Containers

אז מה זה container ומה הפתרון שהוא מציע?

**container** הוא חבילה קטנת משקל (lightweight package), שמכילה את כלל הדרישות להפעלת האפליקציה הדרושה לנו.

קונטיינר בא על מנת לפתור את הבעיה של חלוקת משאבים וניהולם. איך הוא עושה את זה?

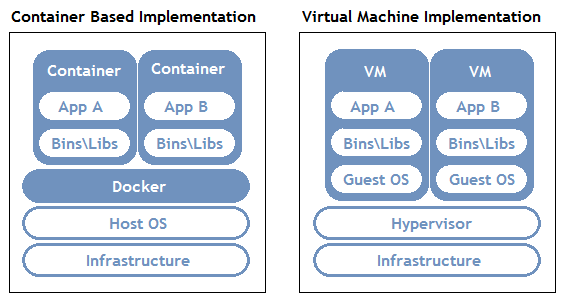
נגיד למשל שיש לנו אפליקציה שדורשת שלושה services שונים. על מנת לבצע חלוקה הגיונית, נשים כל service בשרת נפרד, או ב-VM נפרד. הפתרון הזה הוא יקר, כיוון שנדרשים שלושה שרתים לאפליקציה אחת, בנוסף זהו פתרון איטי מבחינת ביצועים ותקשורת בין ה-services.

זו הבעיה ש-container פותר לנו, הוא למעשה מיישם את הרעיון של microservices.

אבל רגע, מה זה microservices?

הם גישה ארכיטקטונית וארגונית לפיתוח תוכנה שבה התוכנה מורכבת משירותים **עצמאיים** קטנים בעלי תקשורת ביניהם. ארכיטקטורת מבוססת microservices הופכת פיתוח מהיר יותר ומאורגן יותר.

*חשוב להדגיש, שקונטיינרים מציעים בידוד בין הסביבות, ולא וירטואליזציה, זה לא virtual machine*



Docker:

למעשה מהווה את ה-container engine שלנו. זוהי פלטפורמה ליצירת וניהול containers.

docker מאפשר לנו להריץ כמה קונטיינרים שנרצה על גבי host יחיד (מכות תלוי במגבלת המשאבים שקיימת ל-host)

כיצד בעצם אנחנו יכולים להריץ קונטיינרים? התשובה היא images

מה זה image? וכיצד אנחנו משיגים אותו?

image זהו קובץ באמצעותו ניתן ליצור את הקונטיינר שלנו. קובץ זה מתאר את הדרישות ליצירת הקונטיינר והפעולות שצריך לבצע על מנת ליצור אותו: החל ממערכת ההפעלה ועד ל-services הנוספים שנרצה להתקין עליו.

image אינו ניתן לשינוי ברגע שנוצר, ניתן ליצור אחד חדש עם השינוי הרלוונטים אך לא ניתן לערוך אחד קיים

על מנת ליצור image, נצטרך ליצור קובץ בשם Dockerfile, או לשלוף image מתוך image registry, לדוגמא: docker hub.

דוגמא ל-Dockerfile:

