

## פרויקט מבחן לקבלה לעבודה בחברת 1 2 3 קומפליטד

הערה חשובה: כמובן שלא תקבל כסף עבור פיתוח הפרויקט והוא שייך לך. כמו כן בכל מקרה לא מובטחת לך עבודה. יחד עם זאת מספר בוגרי קורסי הסבה הצליחו לבצע פרויקט מסוג זה (או אחר) והצטרפו לחברתנו לאחר שנמצאו מתאימים.

הפרויקט עלול להראות מפחיד הרבה יותר ממה שהוא באמת. אני ממליץ בחום לבצע את הפרויקט שלב אחר שלב ולא לקפוץ בין השלבים. רק אחרי שוודאת שהשלב הספציפי עובד כשורה אז עבור לשלב הבא.

במסגרת פרויקט זה תצטרך ללמוד לבד שימוש בסיסי במוצרים הבאים:

1. Apache Kafka
2. MongoDB
3. Redis
4. Docker and Docker Compose
5. GIT

מבחן הכניסה ייכתב ב Python.

בדיקת המבחן תעשה בפגישת זום (או פנים אל פנים אם יתאפשר).

תיאור הפרויקט: הפרויקט מורכב ממספר שרתים מיקרוסרביסים (Docker Containers) שלכל אחד תפקיד מסוים. עליך לכתוב 3 תוכניות מחשב (Python MAIN) המעבירות אובייקט של EVENT משרת נתונים לחברו

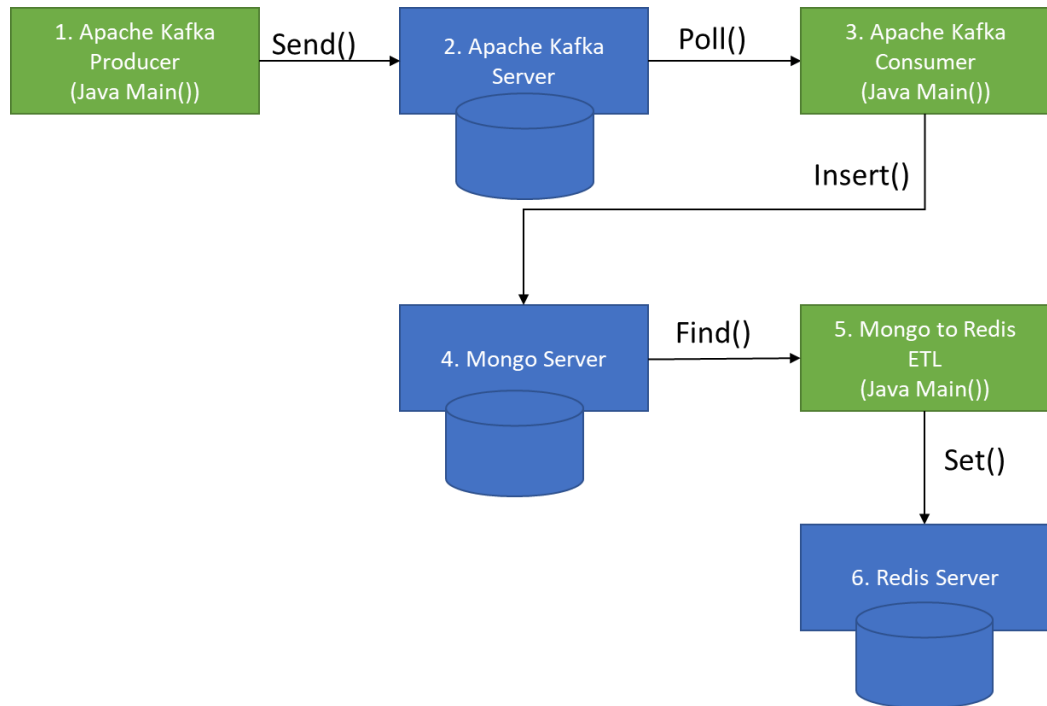
### שדות האובייקט EVENT

```
int reporterId;
Date timestamp;
int metricId; // Random from 1..10
int metricValue; // Random from 1..100
String message; // "HELLO WORLD"
....
```

