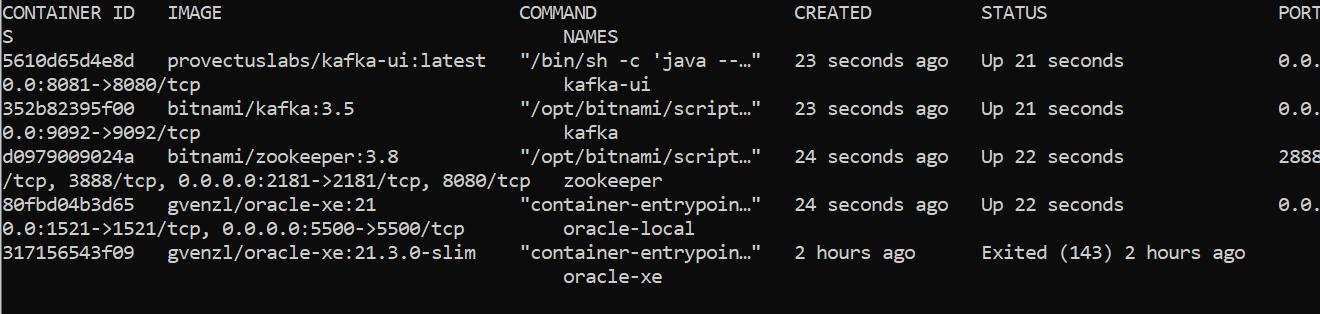
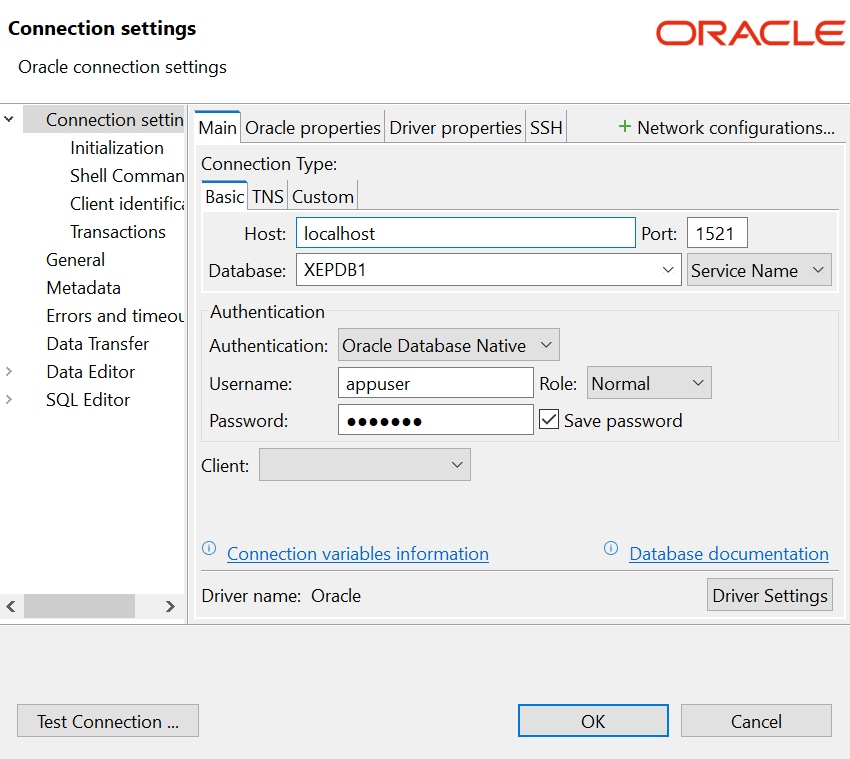
**דרישות פרויקט kafka stream:  
התקנה:**  
יש להריץ דרך ה cmd בתקייה שבה נמצא ה docker את הפקודות הבאות:

1. יש להריץ את הפקודה - docker-compose up -d
2. יש להריץ את הפקודה - docker ps -a ולבדוק שיש את ה output הבא:
3. 
4. הורדת ה docker בפקודה הבאה: docker compose down -v
5. יש להתחבר לאורקל דרך ה cmd או דרך ui – אני ממליצה על dbeaverפרטי החיבור:  
   
6. יש לבדוק את ה kafka-ui או פקודות הרצה ב cmd-   
   docker exec -it kafka bash   
   ואחר כך - kafka-topics.sh --list --bootstrap-server localhost:9092  
   ולראות שאכן קיים הטופיק: output\_results  
   http://localhost:8081/  
     
   A screenshot of a computer

   AI-generated content may be incorrect.

**בסיום שלב זה אתם מוכנים להתחיל את הפרויקט.**   
חלק א -  
**input- קבלת הקלט מתוך טופיק :**

1. טופיק – input\_zip\_topic  
   הקלט אמור להיות מהצורה:  
     
   {zipPath: “path in window to zip”}  
     
   \*\* מאחר והנתיבים לוקאלים יש ליצור אינדיקציות בטופיק לכל אחד מהזיפים שניתנים כקלט בנתיב לבחירה.
2. תיקיית קבצי זיפ שמכיל זיפים עם תמונות, זיפים עם csv וקובץ metadata, זיפים שאינם מכילים metadata וזיפים תקולים. **בחלק א נעבוד רק על זיפים שמכילים קובץ metadata ו csv-** הם יהיו בקידומת 111\_csv\_with\_metadata\_[Random].zip

**לוגיקות –**

1. יש לקרוא את האינדיקציות בטופיק בצורה סטרימית, לפלטר הודעות null.
2. יש לוודא קבלת metadata מהזיפ – לפתוח אותו, לקרוא את קובץ ה metadata שהוא בפורמט json (ניתן לבצע על ידי אוביקטים של ספרינג) יש להעביר את הקובץ לתוך אוביקט map של java. יש לקבל מתוך הקובץ את ה reasourceID ולהעשיר אותו בנתונים מתוך הטבלה – **resource\_description –** טבלה זו סטטית ומשתנה לעיתים רחוקות, יש לטעון אותה לתוך cache ולרפרש אותה כל 5 דקות. יש לקחת את מספר הגרסה, שם ה resource וסוג הקבצים מתוך הטבלה ולהכניס לתוך map ה metadata
3. יש לקבל את הזיפ מהתיקייה, לפתוח אותו, לעבור על ה csv שלו ולפרסר אותו להודעות string בצורה אסינכרונית כך שכל הודעה סופית תכיל רשומה של json שטוח מתוך ה csv. כידוע- csv הוא בפורמטר ||| ולכן יש לפרסר לפי הסגח ולפי מידע על שדות וסוגים - field\_type מהטבלה - props\_data

**Output:**

יש לכתוב כל רשומת של json בשתי שורות:  
שורה ראשונה – json של metadta, יש לקחת את ה map של ה metadata ולהפוך אותו ל json   
שורה שנייה – ההודעה עצמה – json לאחר הפרסור  
בין שתי השורות \n  
יש לשלוח את כל ההודעות לטופיק output\_results  
יש לשלוח את ההודעות עם key ייחודי לכל רשומה בכל csv