פרויקט DevOps

מגישים

ראובן יצחקוב / 322855909 / 0543544493 / Reuven.itzhakov@e.braude.ac.il

שלו כהן / 207534728 / 0509985884 / Shalev.cohen@e.braude.ac.il

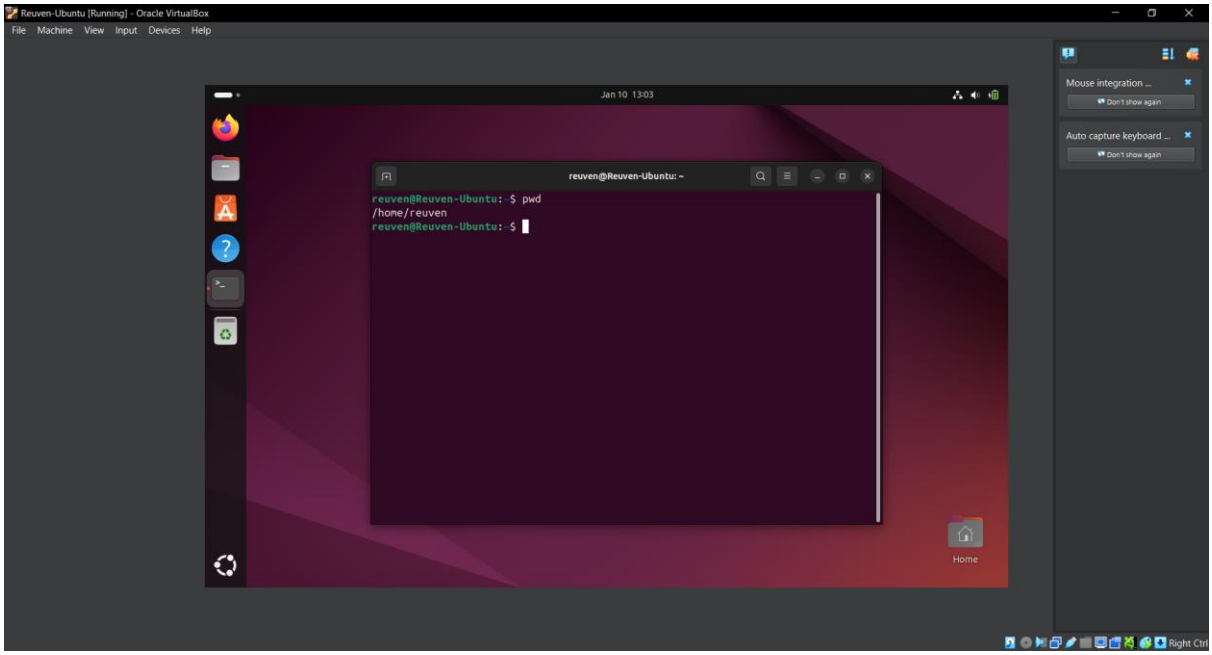
קישור לגיטהב: https://github.com/reuven-itzhakov/DevOps-Project

הרצאה 1

הורדנו Image של Ubuntu והגדרנו אותו ב-VirtualBox.

ביצענו התקנה והפעלנו את המכונה הווירטואלית.

פתחנו טרמינל והקלדנו את הפקודה pwd:



המערכת הציגה את נתיב התיקייה הנוכחית.

הרצאה 2

פתחנו את הטרמינל בתוך מכונת ה-Ubuntu.

הקלדנו את הפקודה pwd:

```
reuven@Reuven-Ubuntu:~$ pwd
/home/reuven
```

what is your username? reuven

what is your machine name? Reuven-Ubuntu

what is your current directory? /home/Reuven

יצרנו קובץ test.file באמצעות touch, ולאחר מכן השתמשנו בls על מנת לבדוק שאכן נוצר.
לאחר מכן רשמנו לקובץ שיצרנו את הטקסט "Hello World!" באמצעות echo, ואז הדפסנו את התכונה של הקובץ באמצעות הפקודה cat:

```
reuven@Reuven-Ubuntu:~$ touch test.file
reuven@Reuven-Ubuntu:~$ ls
Desktop    Downloads  my_test_project  Pictures  snap       test.file
Documents  Music      personal         Public    Templates  Videos
reuven@Reuven-Ubuntu:~$ echo "Hello World!" > test.file
reuven@Reuven-Ubuntu:~$ cat test.file
Hello World!
```

