Nama : Alpian Roymundus Siringo-ringo

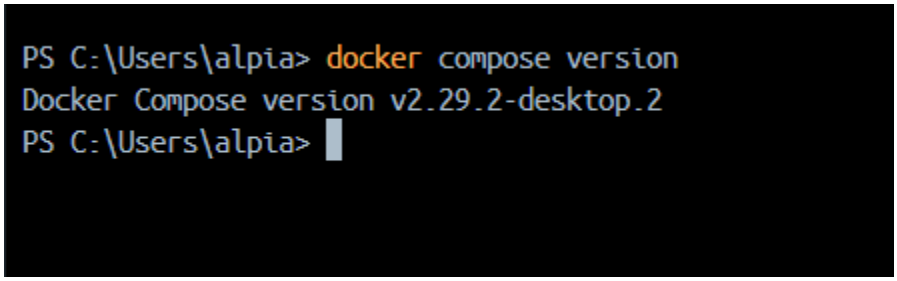
NIM : 11211009

Sistem Terdistribusi A

Docker Compose

1. **Menginstall Docker Compose**

   Docker versi terbaru, Docker Compose sudah tersedia secara otomatis di dalam Docker nya, kita bisa check menggunakan "docker compose version' dan untuk menggunakan Docker compose kita bisa menggunakan perintah "docker compose"

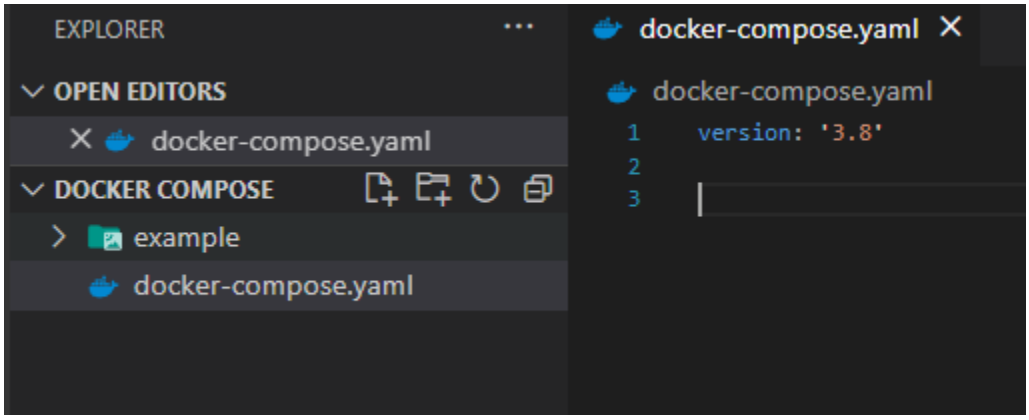


2. **Configuration File**

   Docker Compose menyimpan konfigurasi nya dalam bentuk file YAML:https://yaml.org/

   File YAML mirip JSON, namun lebih sederhana

   Biasanya file konfigurasinya disimpan dalam file bernama docker-compose.yaml



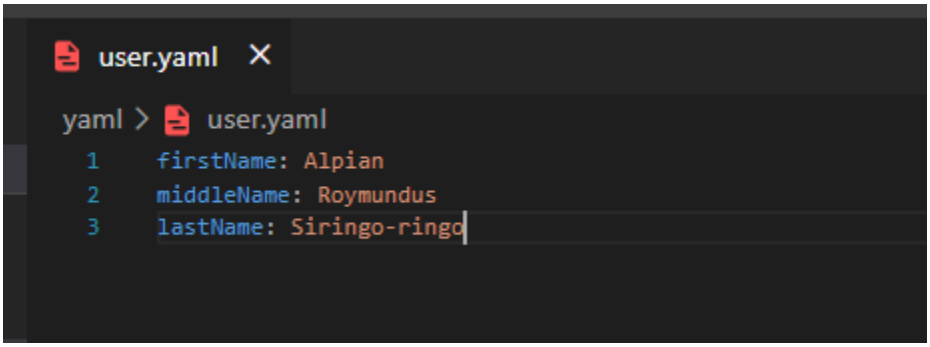
3. **Yaml**

   Yaml adalah sebuah jenis file yang biasa digunakan untuk menyimpan konfigurasi

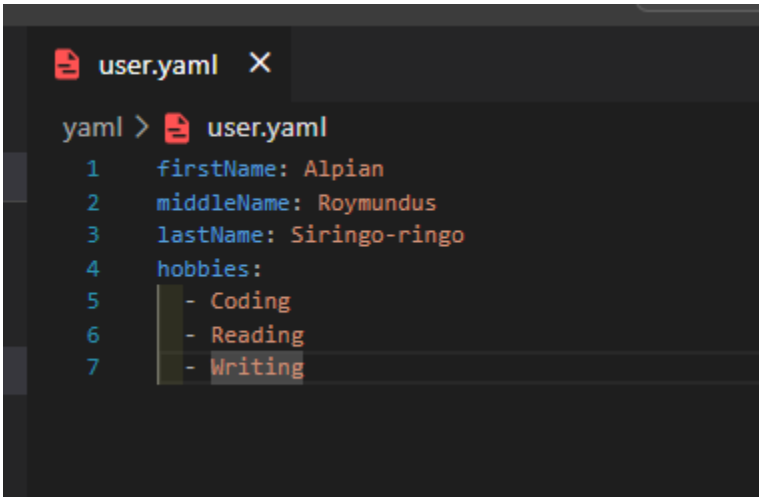
   Yaml mirip seperti JSON, hanya saja tidak menggunakan kurung kurawal

   Yaml akan memiliki attribute dan value

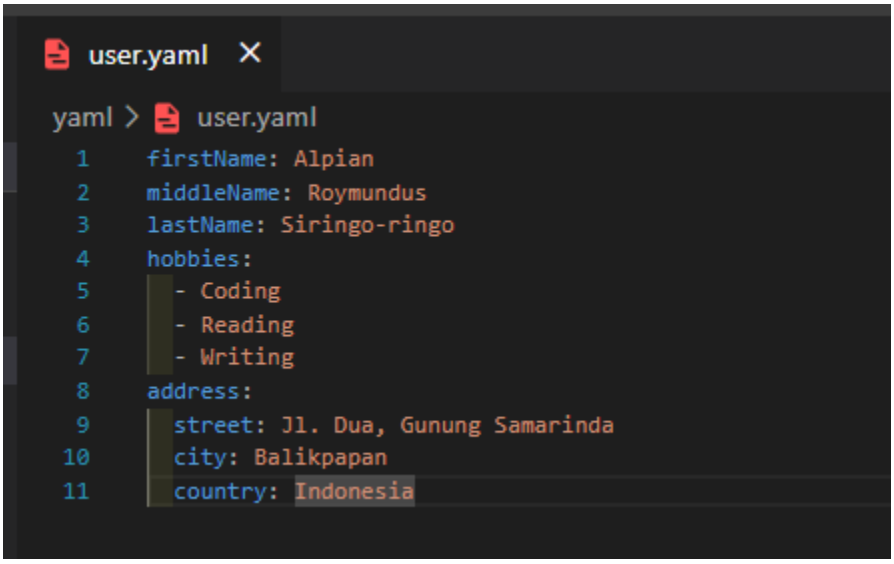
a. **Yaml Attribute**



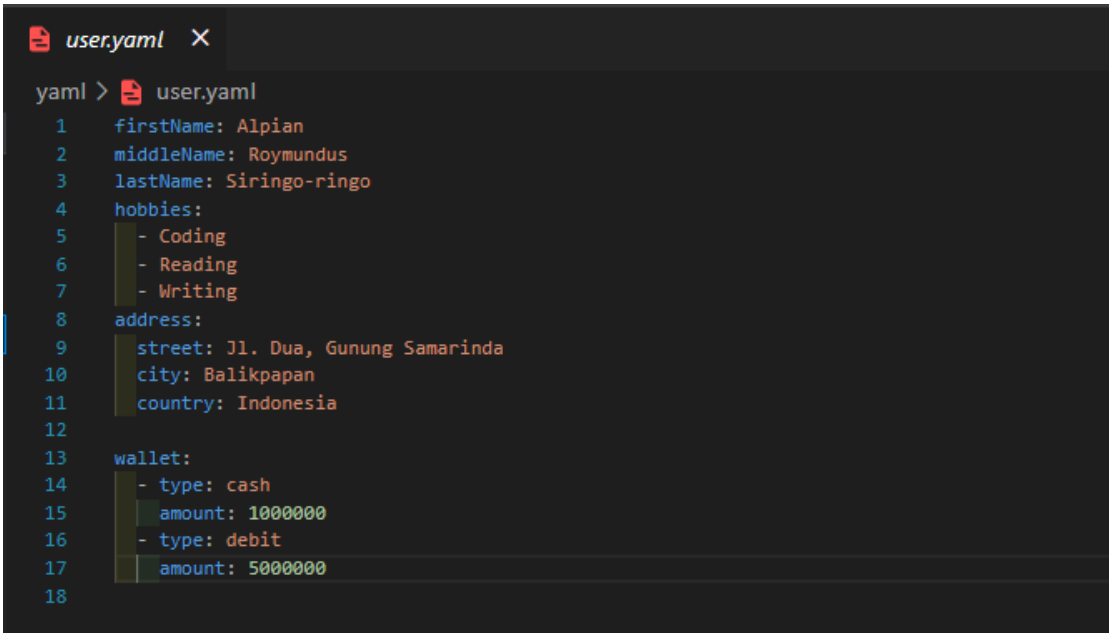
b. **Yaml Array**



c. **Yaml Nested Object**

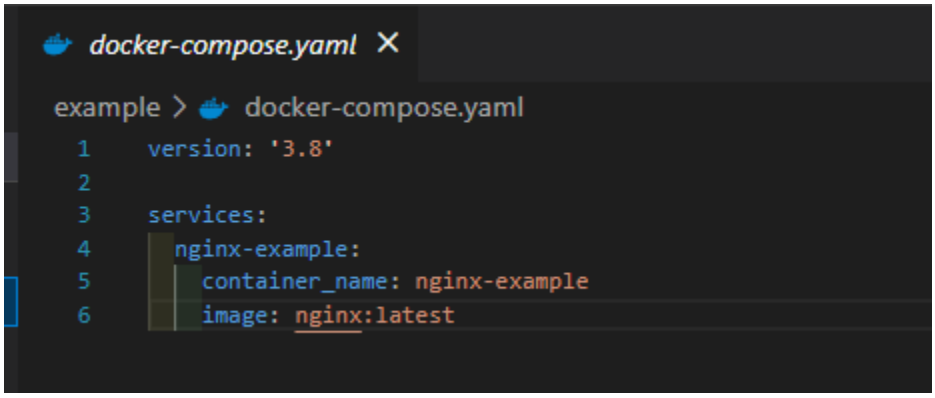


d. Yaml Array Nested Object



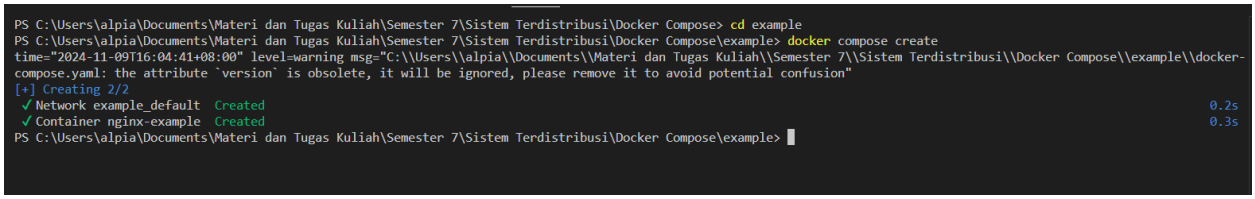
4. **Membuat Container**

Docker kini memungkinkan pembuatan container secara langsung melalui file konfigurasi docker-compose.yaml, tanpa perlu menggunakan perintah docker create. Dengan menambahkan bagian services di dalam file YAML, pengguna dapat mendefinisikan container yang ingin dibuat dengan mudah. Pada bagian services, pengguna dapat menetapkan nama container menggunakan container_name serta menentukan image yang akan digunakan untuk container tersebut. Hal ini menyederhanakan pengaturan dan pengelolaan container dalam satu file konfigurasi, membuatnya lebih terstruktur dan mudah dipelihara.



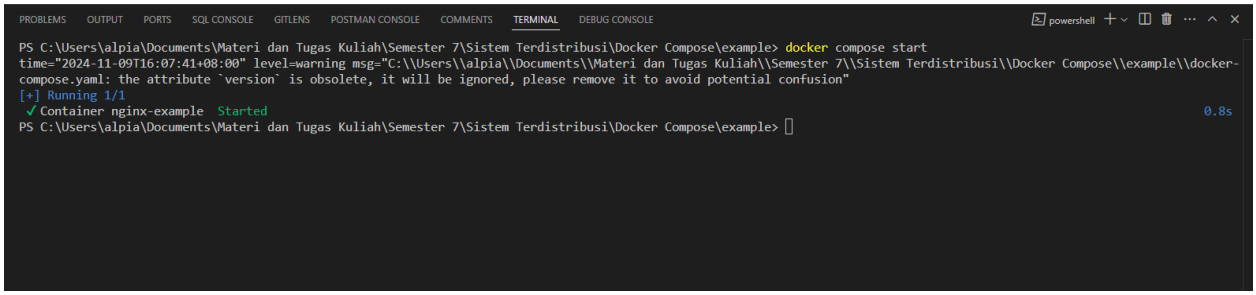
Setelah membuat konfigurasi file, Container tidak langsung jadi, kita harus membuatnya dengan menggunakan Docker Compose, yaitu dengan perintah :

docker compose create



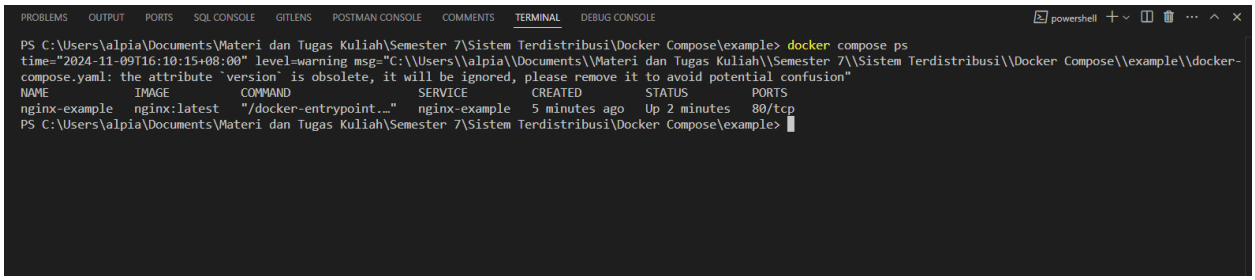
5. **Menjalankan Container**

Setelah membuat container, container tersebut tidak akan berjalan secara otomatis. Untuk menjalankannya, Anda harus melakukannya secara manual, baik dengan perintah `docker container start` atau menggunakan Docker Compose. Jika menggunakan Docker Compose, Anda dapat menjalankan container dengan perintah `docker compose start`. Perintah ini memungkinkan Anda untuk memulai semua container yang telah didefinisikan dalam file `docker-compose.yaml` secara efisien.



6. **Melihat Container**

Untuk melihat container yang berjalan, biasanya kita menggunakan perintah `docker container ls`. Namun, perintah ini akan menampilkan semua container yang ada, baik yang dibuat secara manual maupun melalui Docker Compose. Jika Anda hanya ingin melihat status container yang didefinisikan dalam file konfigurasi Docker Compose, Anda bisa menggunakan perintah `docker compose ps`. Perintah ini hanya menampilkan container yang terkait dengan konfigurasi di file `docker-compose.yaml`, sehingga memudahkan pemantauan container yang dikelola oleh Docker Compose.



7. **Menghentikan Container**

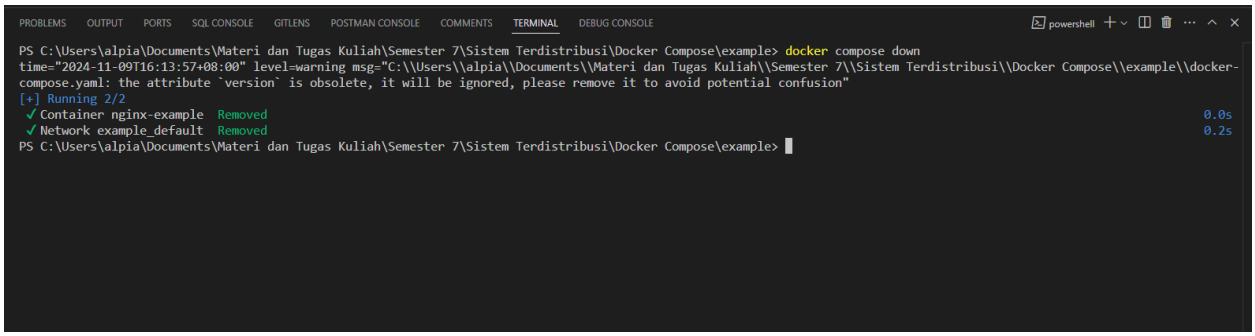
Untuk menghentikan Container, kita bisa menggunakan perintah :

docker compose stop

Menghentikan Container hanya men-stop Container, tidak akan menghapus Container nya.

8. Menghapus Container

Jika Anda sudah tidak memerlukan container yang didefinisikan dalam file konfigurasi Docker Compose, Anda dapat menghapusnya. Penghapusan dapat dilakukan secara manual dengan perintah `docker container rm`, atau lebih praktis menggunakan Docker Compose. Untuk menghapus container beserta semua network dan volume yang terkait, gunakan perintah `docker compose down`. Perintah ini secara otomatis akan menghapus semua container, network, dan volume yang digunakan oleh container tersebut, sehingga membantu menjaga lingkungan tetap bersih dan terorganisir.



9. Project Name

Seperti yang dijelaskan sebelumnya, saat menggunakan Docker Compose, informasi konfigurasi Docker Compose akan disimpan dalam bentuk proyek. Secara default, nama proyek ini diambil dari nama folder tempat file `docker-compose.yaml` berada. Untuk melihat daftar proyek yang sedang berjalan, Anda dapat menggunakan perintah `docker compose ls`. Perintah ini menampilkan semua proyek Docker Compose aktif beserta informasi terkait, memudahkan Anda untuk memantau dan mengelola berbagai proyek

dalam sistem.

```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE                                    powershell

PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\example> docker compose ls
NAME                STATUS              CONFIG FILES
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\example> docker compose create
time="2024-11-09T16:16:35+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\example\\docker-
compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/2
 ✓ Network example_default  Created                                                                                                                    0.1s
 ✓ Container nginx-example  Created                                                                                                                    0.1s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\example> docker compose start
time="2024-11-09T16:16:40+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\example\\docker-
compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
 ✓ Container nginx-example  Started                                                                                                                    0.5s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\example> docker compose ls
NAME                STATUS              CONFIG FILES
example             running(1)          C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\example\docker-compose.yaml
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\example>
```

10. Service

Dalam konfigurasi Docker Compose, container disimpan dalam konfigurasi bernama services

Kita bisa menambahkan satu atau lebih services dalam konfigurasi file nya.

a. Service

```yaml
# docker-compose.yaml
version: '3.8'

services:
  nginx-example:
    container_name: nginx-example
    image: nginx:latest
  mongodb-example:
    container_name: mongodb-example
    image: mongo:latest
```

b. Menjalankan Container

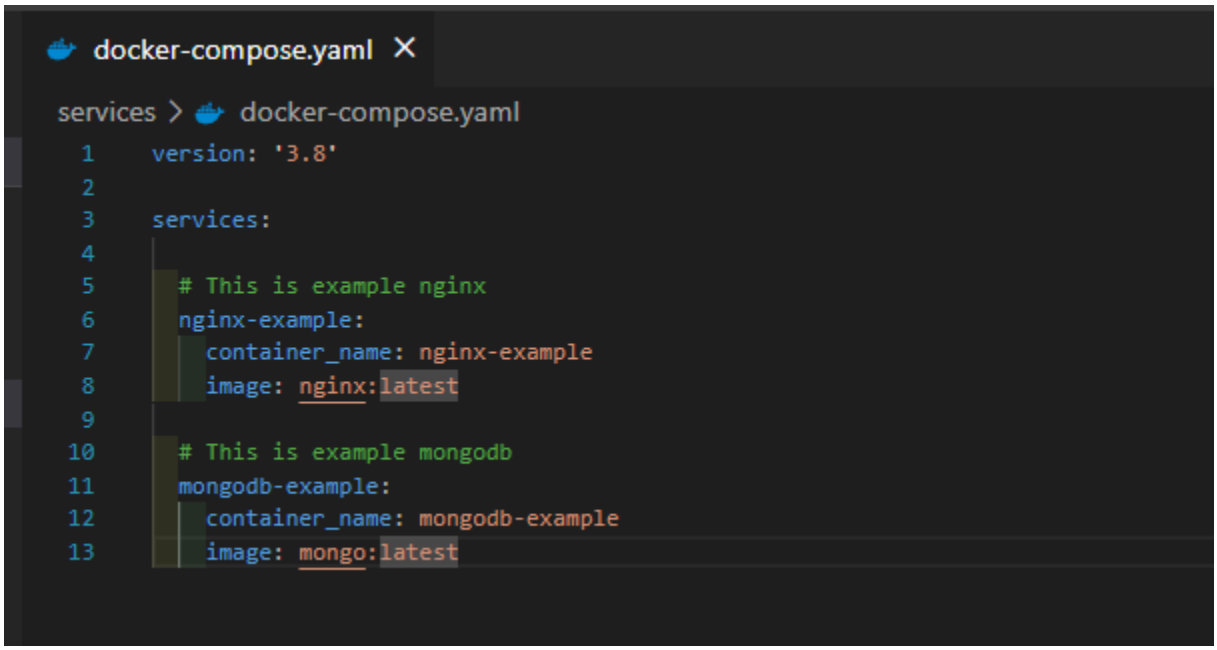
```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE                                    powershell

PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\services> docker compose create
time="2024-11-09T16:23:50+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\services\\docker
-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 3/3
 ✓ Network services_default   Created                                                                                                                  0.1s
 ✓ Container nginx-example    Created                                                                                                                  0.1s
 ✓ Container mongodb-example  Created                                                                                                                  0.2s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\services> docker compose start
time="2024-11-09T16:24:19+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\services\\docker
-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
 ✓ Container mongodb-example  Started                                                                                                                  0.7s
 ✓ Container nginx-example    Started                                                                                                                  0.8s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\services> docker compose ps
time="2024-11-09T16:25:32+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\services\\docker
-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME                IMAGE               COMMAND                  SERVICE             CREATED              STATUS              PORTS
mongodb-example     mongo:latest        "docker-entrypoint.s…"   mongodb-example     About a minute ago   Up About a minute   27017/tcp
nginx-example       nginx:latest        "/docker-entrypoint.…"   nginx-example       About a minute ago   Up About a minute   80/tcp
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\services>
```

11. Komentar

Salah satu keunggunlan menggunakan Yaml dari pada JSON adalah, di Yaml kita bisa menambahkan komentar dengan diawali karakter #

Di JSON kita tidak bisa menambahkan komentar

Komentar secara otomatis akan dihiraukan oleh Docker Compose.



