**שלב א-**

תיאור קצר של השלבים שעשיתי (כולל פקודות):

1. פתחנו תיקייה לפרוייקט דרך הVS
2. הפעלנו את הDOCKER
3. - Docker network create jenkins-net ניצור רשת ל־Docker   
   **עבור הבונוס**- כדי שהagent ידע לדבר עם הmaster זה בעצם יוצר להם רשת משותפת שהם מכירים זה את זה לפי שם. אחרת agent לא היה מכיר Jenkins והיינו צריכים לפתוח פשוט את הרשת הזו בשלב יותר מאוחר.
4. נריץ פקודה להורדת Jenkins ופתיחת פורט ונתינת שם לקונטיינר

docker run -d \

--name jenkins \ שם הקונטיינר.

-p 8080:8080 \ פותח את ה־UI ב־<http://localhost:8080>.

-p 50000:50000 פורט ל־agents  
-v jenkins\_home:/var/jenkins\_home → יוצר volume בשם jenkins\_home שישמור את כל ההגדרות.

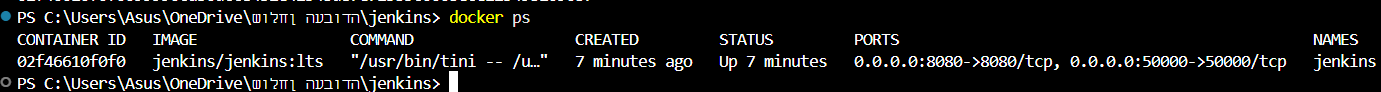
-v jenkins\_home:/var/jenkins\_home \ מחבר לרשת Docker שהכנו.

--network jenkins-net \

jenkins/jenkins:lts האימג' הרשמי של Jenkins.

1. מספרי פורטים לdocker file:  
    **8080** ממשק ה־Web של Jenkins (UI).  
    **50000** חיבורי Agents (JNLP)
2. : jenkins/jenkins:lts זה Image רשמי מה־Docker Hub שמכיל את Jenkins בגרסת LTS.

ה־Container הוא בעצם Jenkins שעובד על המחשב שלך, עם פורטים פתוחים (8080 וכו').

1. בודקים שהכל עובד  
   docker ps: מראה a running Jenkins container  
   

סיסמה ראשונית   
תמונה שמכילה טקסט, צילום מסך, גופן

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

הורדנו plug in

תמונה שמכילה טקסט, צילום מסך, מספר, תצוגה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.סיסמה 1234

תמונה שמכילה טקסט, צילום מסך, גופן

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

1. עצרנו את הקונטיינר docker stop Jenkins
2. הפעלנו מחדש docker start Jenkins
3. תמונה שמכילה טקסט, צילום מסך, פני אדם, מערכת הפעלה

   תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

התחברנו שוב מחדש

1. לטובת פתיחת repository העדפתי לעשות את זה דרך הgitHub זה לא אמור לשנות.

תמונה שמכילה טקסט, צילום מסך, גופן, תוכנה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

נשתמש בפקודות לריפוסטיטורי חדש-

תמונה שמכילה טקסט, צילום מסך, תוכנה, תכונות מולטימדיה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

**עבור מה שביקשו - scripts/hello.sh- הוספתי הערות מעל כל הדפסה לאיזה דרישה היא מתייחסת**• Must be executable- בווינדוס אין עם זה בעיה (בלינוקס צריך להריץ לזה פקודה)   
**עבור מה שביקשו - Jenkinsfile - הוספתי הערות מעל כל הדפסה לאיזה דרישה היא מתייחסת**

סוף שלב ב'-

תמונה שמכילה טקסט, צילום מסך, תוכנה, תצוגה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

בשלב הזה אנחנו לא רואים את ב Jenkins UI

שלב C:   
תמונה שמכילה טקסט, צילום מסך, גופן, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

הJenkins מתעדכן מול הגיט לבדוק אם היו שינויים כל 5 דק'

הוספתי את הURL לגיט

תמונה שמכילה טקסט, צילום מסך, מספר, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

