через CLI-команди:

* building\_sensors\_olesia — для зберігання даних з сенсорів
* temperature\_alerts\_olesia — для температурних алертів
* humidity\_alerts\_olesia — для алертів вологості

Команда для перевірки топіків виконана через KafkaAdminClient, результат виведено через print(topic)



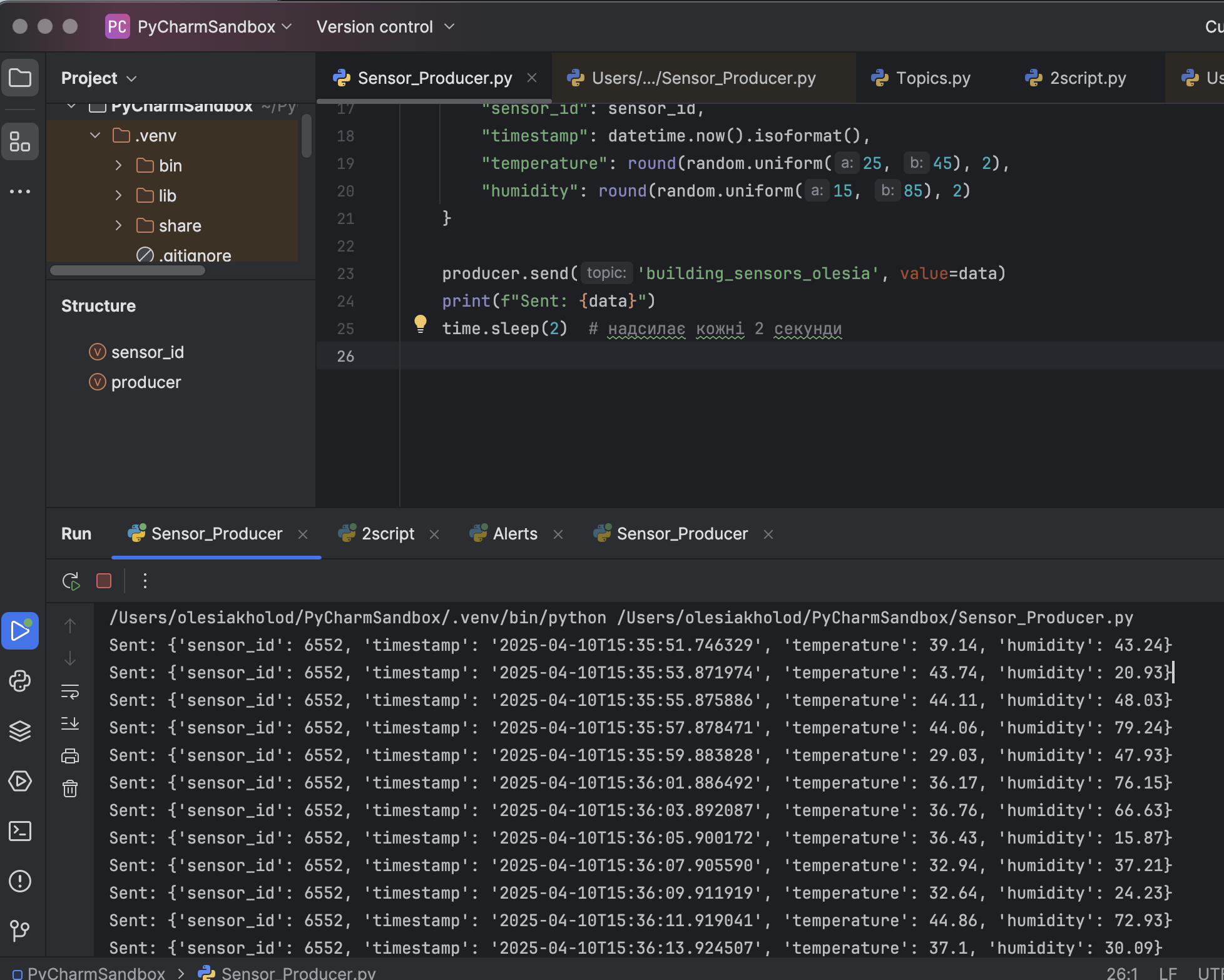
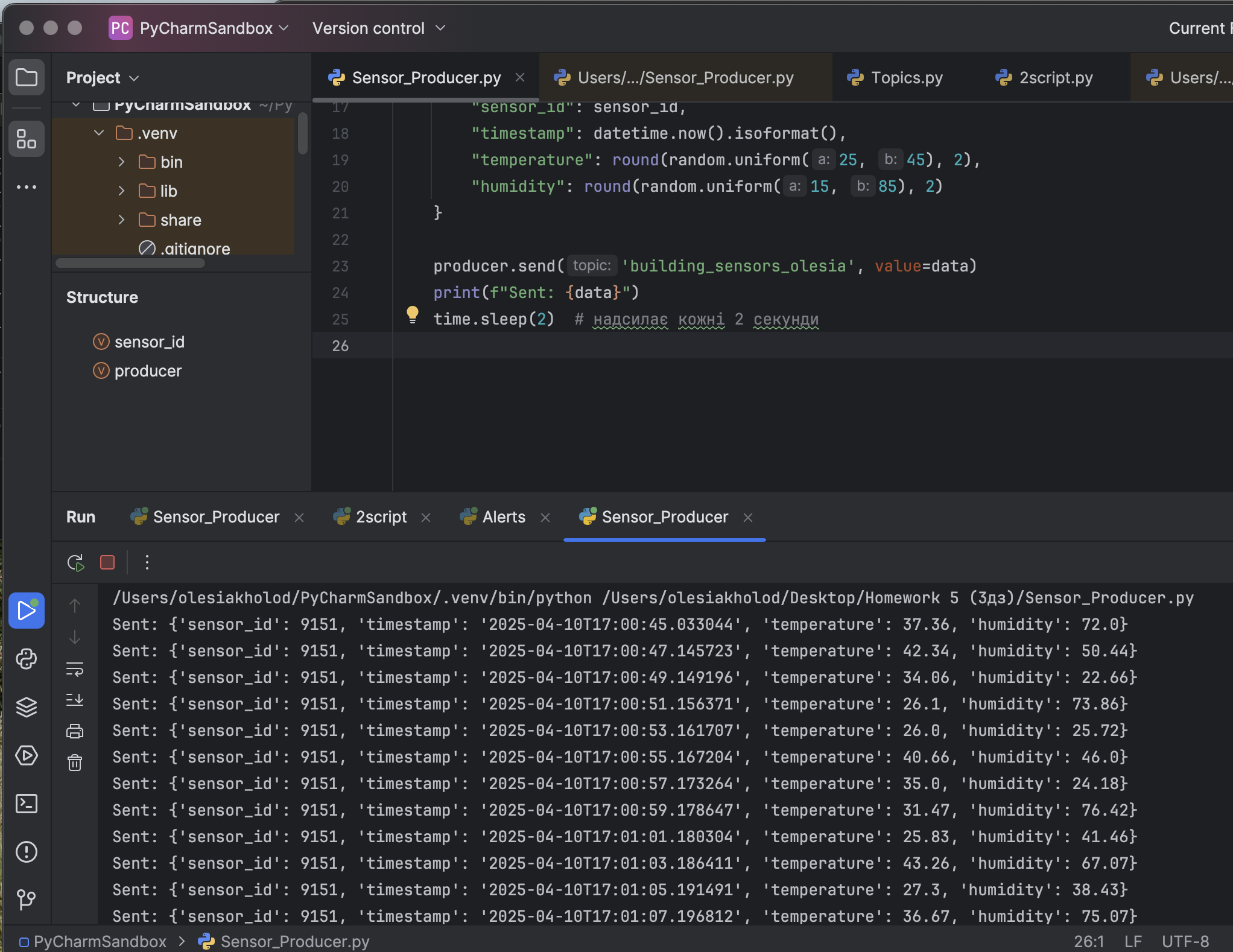


