DOCKER COMPOSE

Learning Solution

WHAT IS DOCKER COMPOSE

Compose is a tool for defining and running multi-container Docker applications. With Compose, you use a YAML file to configure your application’s services. Then, with a single command, you create and start all the services from your configuration

HOW TO INSTALL DOCKER COMPOSE

- #curl -L "https://github.com/docker/compose/releases/download/1.26.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
- #chmod +x /usr/local/bin/docker-compose
- #docker-compose --version

YAML FILE

YAML Ain't Markup Language is a data serialization language that matches user’s expectations about data. It designed to be human friendly and works perfectly with other programming languages.

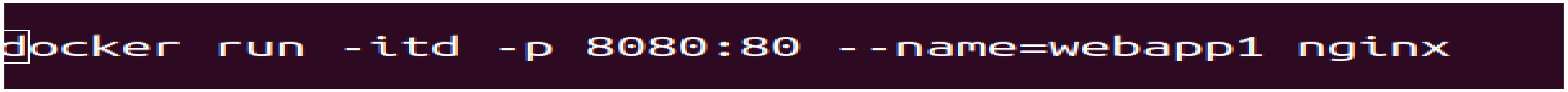
EXAMPLE OF YAML FILE

- YAML FORMAT IS **key=value**.
- Example:

```
--- #start of file
    -  colors:
        - red
        - orange
        - blue
    -  cars:
        - ford
        - mazda
```

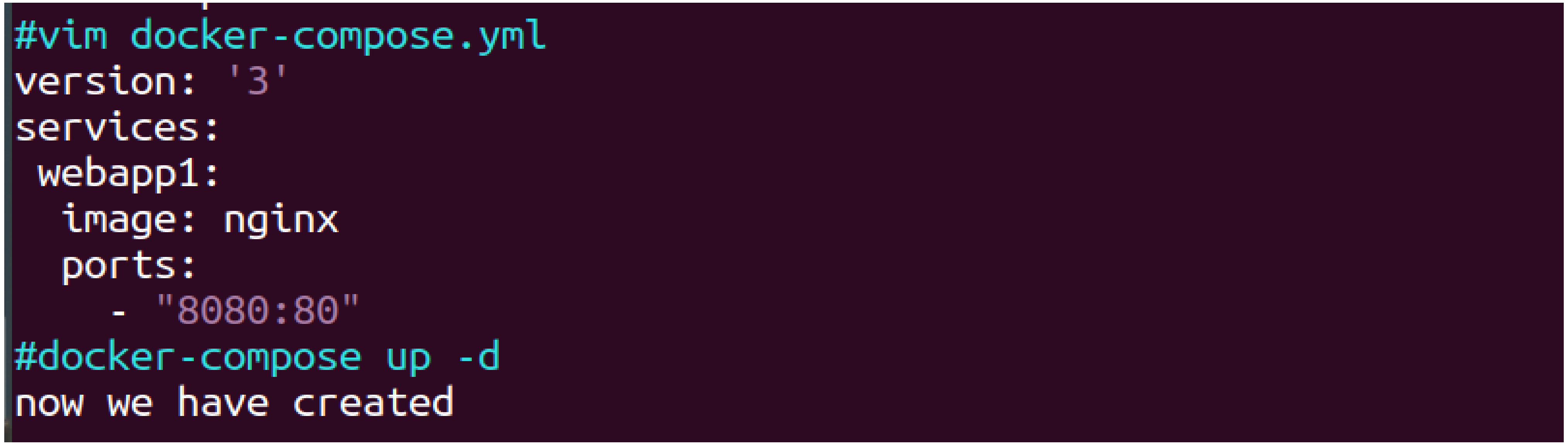
CREATE DOCKER COMPOSE

- MANUALLY



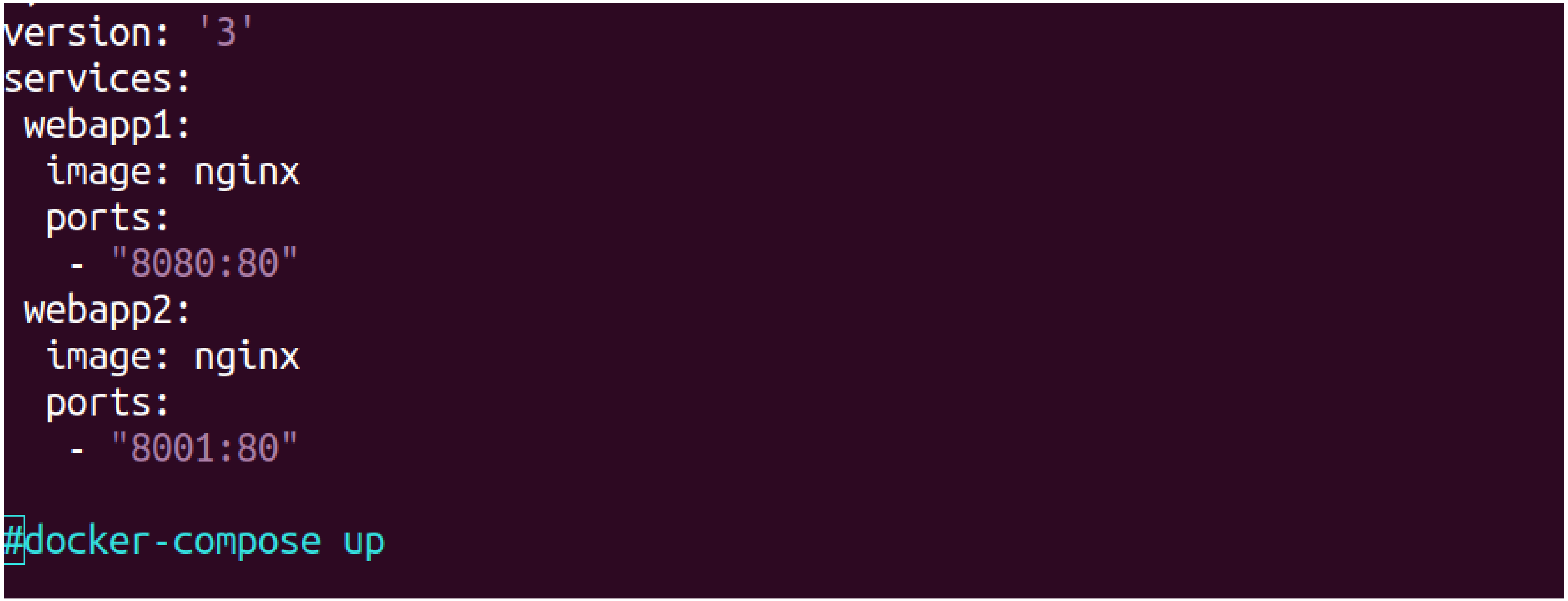
```
docker run -itd -p 8080:80 --name=webapp1 nginx
```

- WITH DOCKER-COMPOSE



```
#vim docker-compose.yml
version: '3'
services:
 webapp1:
  image: nginx
  ports:
    - "8080:80"
#docker-compose up -d
now we have created
```

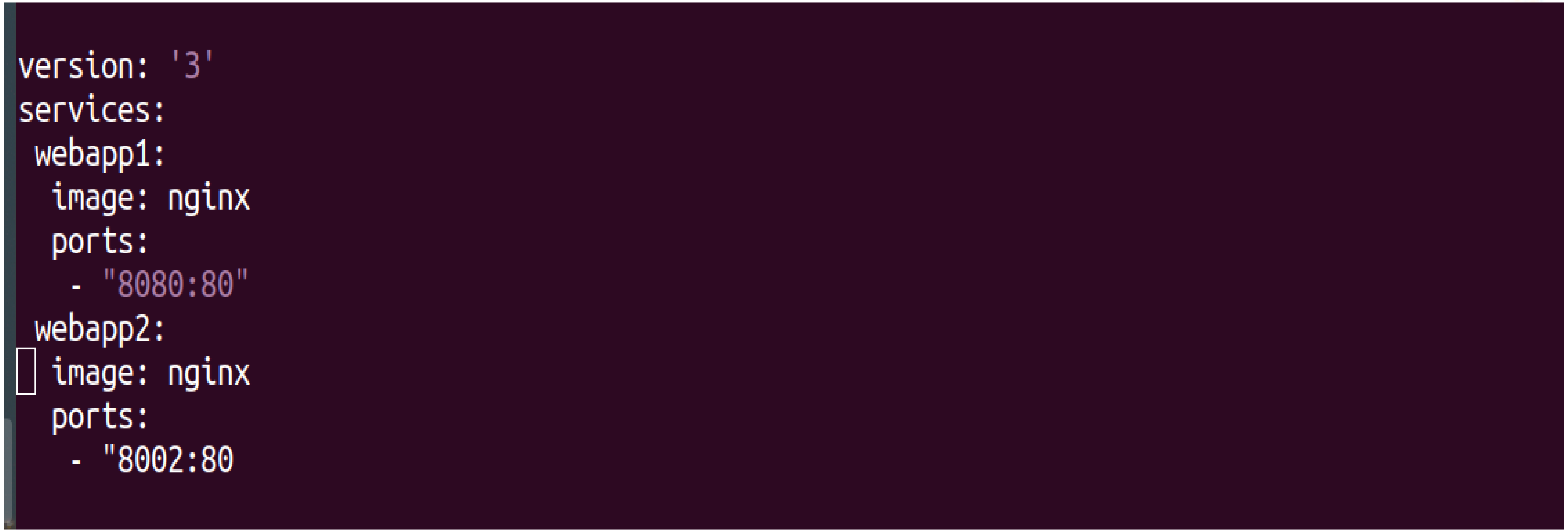
TO CREATE MULTIPLE CONTAINERS



```
version: '3'
services:
 webapp1:
  image: nginx
  ports:
   - "8080:80"
 webapp2:
  image: nginx
  ports:
   - "8001:80"

#docker-compose up
```