הרצה רגילה default

תמונה שמכילה טקסט, גופן, צילום מסך, קו

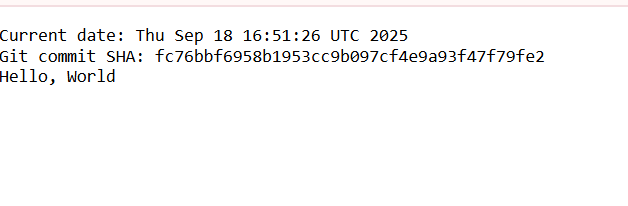
תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

הרצה עם פרמטר

תמונה שמכילה טקסט, צילום מסך, גופן, קו

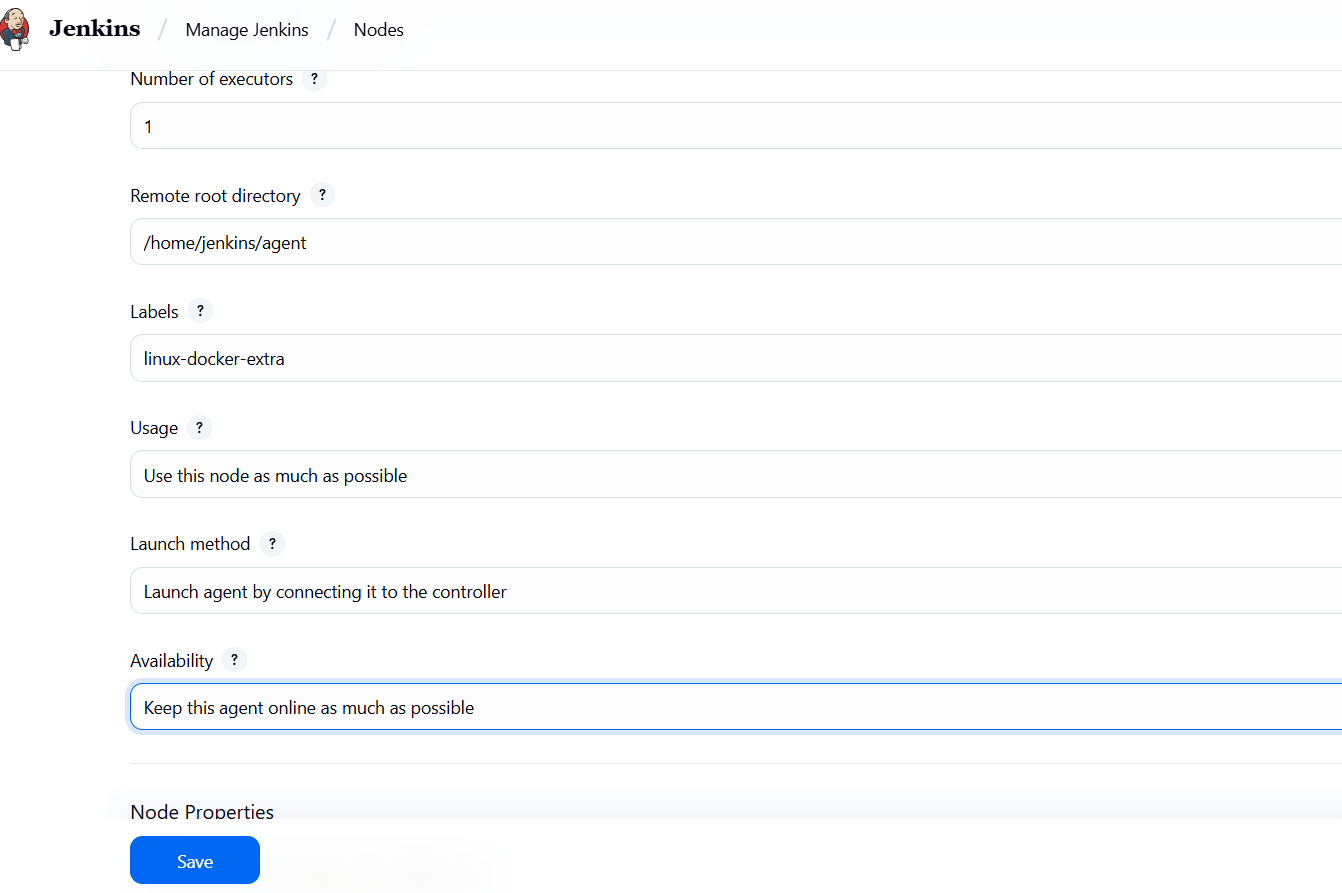
תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