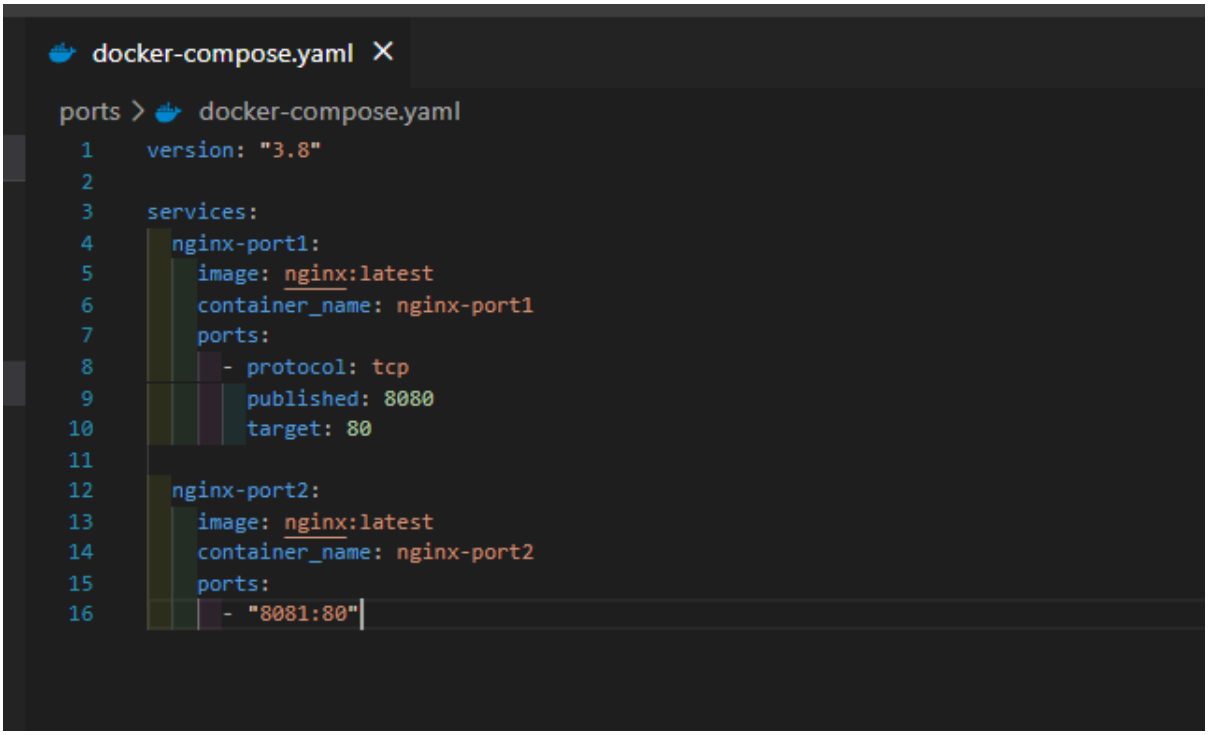
12. Port

Saat membuat Container, kita bisa mengekspose port di Container keluar menggunakan Port Forwarding

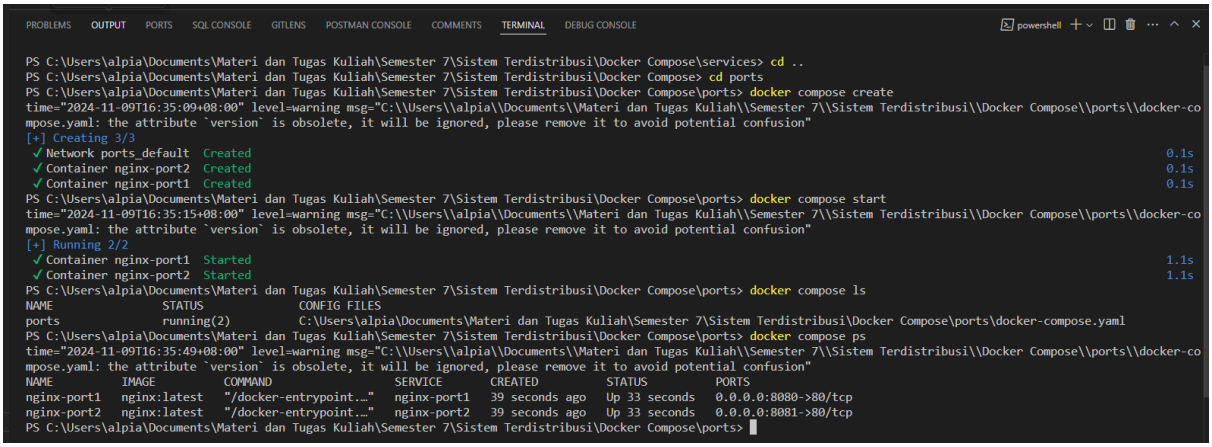
Kita juga bisa melakukan hal tersebut di konfigurasi file Docker Compose dengan menggunakan attribute ports