RECREATE OR DO ANY CHANGES

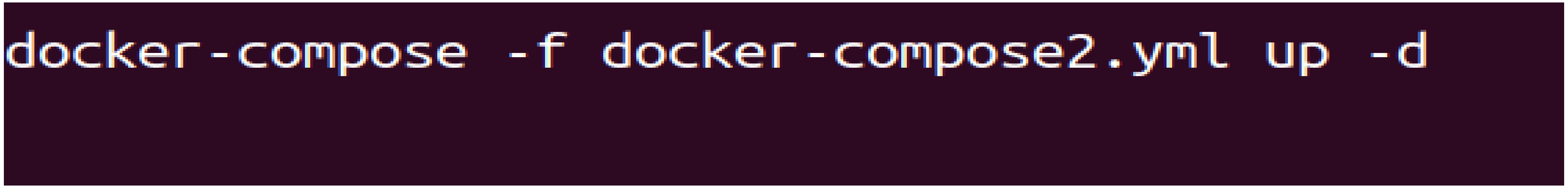


```
version: '3'
services:
 webapp1:
  image: nginx
  ports:
   - "8080:80"
 webapp2:
  image: nginx
  ports:
   - "8002:80
```

In the above example it will change the port number if existing any container is created

DOCKER COMPOSE

- By default docker-compose only find **docker-compose.yml** file but if we want to change the name



```
docker-compose -f docker-compose2.yml up -d
```