## שלב מקדים לפיתוח הפרויקט

מומלץ לבצע את הפרויקט על מחשב נייד או ניח בעל לפחות 8 GB RAM  
כל ההתקנות חייבות לרוץ על סביבת הפיתוח (אפשר בעזרת DOCKER COMPOSE)  
בפרויקט עד 7 קונטיינרים (בפועל נסתפק גם ב 3 קונטיינרים כשכל קונטיינר מריץ את הפייתון MAIN של סעיפים 1, 3 ו 5). אפשרי שכל שאר השרתים יתקנו מקומית או על VM  
ודא שיש לך התקנת דוקר וסביבת דוקר עובדת על המחשב.

## ארכיטקטורת הפרויקט (היכן שכתוב JAVA החלף ב פייתון).



## שלב א: APACHE KAFKA - מנגנון תורים

1. התקן APACHE KAFKA , קובייה מספר 2 בשרטוט, ובדוק כיצד להריץ השרת (קודם כל תצטרך להריץ ZOOKEEPER). בדוק שאתה יודע כיצד לשלוח הודעה, לקבל הודעה, ליצור TOPIC, למחוק TOPIC ולבדוק האם יש הודעות ב TOPIC (בכלי CLI של קפקא)  
2. KAFKA Producer , קובייה מספר 1 בשרטוט:
  - a. בנה תוכנית main() שמייצרת באופן רנדומאלי EVENT כל שניה ושולחת את האובייקט לקפקא כ JSON.
  - b. חשוב ששדה ב timestamp יכיל את התאריך והשעה הנוכחיים.
  - c. חובה שבכל הרצה של קובייה מספר 1 ה reporter\_id יתחיל מערך 1 ויתקדם בקפיצות של 1.
  - d. בדוק בעזרת כלים של קפקא שאכן האובייקט נשלח ל KAFKA. ניתן לחלק את התוכנית למספר CLASSES
3. KAFKA Consumer , קובייה מספר 3 בשרטוט: בנה תוכנית main() שמושבת את האובייקט (JSON) מ KAFKA ומדפיסה את האובייקט למסך. ניתן לחלק את התוכנית למספר CLASSES

## שלב ב: Mongo DB - DOCUMENT DATABASE

1. התקן MongoDB , קובייה מספר 4 בשרטוט, ובדוק כיצד להריץ השרת. MONGODB הוא דאטה בייס השומר JSON בתוך COLLECTIONS (המקביל לטבלה ושורות בטבלה). ודא שאתה יודע כיצד לבצע INSERT UPDATE DELETE ו SELECT הקרוי FIND במונגו.
2. KAFKA Consumer , קובייה מספר 3 בשרטוט: הוסף ל CONSUMER קוד שמכניס את ה JSON לתוך COLLECTION ששמו EVENTS. כך שעבור כל EVENT שנוצר על ידי ה PRODUCER ונשלח ל KAFKA אז ה

CONSUMER עושה לו INSERT לתוך MONGO. יש לוודא שה timestamp אכן נכנס בצורה נכונה כשדה DATE למונח.

## שלב ג: KEY VALUE STORE DATABASE - REDIS DB

1. התקן REDIS, קובייה מספר 6 בשרטוט, ובדוק כיצד להריץ השרת. REDIS הוא דאטה בייס השומר KEY VALUE (מזכיר HashMap). ודא שאתה יודע כיצד לבצע פעולות SET, GET, CLEARALL תהליך משיכה, שנוי וטעינה (Extract Transform Load) למידע מ MONGO ל REDIS, קובייה מספר 5.
2. בנה תוכנית Main() הפועלת כל 30 שניות (THREAD). התוכנית תעתיק את כל ה JSON מ MONGO לרדיס אך תעשה זאת אך ורק ל JSONים שלא הועתקו עדיין. לשם כך השתמש בשדה ה timestamp. כאשר מתעורר ה THREAD הוא בודק מה timestamp האחרון שהועתק ל REDIS וקורא את כל ה JSONים עם ה Timestamp הגדול ממנו. וודא שהשליפה ממונח יעילה מבחינת ביצועים ואינה שולפת מידע מיותר. מבנה ה KEY ברדיס:

<reporterId>:<timestamp YYYYMMDDHH24:MISS>

לדוגמא: אם

reporterId=54

timestamp=23/1/2021:09:30:22

אז ה KEY יהיה:

KEY=54:23/1/2021:09:30:22

ה VALUE יהיה מחרוזת ה JSON שנקראה ממונח

### הערות נוספות:

1. אחרי סיום ביצוע הפרויקט יש לשלוח אותו לאיל רובין, בלינק ל [Eyal.rubin@123completed.com](mailto:Eyal.rubin@123completed.com).
2. ניתן להוסיף CLASSES וקוד לפי הצורך
3. אסורים ליטראלים (קבועים מספריים או מחרוזתיים) בקוד. יש להוציאם לקובץ קונפיגורציה ב YAML.
4. ניתן לעשות כל שימוש נוסף הנראה רצוי ב REDIS או מונח. זאת אומרת שניתן להוסיף KEYS ו COLLECTIONS את/אתה מוזמן לפנות אלי בכל שאלה.
5. לחברת 1 2 3 קומפליטד יש הזכות להפסיק התהליך בכל רגע לפי שיקול דעתה הבלעדי.
7. בזמן פגישת הבדיקה
  - a. הדבר החשוב ביותר הוא הדמו בו עליך להציג את הפתרון באופן שותף וללא תקלות. כל תקלה או התמהמהות בהסבר או בעיה בדמו תקשה עלינו להמשיך בתהליך בחינת קבלתך כעובד.
  - b. לאחר מכן ניכנס לבחון את איכות הקוד ורמת ההבנה שלך בפרויקט. אין צפי להבנה מליאה בשרתים עצמם אך בהחלט יש צפי שתוכל להסביר כל שורת קוד ולנהל דיון אינטליגנטי בנושא.
  - c. הדמו של הפרויקט מתחיל כשאין שום דאטה בשרתי הדאטה וכל הקונטיינרים למטה (לא עובדים)
  - d. שאלות בדמו של הפרויקט:
    - i. להרים קונטיינר קפקא
    - ii. לייצר TOPIC בקפקא
    - iii. להריץ את ה PRODUCER
    - iv. להוכיח שיש דאטה ב TOPIC בעזרת כלי קפקא
    - v. להרים MONGO
    - vi. לייצר COLLECTION
    - vii. להריץ CONSUMER
    - viii. להראות שיש דאטה ב MONGO
    - ix. להרים REDIS
    - x. להריץ ETL (קובייה 5)
    - xi. להראות שיש דאטה ב REDIS
    - xii. לעצור ETL
    - xiii. להריץ שוב ETL
    - xiv. להראות שה ETL ממשיך לעבד רק מהמקום האחרון שהוא עיבד.
8. אתה לא חייב לעשות את הפרויקט בשום מקרה אך זו משימה שמהווה תנאי הכרחי לשקילת כניסתך לחברה.

בהצלחה,








איל רובין  
3 2 1 קומפליטד, מנכ"ל  
[eyal.rubin@123completed.com](mailto:eyal.rubin@123completed.com)  
[www.123completed.com](http://www.123completed.com)  
054-2607104



**Eyal Rubin**

CEO

 [www.123completed.com](http://www.123completed.com)

 +972-54-260-7104  
 +972-3-503-0027  
 +972-72-250-9027  
 [eyal.rubin@123completed.com](mailto:eyal.rubin@123completed.com)