Attribute ports berisi array object port

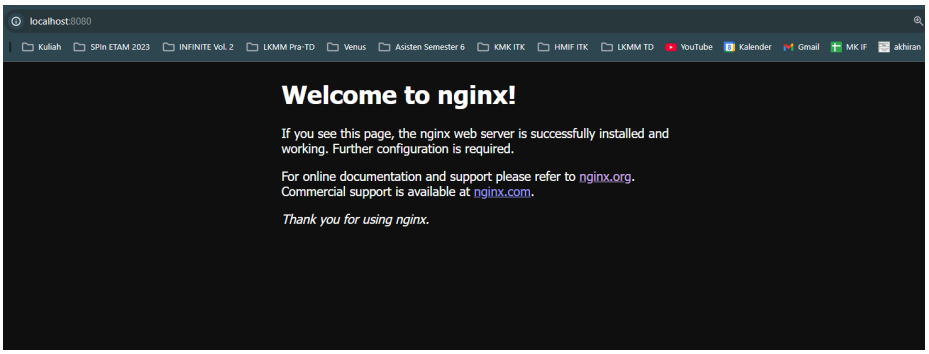
a. Port



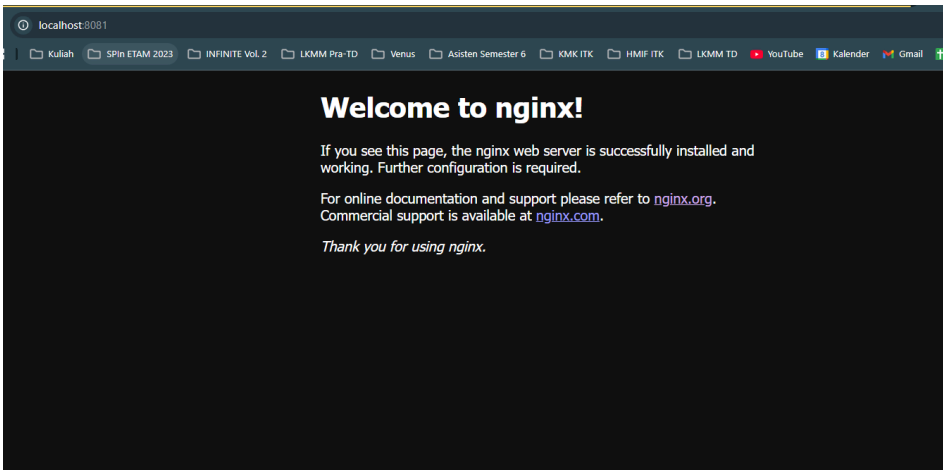
b. Menjalankan Container



Localhost:8080



Localhost:8081

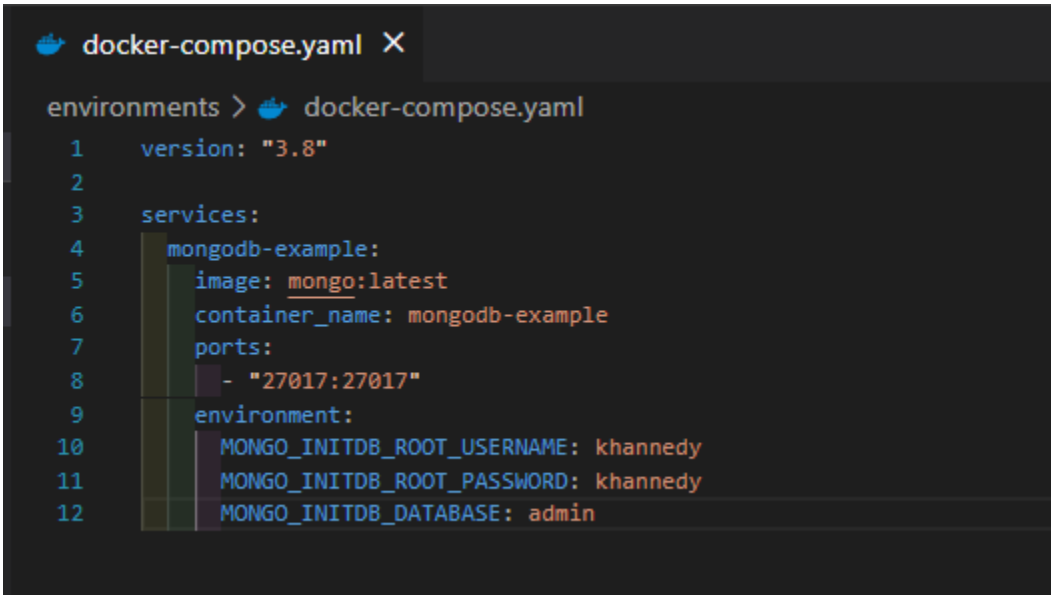


13. Environment Variable

Saat membuat container, kita juga menambahkan environment variable untuk digunakan di dalam container.