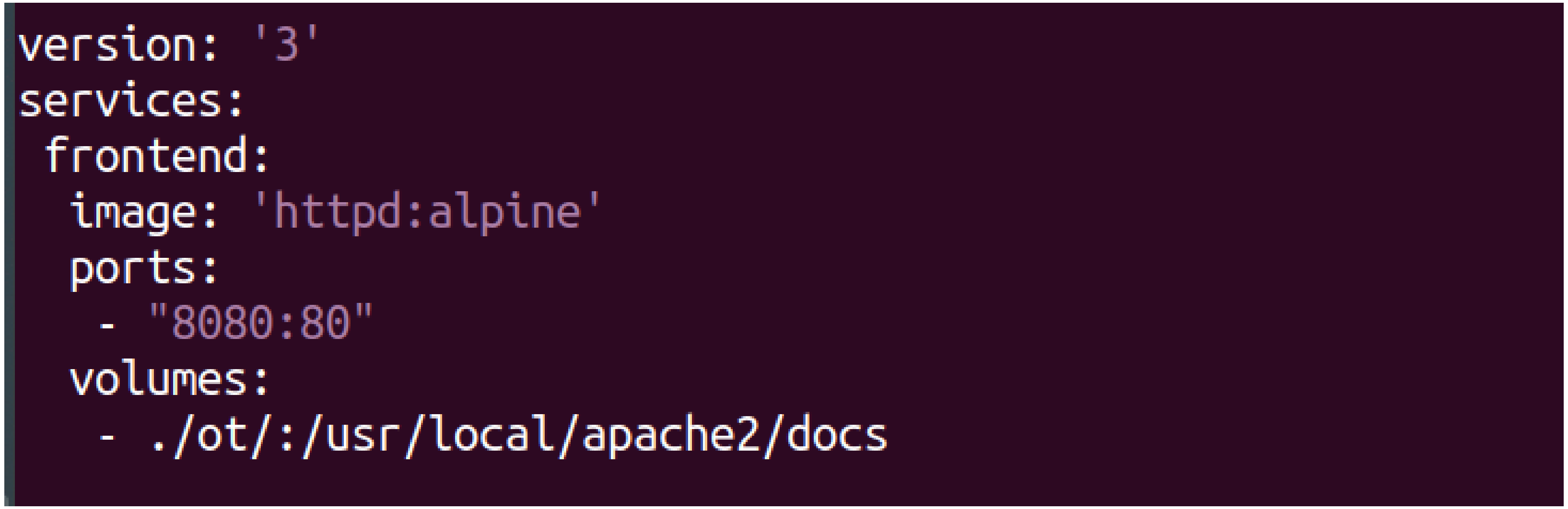
DOCKER COMPOSE COMMANDS

- #docker-compose up --no-start -> it will create container but will not run
- #docker-compose rm -> to delete stopped container
- #docker-compose start -> to run the container
- #docker-compose stop -> to stop the container
- #docker-compose ps -> to show all container running process

DOCKER COMPOSE COMMANDS

- #docker-compose pause -> to pause
- #docker-compose unpause -> to unpause
- #docker-compose kill -> to kill
- #docker-compose port webapp1 80 -> to check the port
- #docker-compose exec webapp1 ls -> to run ls command in webapp1 container

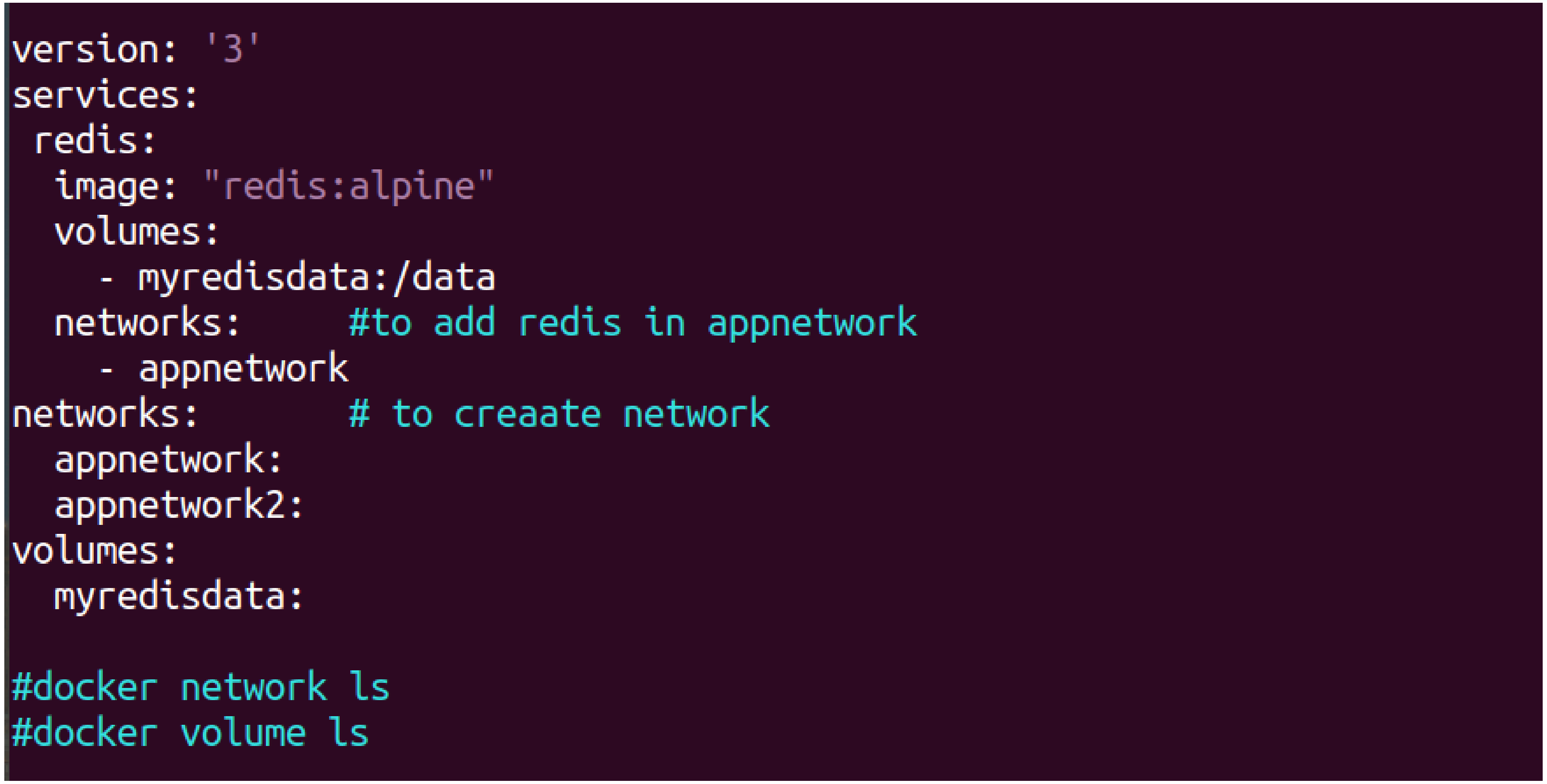
USE BIND MOUNT IN DOCKER COMPOSE



```
version: '3'
services:
 frontend:
  image: 'httpd:alpine'
  ports:
   - "8080:80"
  volumes:
   - ./ot/:/usr/local/apache2/docs
```