הרצאה 3

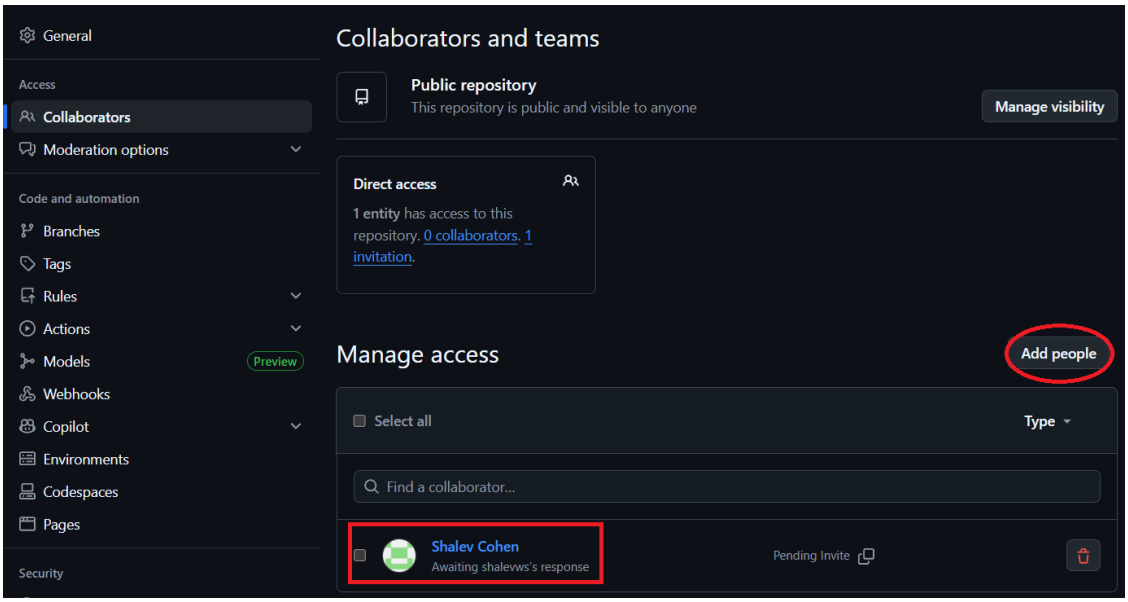
1. תשתמשו ב-git לפעולות השונות

יצרנו repo (הריפוסיטורי שנמצא בדף הראשון של מסמך זה – Devops Project)
יצרנו קובץ והוספנו אותו באמצעות git add, ערכנו אותו וביצענו commit, לאחר מכן מחקנו אותו, ביצענו commit שוב, ולבסוף push:

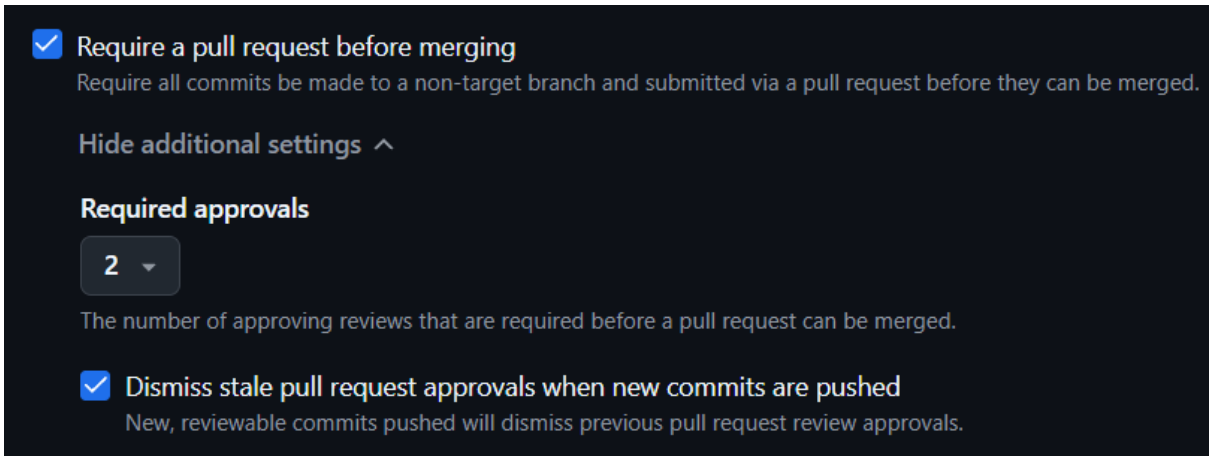
```
Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ touch text.txt

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git add text.txt

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git commit -m "Created file text.txt"
[main 372f9cf] Created file text.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 text.txt

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   text.txt

no changes added to commit (use "git add" and/or "git commit -a")

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git add text.txt

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git commit -m "Edited file text.txt"
[main acbcae5] Edited file text.txt
 1 file changed, 1 insertion(+)

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git rm text.txt
rm 'text.txt'

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git commit -m "Deleted file text.txt"
[main c0d9c90] Deleted file text.txt
 1 file changed, 1 deletion(-)
 delete mode 100644 text.txt

Reuven@DESKTOP-PPBCLCJ MINGW64 ~/Desktop/ספואבד/DevOps Project (main)
$ git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 731 bytes | 731.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/reuven-itzhakov/DevOps-Project.git
   6f7a039..c0d9c90  main -> main
```

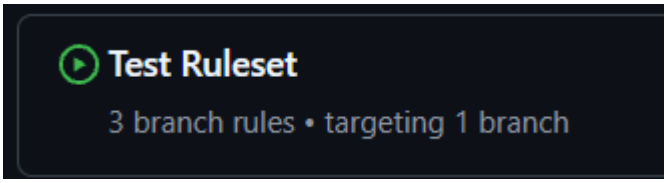
2. הוסיפו חברי צוות ו-Ruleset

הוספנו חברי צוות כ-Collaborators במאגר:



הגדרנו "Branch protection rules" כך שלא ניתן לדחוף קוד ישירות ל-Master
הגדרנו חובה לפתוח PR עבור כל שינוי, עם דרישה לאישור של שני חברי צוות לפחות לפני המיזוג.





3. מה זה Dependabot?

Dependabot זה כלי אוטומטי של GitHub שסורק את הספריות והתלויות בפרויקט.
צריך אותו כדי לזהות פרצות אבטחה או גרסאות מיושנות של קוד חיצוני שאנו משתמשים בו.
הוא פועל ברקע באופן קבוע ושולח התראות או פותח Pull Requests אוטומטיים לעדכון גרסאות.
הוא חוסך זמן בתחזוקה ומונע פריצות לאתר/אפליקציה שנובעות מחורי אבטחה בספריות צד-שלישי.