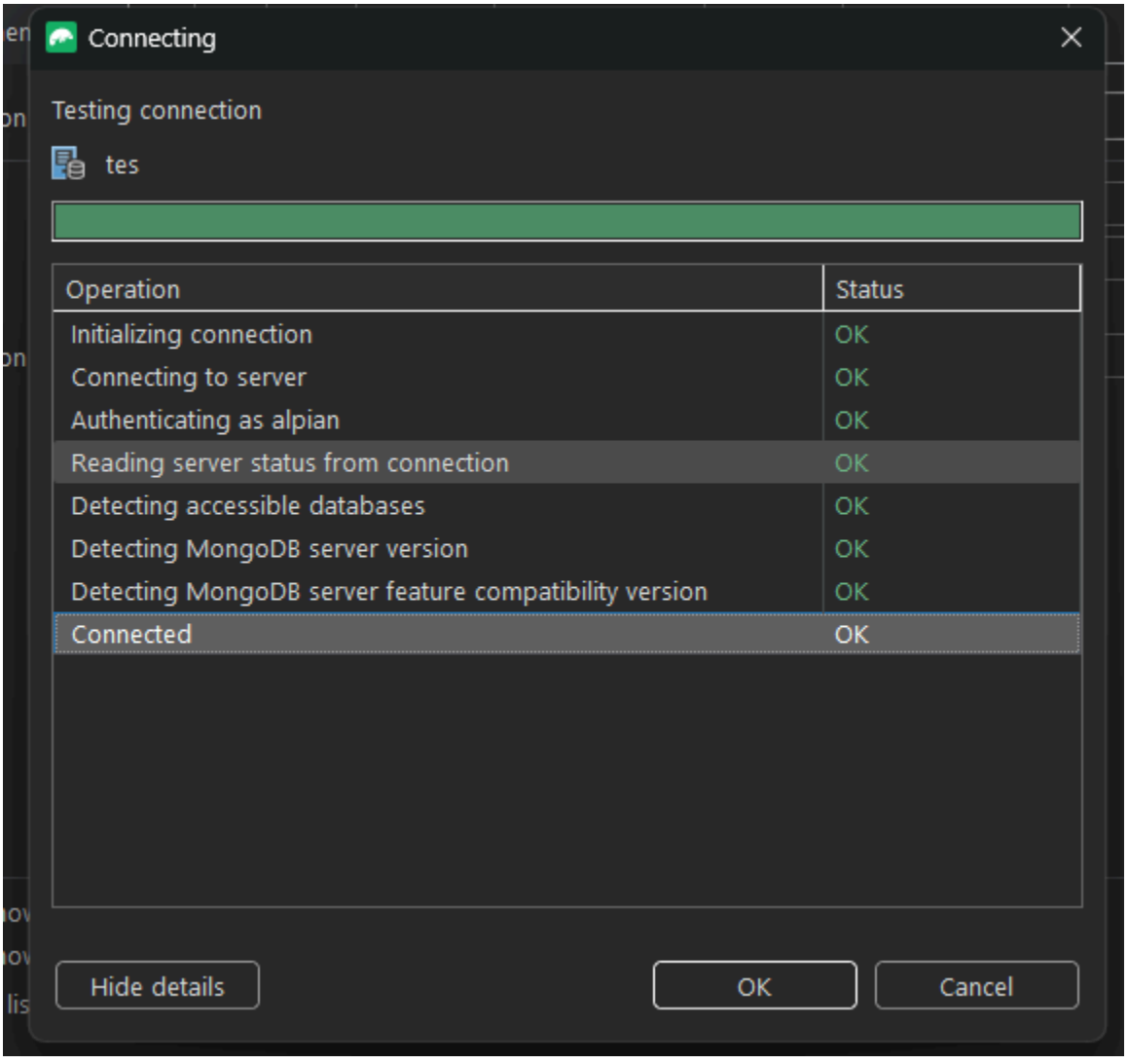
Saat menggunakan konfigurasi file Docker Compose, kita bisa tambahkan environment variable dengan menggunakan attribute environment.