Learning Solution

CREATE NETWORK AND VOLUME

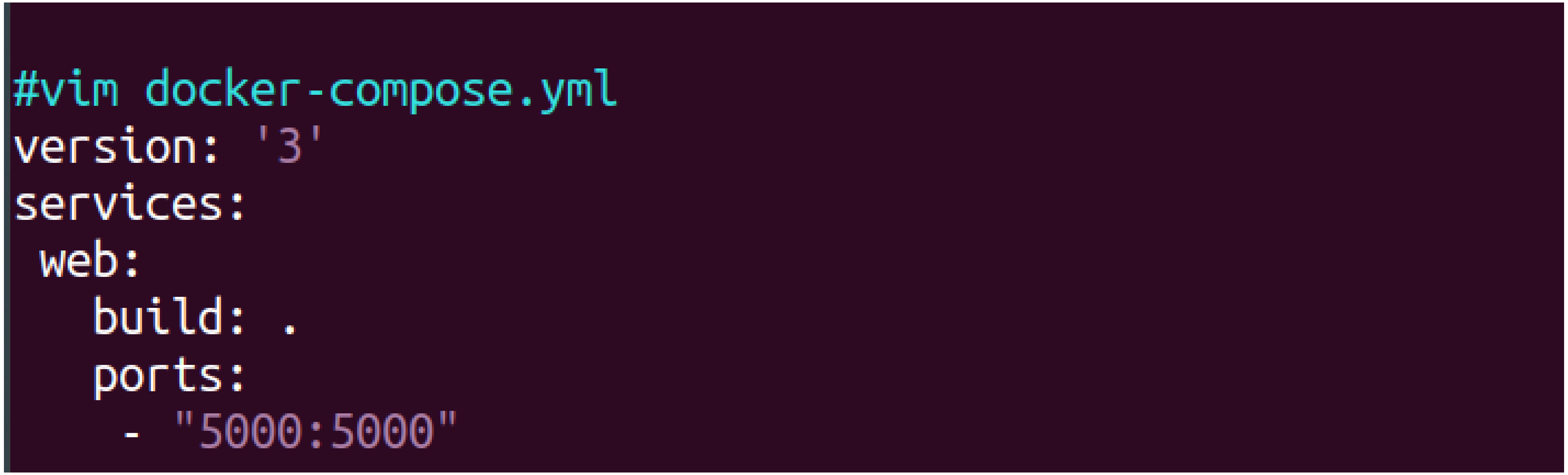


```
version: '3'
services:
 redis:
   image: "redis:alpine"
   volumes:
      - myredisdata:/data
   networks:      #to add redis in appnetwork
      - appnetwork
networks:         # to creaate network
   appnetwork:
   appnetwork2:
volumes:
   myredisdata:

#docker network ls
#docker volume ls
```

Learning Solution

DOCKER FILE IN DOCKER COMPOSE



```
#vim docker-compose.yml
version: '3'
services:
 web:
   build: .
   ports:
    - "5000:5000"
```

Learning Solution