

## EASS שיעור 10 - docker compose, Streamlit

חלק שלישי של המטלה-

- Docker-compose
- סרטון קצר שמציג כיצד להריץ את האתר (docker-compose up וכו')
- לפחות 3 מיקרוסרוויסים (backend, frontend הם שני מיקרוסרוויסים שחייב לעשות)

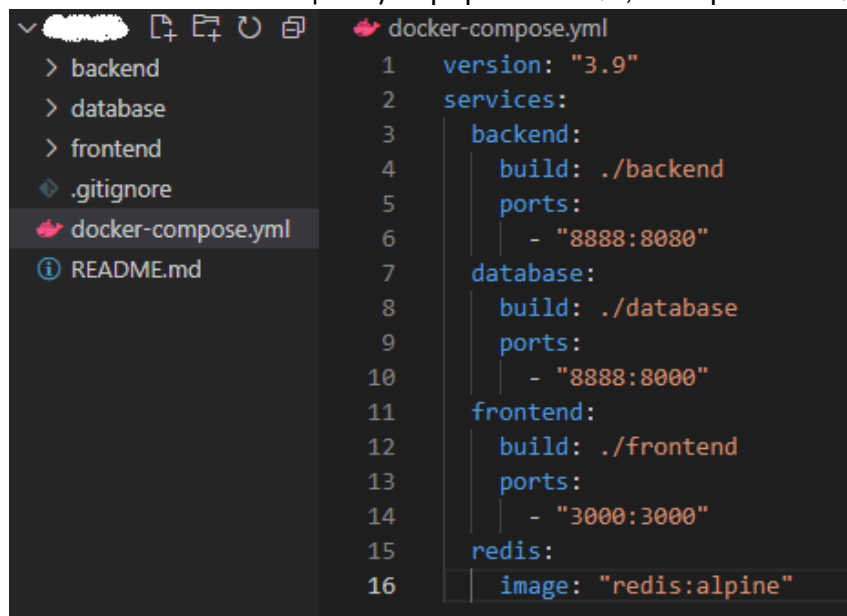
:Docker-compose

מאפשר חיבור של כמה microservices באמצעות קובץ אחד מסוג yml.

נבצע את הבנייה של image באמצעות פקודה docker-compose build (במקום docker build שעשינו עד כה)

נבצע את יצירת הקונטיינר באמצעות פקודה docker-compose up (במקום docker run שעשינו עד כה)

עבור הפרוייקט שלנו, עלינו לבנות קובץ docker-compose.yml בצורה הבאה:



```

1  version: "3.9"
2  services:
3    backend:
4      build: ./backend
5      ports:
6        - "8888:8080"
7    database:
8      build: ./database
9      ports:
10       - "8888:8000"
11   frontend:
12     build: ./frontend
13     ports:
14       - "3000:3000"
15   redis:
16     image: "redis:alpine"

```

נשים לב למילים השמורות:

- Version- הגרסה של docker-compose
- service- הצגת המיקרוסרוויסים שנרצה לחבר באמצעות docker-compose
- build- מיקום התיקיה של המיקרוסרוויס
- ports- הפורטים שדרכם המיקרוסרוויס יעבוד. הפורט השמאלי הוא הפורט החיצוני (זה שנרשום בדפדפן לצד: localhost) והפורט הימני הוא הפורט הפנימי של הקונטיינר.
- Image- קריאה לimage שקיים בdocker hub ונרצה להשתמש בו (במקום ליצור image עם תיקייה נוספת שיש בה Dockerfile ושאר הרכיבים שבונים אותו). נרשום את שם ה-image והגרסה שלו כמו שקיים ב-docker hub. במידה ונשתמש בזה ניישם את השימוש בסרוויס זה בתוך אחד מקבצי py. בעזרת ספריות client של פייתון.
- Volumes- העתקת קובצים לוקליים לקונטיינר (מתאים לשימוש בסרוויס database). משמאל לנקודותיים הנתיב הלוקלי, מימין לנקודותיים הנתיב בקונטיינר.

```

volumes:
  - ./database:/src

```

:Streamlit

ספרייה בפייתון שיוצרת UI (user interface).

שימוש: תחילה יש להריץ `pip install streamlit`.

הערה: אם נבחר להשתמש ב-streamlit ב-frontend שלנו, נשים לב שכאשר ניצור את ה-dockerfile של ה-frontend נכתוב בקובץ requirements.txt את streamlit וכך פעולת ההתקנה תתבצע גם בקונטיינר.

ניצור תיקייה וקובץ

```
/natal$ mkdir streamlit-test
/natal$ cd streamlit-test
/natal/streamlit-test$ vi api.py
```

דוגמה לתוכן של קובץ (מהמצגת שיעור)

```
import streamlit as st
st.title("Hello, Streamlit!")
st.write("Here's our first attempt at using Streamlit")
st.markdown("""
# Streamlit is **awesome**.

That's why we use it for all our projects.
""")

st.latex(r'''
a + ar + a r^2 + a r^3 + \cdots + a r^{n-1} =
\sum_{k=0}^{n-1} ar^k =
a \left(\frac{1-r^n}{1-r}\right)
''')
```

ונריץ את הפקודה

```
natalieaflalo@LAPTOP-1GN0VKD:/mnt/c/Users/natal/streamlit-test$ streamlit run api.py --server.port 8501

You can now view your Streamlit app in your browser.

Network URL: http://172.19.88.143:8501
External URL: http://77.137.65.14:8501
```

כעת כאשר ניגש לקישור שיופיע על המסך נקבל את הדף הבא-