הרצאה 4

יצרנו את הסקריפט הבא ושמרנו אותו כ-backup_script.sh

```bash
#!/bin/bash

# 1. Define variables
SOURCE_DIR=~/personal
# Create a timestamp in format: YYYYMMDDHHMMSS
TIMESTAMP=$(date +%Y%m%d%H%M%S)
DEST_DIR=~/backup_archive_$TIMESTAMP

# Check if the source directory exists before starting
if [ ! -d "$SOURCE_DIR" ]; then
  echo "Error: Source directory $SOURCE_DIR does not exist."
  exit 1
fi

# 2. Perform backup (recursive copy)
echo "Copying $SOURCE_DIR to $DEST_DIR..."
cp -r "$SOURCE_DIR" "$DEST_DIR"

# 3. Rotation: keep only the 3 most recent backups
# Explanation of the command:
# ls -dt: lists all directories matching backup_archive_* sorted by time (newest first)
# tail -n +4: skips the first 3 lines (the newest backups) and passes the rest (older ones)
# xargs -r rm -rf: takes the list of old backups and removes them (-r avoids errors if the list is empty)

ls -dt ~/backup_archive_* | tail -n +4 | xargs -r rm -rf

# 4. Show final result (as in the image you sent)
echo "Done. Current backups list:"
ls -ld ~/backup_archive_*
```

הסבר על הפקודות החשובות

date +%Y%m%d%H%M%S: הפקודה הזו מייצרת את המחרוזת (למשל 20241118055424) שדרושה לשם התיקייה.

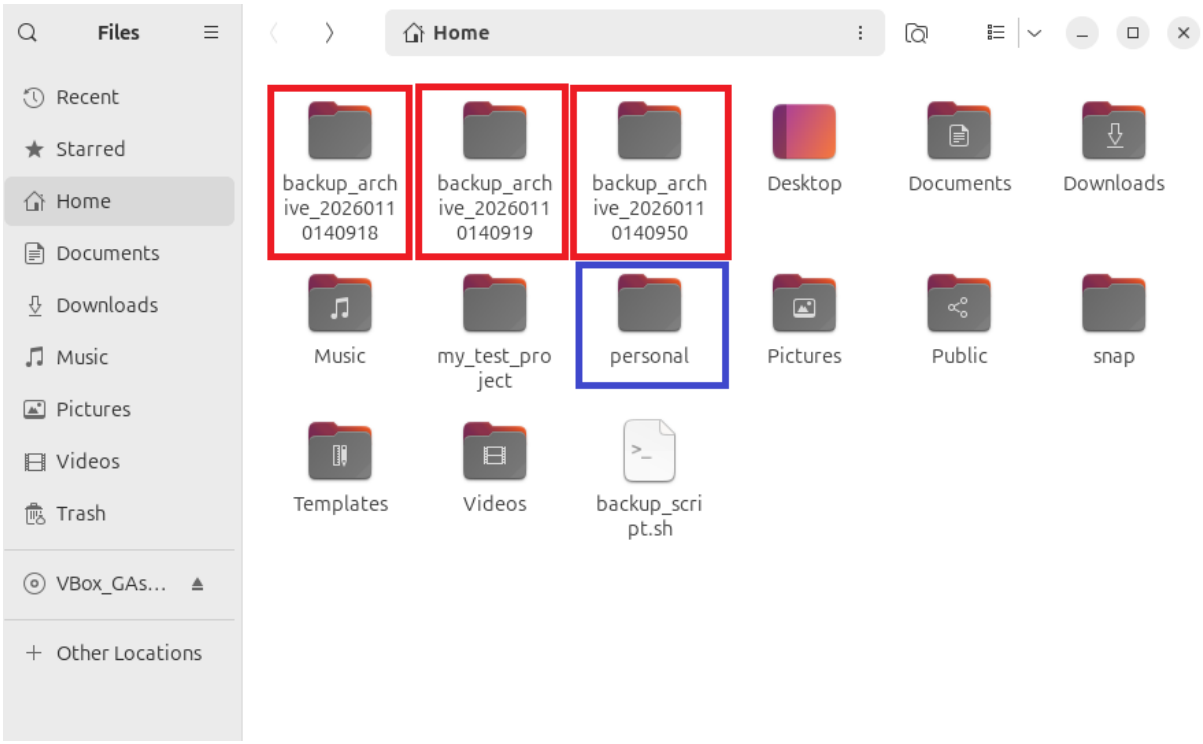
cp -r: מעתיק תיקייה על כל תכולתה (recursive).

ls -dt ... | tail ... | xargs ...: זהו "Pipeline" שמנהל את המחיקה:
אנחנו מבקשים מהמערכת: "תן לי את רשימת הגיבויים מסודרים מהחדש לישן".
"תתעלם מ-3 הראשונים (הכי חדשים)".
"את כל מה שנשאר (הישנים) – תמחק".

לאחר הרצת הסקריפט נקבל את הפלט הבא:

```
reuven@Reuven-Ubuntu:~$ ./backup_script.sh
Copying /home/reuven/personal to /home/reuven/backup_archive_20260110140950...
Done. Current backups list:
drwxrwxr-x 2 reuven reuven 4096 Jan 10 14:09 /home/reuven/backup_archive_20260110140918
drwxrwxr-x 2 reuven reuven 4096 Jan 10 14:09 /home/reuven/backup_archive_20260110140919
drwxrwxr-x 2 reuven reuven 4096 Jan 10 14:09 /home/reuven/backup_archive_20260110140950
```