### **📌 2. Генерація та відправка даних у building\_sensors\_olesia**

Написано Python-скрипт Sensor\_Producer.py, який:

* Створює унікальний sensor\_id для кожного запуску
* Кожні 2 секунди генерує випадкові значення температури (25–45°C) та вологості (15–85%)
* Надсилає дані у топік building\_sensors\_olesia

✅ Продемонстровано **одночасну роботу двох запусків** з різними sensor\_id — 6552 та 9151.

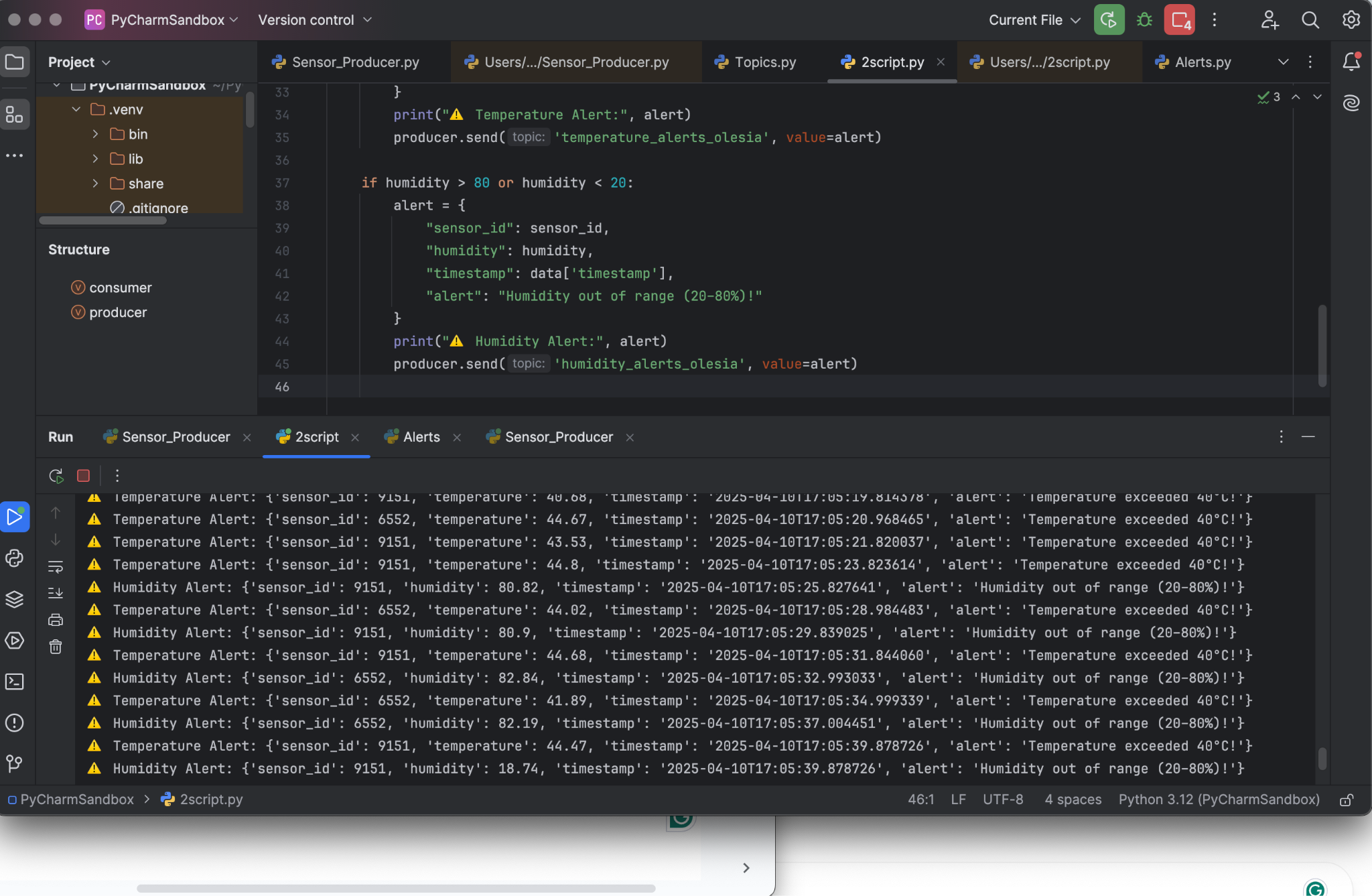


### **📌 3. Обробка даних і фільтрація алертів**

Написано скрипт 2script.py, який:

* Підписується на топік building\_sensors\_olesia
* Зчитує повідомлення та перевіряє умови:  
  + температура > 40°C → створює алерт для temperature\_alerts\_olesia
  + вологість > 80% або < 20% → створює алерт для humidity\_alerts\_olesia
* Алерти містять sensor\_id, значення, час та текст попередження

✅ Фільтровані дані успішно надсилаються у відповідні топіки.



### **📌 4. Отримання алертів з відповідних топіків**

Скрипт Alerts.py:

* Підписується на temperature\_alerts\_olesia та humidity\_alerts\_olesia
* Виводить усі отримані алерти в консоль

✅ Відображено результат із обох топіків (видно у консолі — червоні повідомлення з алертами)