אחרי כ-5 דקות מאז שעשינו push:

באופן אוטמטי בוצע build עם הפלט הבא:  


בונוס:   
עד עכשיו השתמשנו ב controller- הmasetr   
הוא  **Jenkins הראשי** זה שמנהל את ה־ ) Pipelines, מקבל טריגרים (למשל commit חדש, מתזמן עבודות, שומר לוגים ו־Artifacts. הוא לא חייב בעצמו להריץ את הבנייה (build) ויכול לנהל, כל מה שהרצנו עד עכשיו רץ על הcontroller.

Agent זה הworker – הnode שלנו. זה בעצם מחשב או **container** נוסף שאליו הקונטרולר שולח משימות , הוא זה שבפועל יבצע את הbuild – בגלל שיכולים להיות כמה כאלה זה בעצם מאפשר לנו לחלק עומסים ולעבוד בכל מיני סביבות עבודה.

לכן ניצור עכשיו node חדש עפ"י הדרישות:

לפני הפעלה-



הפעלת הnode החדש-

בדומה לפקודות בהתחלה שהורדנו Jenkins : הפעם נחבר את הnode לרשת שיצרנו קודם-

docker run -d --name jenkins-agent \ - יוצרים קונטיינר חדש + נתנו לו שם agent

--network jenkins-net \ מחברים את ה־agent לרשת Docker ככה שהagent יכיר את הcontroller שלנו

-e JENKINS\_URL=http://jenkins:8080 \ כתובת הקונטרולר שידע לאן להתחבר

-e JENKINS\_AGENT\_NAME=linux-docker-1 \ השם המזהה שלו בתוך jenkins

-e JENKINS\_SECRET=... \ הסיסמה שלקחנו מ JENKINS

-e JENKINS\_AGENT\_WORKDIR=/home/jenkins/agent \ ספריית העבודה של ה־agent בתוך הקונטיינר (לכאן ירדו הקבצים מהgit ויקרו ההרצות)

jenkins/agent:latest ה־Docker image הרשמית של Jenkins Agent כלומר זה Image שונה משל הקונטרולר ולכן הוא מוריד אותו מחדש

עכשיו רואים בjenkins ui שהnode פעיל:

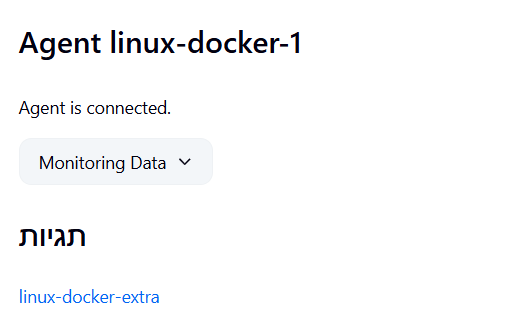
תמונה שמכילה טקסט, צילום מסך, גופן, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

עכשיו נעדכן – Jenkinsfileכך שירוץ רק על הagent שיצרנו -   
label 'linux-docker-extra'

מריצים build ומוודאים בconcole output שזה מריץ על הnode הנכון:





ניתן לראות שהוא label כפי שהתבקשנו-

ניתן לראות גם שהOutput נשמר – בartifact.

כעת נריץ פקודה לעצירת הagent container –

docker stop jenkins-agent

נוודא שהוא הפך לoffline-

תמונה שמכילה טקסט, גופן, צילום מסך, מספר

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

נפעיל מחדש-

docker start jenkins-agent  
תמונה שמכילה טקסט, גופן, קו, מספר

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

נריץ שוב build עם פרמטר test ונראה שהBuild עבר בהצלחה-

תמונה שמכילה טקסט, גופן, צילום מסך, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

צילום של docker ps- עבור סעיף 7 ב Deliverables