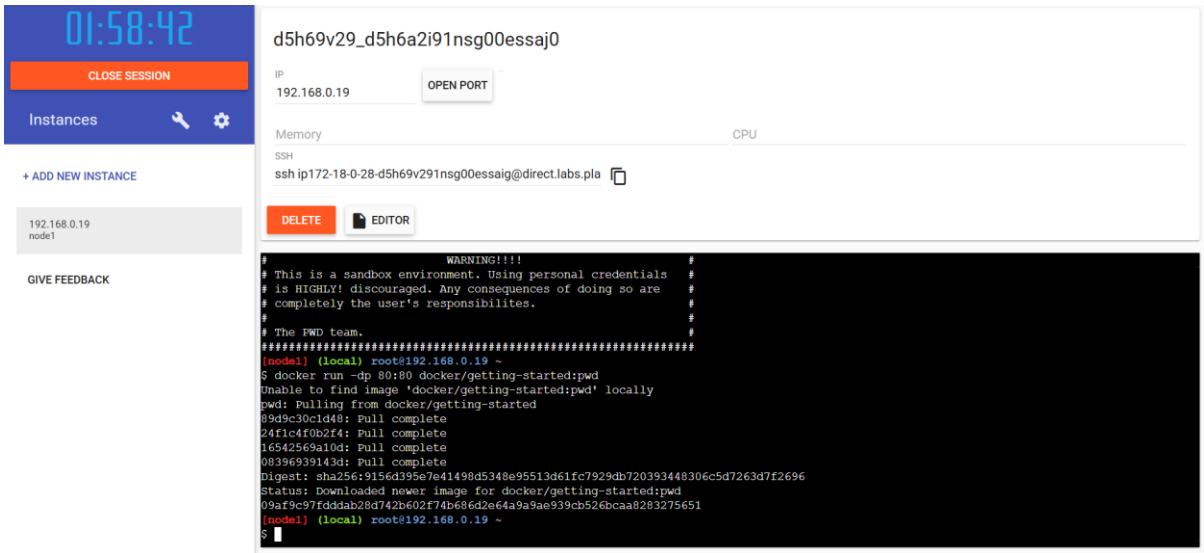
ואת הקבצים הבאים אשר מכילים את תכולת תיקיית personal:



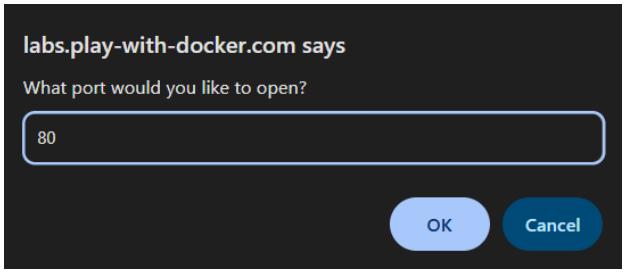
הערה: לאחר כל הרצה של הסקריפט נקבל תיקייה חדשה עם תאריך מתאים ותכולה מותאמת לתכולה של תיקיית personal, במידה ויש יותר מ-3 תיקיות, התיקייה הישנה ביותר תמחק ותוחלף באחת עדכנית.

הרצאה 5

פתחנו את האתר, יצרנו Instance, וכתבנו את הפקודה:

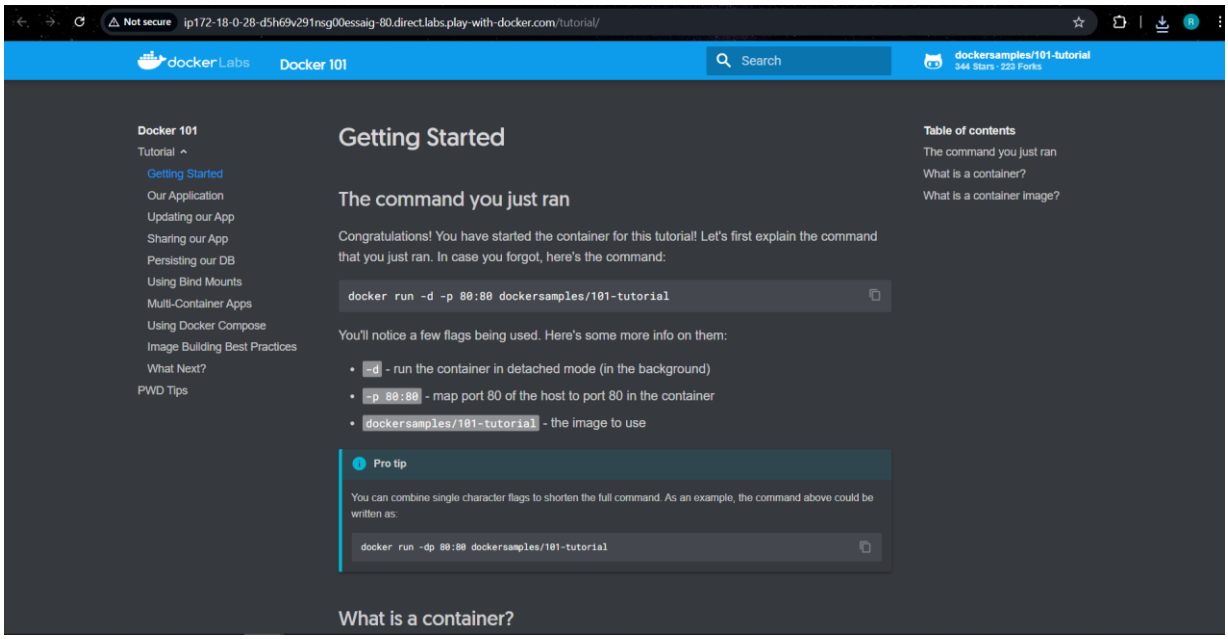


פתחנו פורט 80:

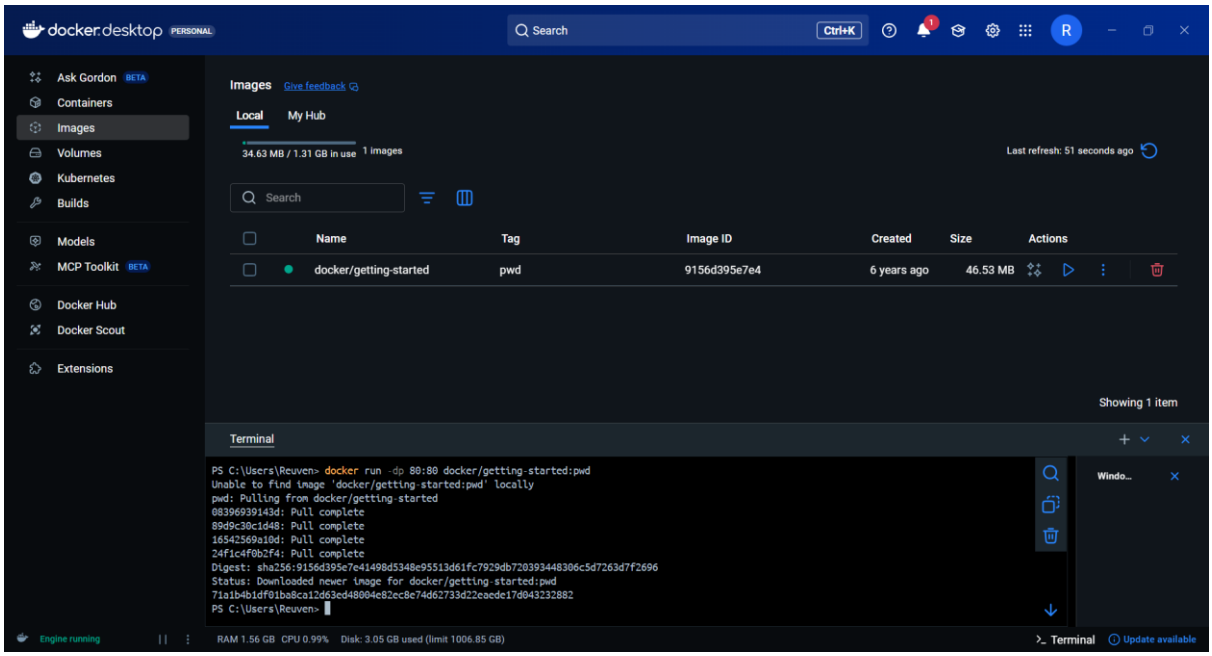


הגענו לאתר הבא:

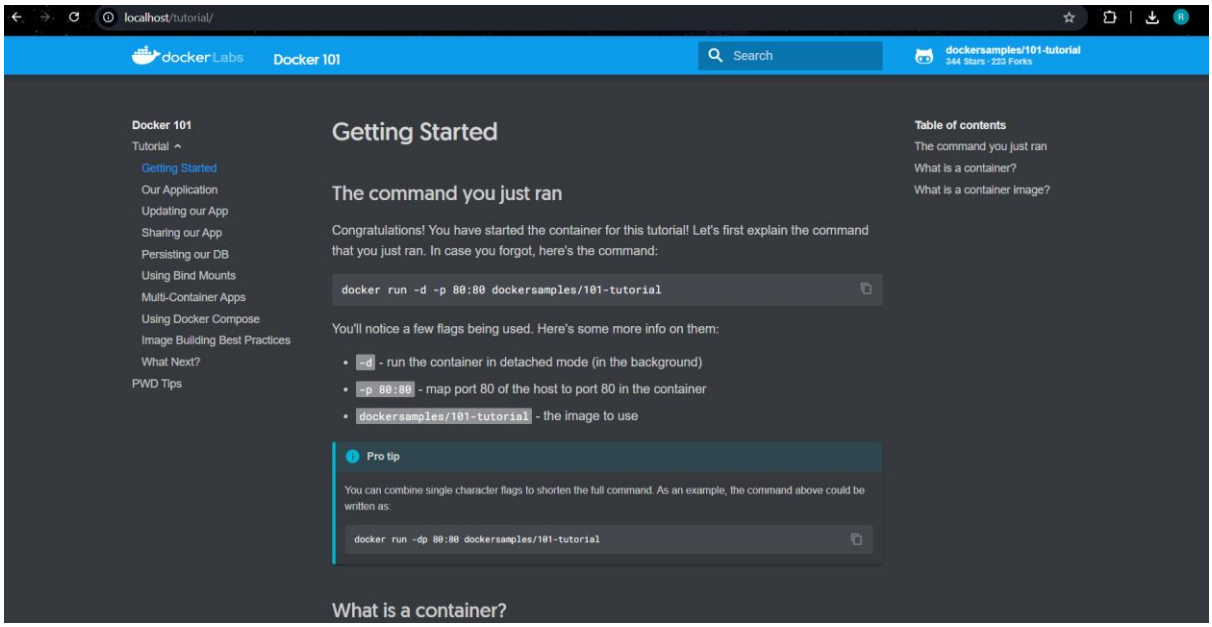
http://ip172-18-0-28-d5h69v291nsg00essaig-80.direct.labs.play-with-/docker.com/tutorial



פתחנו Docker לוקלית, ורשמנו שם את הפקודה:



פתחנו את http://localhost/



יצרנו קובץ app.py:

```
# app.py
print("Hello, Docker!")
```

יצרנו קובץ Dockerfile:

```
# Use python:3.10-slim as the base image
FROM python:3.10-slim

# (Optional but recommended) Set working directory inside the container
WORKDIR /app

# Copy app.py into the container (to the /app directory)
COPY app.py .

# Set the command to run python app.py by default
CMD ["python", "app.py"]
```

נשתמש בפקודה:

docker build -t hello-docker .

```
C:\Users\Reuven\Desktop\[illegible]\DevOps Project>docker build -t hello-docker .
[+] Building 10.3s (9/9) FINISHED                                                                    docker:desktop-linux
 => [internal] load build definition from Dockerfile                                                                 0.1s
 => => transferring dockerfile: 345B                                                                                 0.0s
 => [internal] load metadata for docker.io/library/python:3.10-slim                                                  2.4s
 => [auth] library/python:pull token for registry-1.docker.io                                                        0.0s
 => [internal] load .dockerignore                                                                                    0.1s
 => => transferring context: 2B                                                                                      0.0s
 => [1/3] FROM docker.io/library/python:3.10-slim@sha256:7b68a5fa7cf0d20b4cedb1dc9a134fdd394fe27edbc4c2519756c91d  3.9s
 => => resolve docker.io/library/python:3.10-slim@sha256:7b68a5fa7cf0d20b4cedb1dc9a134fdd394fe27edbc4c2519756c91d  0.1s
 => => sha256:8715e552fa1374bdde269437d9a1c607c817289c2ebbceb9ed9ab1aa9ca86763 1.29MB / 1.29MB                       0.5s
 => => sha256:7da4424a113245eb185ea22f2512eceb36f80ca1d0547c64b117f28495d3c3e5 250B / 250B                           0.5s
 => => sha256:9c27bc7ba63d1ac690daefc68302197d3ab9a91fc5c0e19f447cd57eda92d87c 13.82MB / 13.82MB                     1.3s
 => => sha256:02d7611c4eae219af91448a4720bdba036575d3bc0356cfe12774af85daa6aff 29.78MB / 29.78MB                     2.2s
 => => extracting sha256:02d7611c4eae219af91448a4720bdba036575d3bc0356cfe12774af85daa6aff                            0.7s
 => => extracting sha256:8715e552fa1374bdde269437d9a1c607c817289c2ebbceb9ed9ab1aa9ca86763                            0.1s
 => => extracting sha256:9c27bc7ba63d1ac690daefc68302197d3ab9a91fc5c0e19f447cd57eda92d87c                            0.5s
 => => extracting sha256:7da4424a113245eb185ea22f2512eceb36f80ca1d0547c64b117f28495d3c3e5                            0.0s
 => [internal] load build context                                                                                    0.2s
 => => transferring context: 66B                                                                                     0.0s
 => [2/3] WORKDIR /app                                                                                               2.5s
 => [3/3] COPY app.py .                                                                                              0.1s
 => exporting to image                                                                                               0.8s
 => => exporting layers                                                                                              0.4s
 => => exporting manifest sha256:0fd198b833ea584d46dc60f5794fd92248d83b9cfdeb3dd83d69d356eca0ec74                    0.0s
 => => exporting config sha256:0b6a04bc25e42294d4738308fcff45fa628a0b53e92dc771ace3e76782833191                      0.0s
 => => exporting attestation manifest sha256:8001306f2d7ec2b54281391e0f7a1c4441fc13ac03ed1aed8f47303faaaf7aec        0.1s
 => => exporting manifest list sha256:19a0f63c9ea3cc870f946fcca5b6aee66a83f7dba739094e6ae2cb7a0fdc974a               0.0s
 => => naming to docker.io/library/hello-docker:latest                                                               0.0s
 => => unpacking to docker.io/library/hello-docker:latest                                                            0.1s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/t7jb4teefleqv5rp30nyo3y5g
```

לבסוף נריץ באמצעות docker run hello-docker:

```
C:\Users\Reuven\Desktop\[illegible]\DevOps Project>docker run hello-docker
Hello, Docker!
```

הרצאה 6

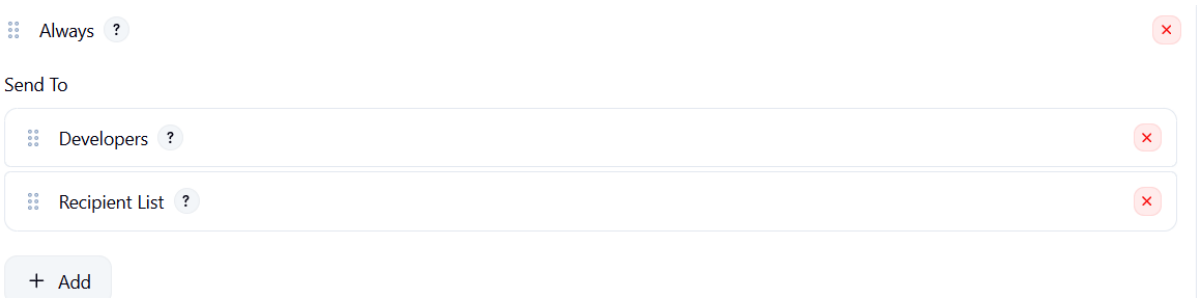
הרצנו docker של jenkins ונכנסנו ל-localhost 8080. שם הדפסנו welcome message והתקנו python:

```
+ echo hello world!
hello world!
+ set -e
+ python --version
Python 3.13.5
```

ביצוע טסט לקוד:

```
+ pytest -v
============================= test session starts ==============================
platform linux -- Python 3.13.5, pytest-9.0.2, pluggy-1.6.0 -- /var/jenkins_home/workspace/Shalev Jenkins/.venv/bin/python
cachedir: .pytest_cache
rootdir: /var/jenkins_home/workspace/Shalev Jenkins
collecting ... collected 1 item

tests/test_basic.py::test_example PASSED                                 [100%]

============================== 1 passed in 0.00s ===============================
Finished: SUCCESS
```

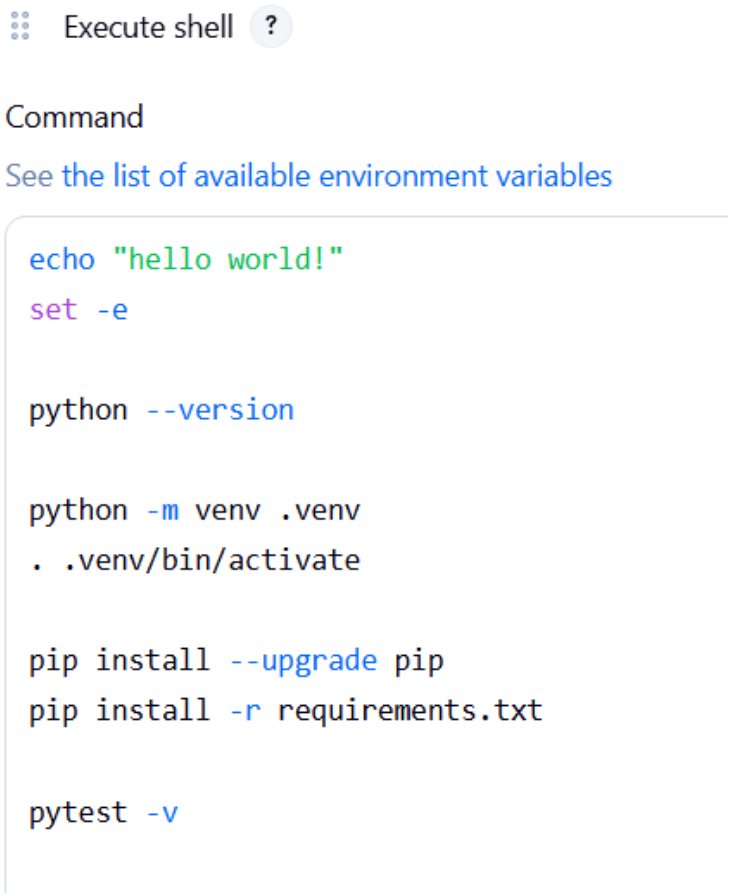
הגדרת jenkis שישלח מייל עבור כל הרצה:



שליחת email:

```
Email was triggered for: Always
Sending email for trigger: Always
Sending email to: coshalev2@gmail.com
Finished: SUCCESS
```

כתיבת המשימה שתורץ: Shell code.



דוגמא מהמצגת:

```
+ echo starting the build...
starting the build...
+ echo building the project...
building the project...
+ sleep 5
+ echo build completed successfully!
build completed successfully!
```

הרצאה 7

שקף 31:

3 nodes :On master node

יצרנו מאסטר נוד והגדרנו אותו, בנוסף יצרנו עוד 2 workers וביצענו להם join למאסטר.

```
[node1 ~]$ kubectl get nodes
NAME    STATUS   ROLES           AGE     VERSION
node1   Ready    control-plane   6m21s   v1.27.2
node2   Ready    <none>          4m3s    v1.27.2
node3   Ready    <none>          3m50s   v1.27.2
[node1 ~]$ kubectl get services
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   6m27s
[node1 ~]$ kubectl get pods
No resources found in default namespace.
```

Minikube:

התקנו minikube והרצנו אותו באמצעות docker.

```
C:\Users\cosha>kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   12d   v1.34.0

C:\Users\cosha>minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured


C:\Users\cosha>kubectl version
Client Version: v1.34.1
Kustomize Version: v5.7.1
Server Version: v1.34.0
```

זוהי רשימה של כל הפודים בתוך הnamespace הנ"ל ב k8s cluster.

```
C:\Users\cosha>kubectl get po -A
NAMESPACE              NAME                                         READY   STATUS    RESTARTS        AGE
default                hello-minikube-bbcb89c6c-dwzr8               1/1     Running   1 (2m31s ago)   12d
kube-system            coredns-66bc5c9577-8qkm7                     1/1     Running   1 (2m31s ago)   12d
kube-system            etcd-minikube                                1/1     Running   1 (2m31s ago)   12d
kube-system            kube-apiserver-minikube                      1/1     Running   1 (2m31s ago)   12d
kube-system            kube-controller-manager-minikube             1/1     Running   1 (2m31s ago)   12d
kube-system            kube-proxy-r8d2p                             1/1     Running   1 (2m31s ago)   12d
kube-system            kube-scheduler-minikube                      1/1     Running   1 (2m31s ago)   12d
kube-system            storage-provisioner                          1/1     Running   3 (103s ago)    12d
kubernetes-dashboard   dashboard-metrics-scraper-77bf4d6c4c-xkwdm   1/1     Running   1 (2m31s ago)   12d
kubernetes-dashboard   kubernetes-dashboard-855c9754f9-t9hzz        1/1     Running   2 (99s ago)     12d
```

Run service:

```
C:\Users\cosha>kubectl get services hello-minikube
NAME             TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
hello-minikube   NodePort   10.111.227.226   <none>        8080:31635/TCP   12d
```

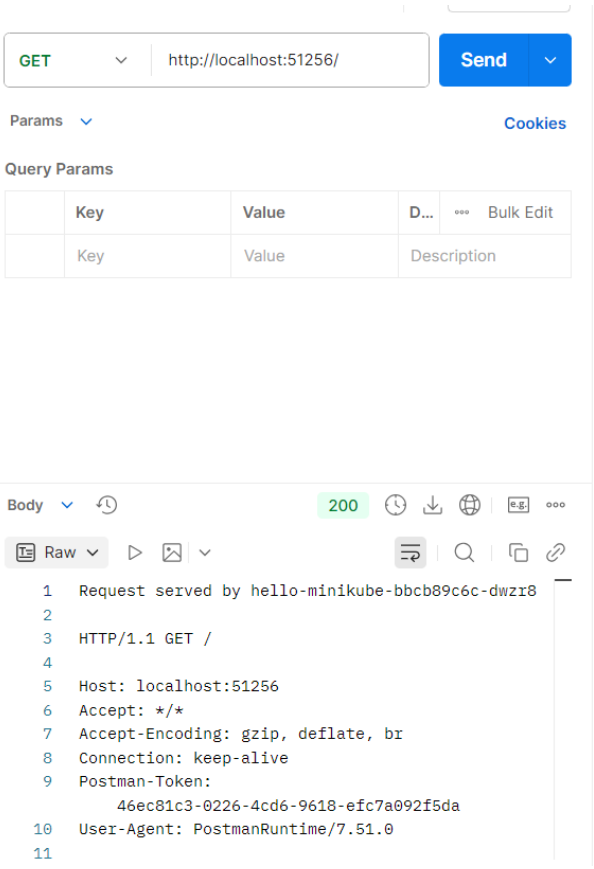
Access Service:

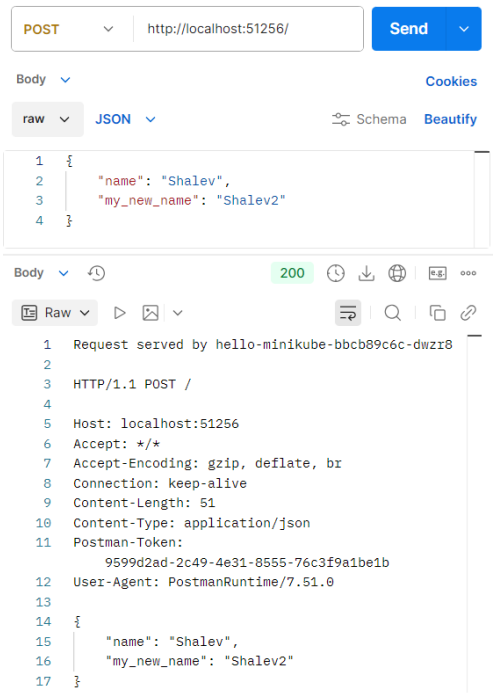
הרצנו dashboard ולאחר מכן ניגשנו אל הweb הנל:

```
Request served by hello-minikube-bbcb89c6c-dwzr8

HTTP/1.1 GET /

Host: 127.0.0.1:51256
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: he-IL,he;q=0.9,en-US;q=0.8,en;q=0.7
Connection: keep-alive
Sec-Ch-Ua: "Google Chrome";v="143", "Chromium";v="143", "Not A(Brand";v="24"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: none
Sec-Fetch-User: ?1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36
```

Postman

התקנו postman וירצנו פקודות get & post וניגשנו אליהן באמצעות הport שיצרנו מהסעיף הקודם בweb.





Cluster management

ניהול הminikube באמצעות מגוון פקודות:

```
C:\Users\cosha>minikube pause
* Pausing node minikube ...
* Paused 18 containers in: kube-system, kubernetes-dashboard, storage-gluster, istio-operator

C:\Users\cosha>minikube unpause
* Unpausing node minikube ...
* Unpaused 18 containers in: kube-system, kubernetes-dashboard, storage-gluster, istio-operator

C:\Users\cosha>minikube config set memory 9001
! These changes will take effect upon a minikube delete and then a minikube start
```

```
C:\Users\cosha>minikube addons list
```

ADDON NAME	PROFILE	STATUS	MAINTAINER
ambassador	minikube	disabled	3rd party (Ambassador)
amd-gpu-device-plugin	minikube	disabled	3rd party (AMD)
auto-pause	minikube	disabled	minikube
cloud-spanner	minikube	disabled	Google
csi-hostpath-driver	minikube	disabled	Kubernetes
dashboard	minikube	enabled ✅	Kubernetes
default-storageclass	minikube	enabled ✅	Kubernetes
efk	minikube	disabled	3rd party (Elastic)
freshpod	minikube	disabled	Google
gcp-auth	minikube	disabled	Google
gvisor	minikube	disabled	minikube
headlamp	minikube	disabled	3rd party (kinvolk.io)
inaccel	minikube	disabled	3rd party (InAccel [info@inaccel.com])
ingress	minikube	disabled	Kubernetes
ingress-dns	minikube	disabled	minikube
inspektor-gadget	minikube	disabled	3rd party (inspektor-gadget.io)
istio	minikube	disabled	3rd party (Istio)
istio-provisioner	minikube	disabled	3rd party (Istio)
kong	minikube	disabled	3rd party (Kong HQ)
kubeflow	minikube	disabled	3rd party
kubetail	minikube	disabled	3rd party (kubetail.com)
kubevirt	minikube	disabled	3rd party (KubeVirt)
logviewer	minikube	disabled	3rd party (unknown)
metallb	minikube	disabled	3rd party (MetalLB)
metrics-server	minikube	disabled	Kubernetes
nvidia-device-plugin	minikube	disabled	3rd party (NVIDIA)
nvidia-driver-installer	minikube	disabled	3rd party (NVIDIA)
nvidia-gpu-device-plugin	minikube	disabled	3rd party (NVIDIA)
olm	minikube	disabled	3rd party (Operator Framework)
pod-security-policy	minikube	disabled	3rd party (unknown)
portainer	minikube	disabled	3rd party (Portainer.io)
registry	minikube	disabled	minikube
registry-aliases	minikube	disabled	3rd party (unknown)
registry-creds	minikube	disabled	3rd party (UPMC Enterprises)
storage-provisioner	minikube	enabled ✅	minikube
storage-provisioner-gluster	minikube	disabled	3rd party (Gluster)
storage-provisioner-rancher	minikube	disabled	3rd party (Rancher)
volcano	minikube	disabled	third-party (volcano)
volumesnapshots	minikube	disabled	Kubernetes
yakd	minikube	disabled	3rd party (marcnuri.com)

```
C:\Users\cosha>minikube start -p aged --kubernetes-version=v1.16.1
* [aged] minikube v1.37.0 on Microsoft Windows 11 Home 10.0.26200.7462 Build 26200.7462
! Specified Kubernetes version 1.16.1 is less than the oldest supported version: v1.28.0. Use `minikube config defaults
kubernetes-version` for details.
! You can force an unsupported Kubernetes version via the --force flag

X Exiting due to K8S_OLD_UNSUPPORTED: Kubernetes 1.16.1 is not supported by this release of minikube


C:\Users\cosha>minikube stop
* Stopping node "minikube"  ...
* Powering off "minikube" via SSH ...
* 1 node stopped.
```