a. Environment Variable



b. Menjalankan Container

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\environments> docker compose create
time="2024-11-09T16:44:45+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\environments\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/2
 ✓ Network environments_default  Created                                                                    0.1s
 ✓ Container mongodb-example     Created                                                                    0.1s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\environments> docker compose start
time="2024-11-09T16:44:55+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\environments\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
 ✓ Container mongodb-example  Started                                                                       0.7s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\environments>
```

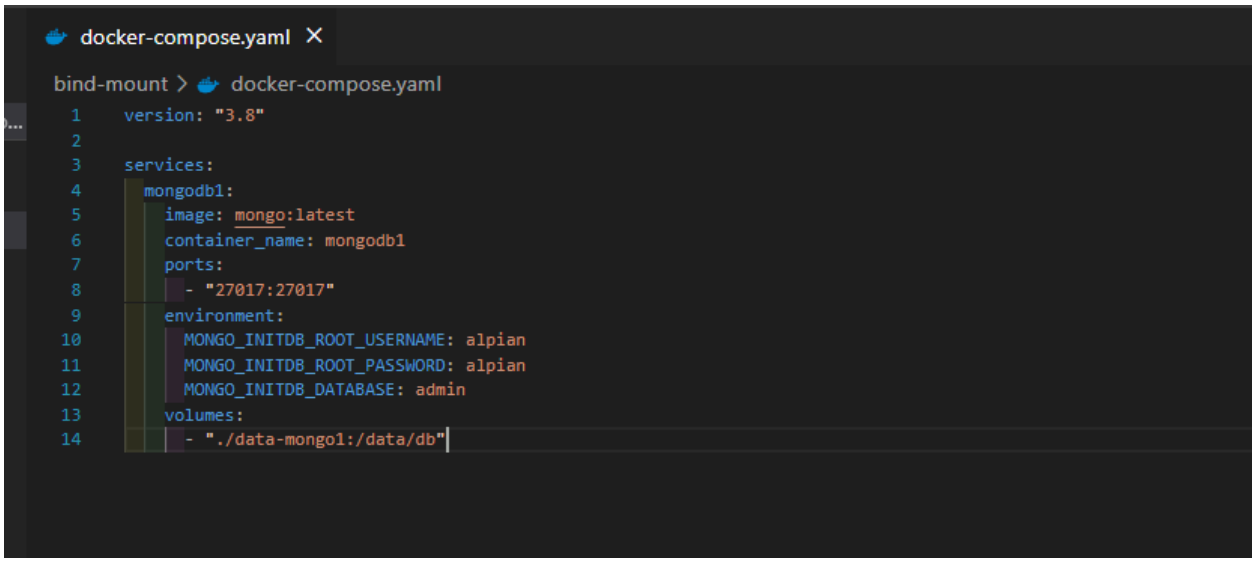


14. Bind Mount

Untuk melakukan bind mount, kita juga bisa lakukan di konfigurasi file Docker Compose

Kita bisa gunakan attribute volumes di services

Kita bisa tambahkan satu atau lebih bind mount jika kita mau.



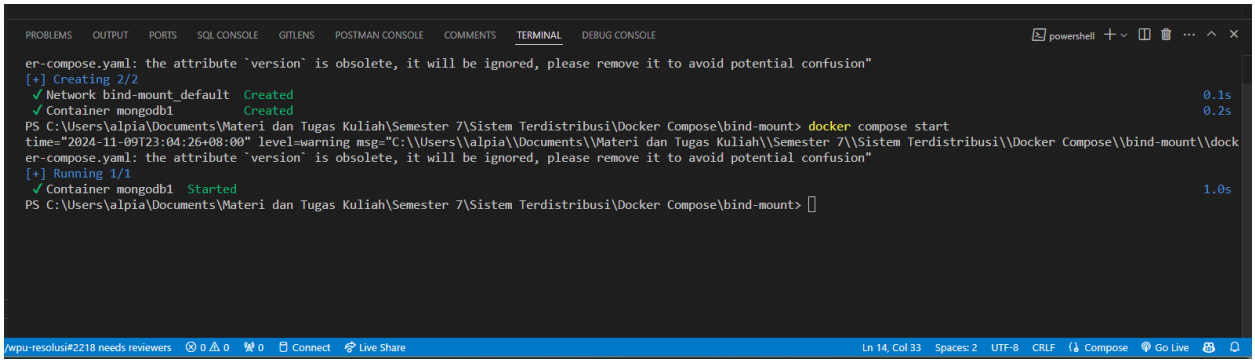
Bind Mount Short Syntax.

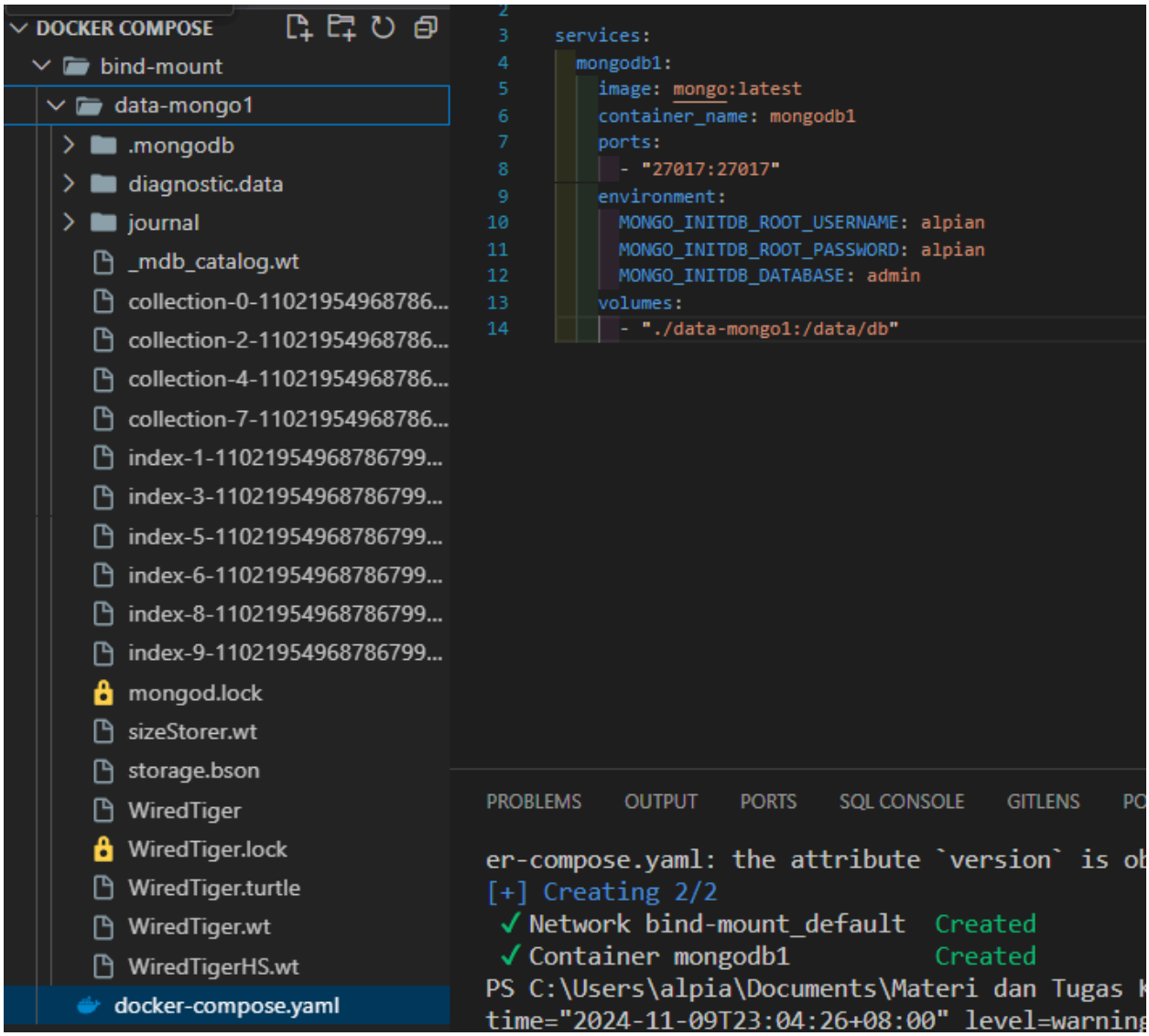
Untuk Bind Mount, kita bisa gunakan short sytax dan long syntax

Untuk short syntax, kita bisa gunakan nilai SOURCE:TARGET:MODE, dimana SOURCE adalah lokasi di host, dan TARGET adalah lokasi di container

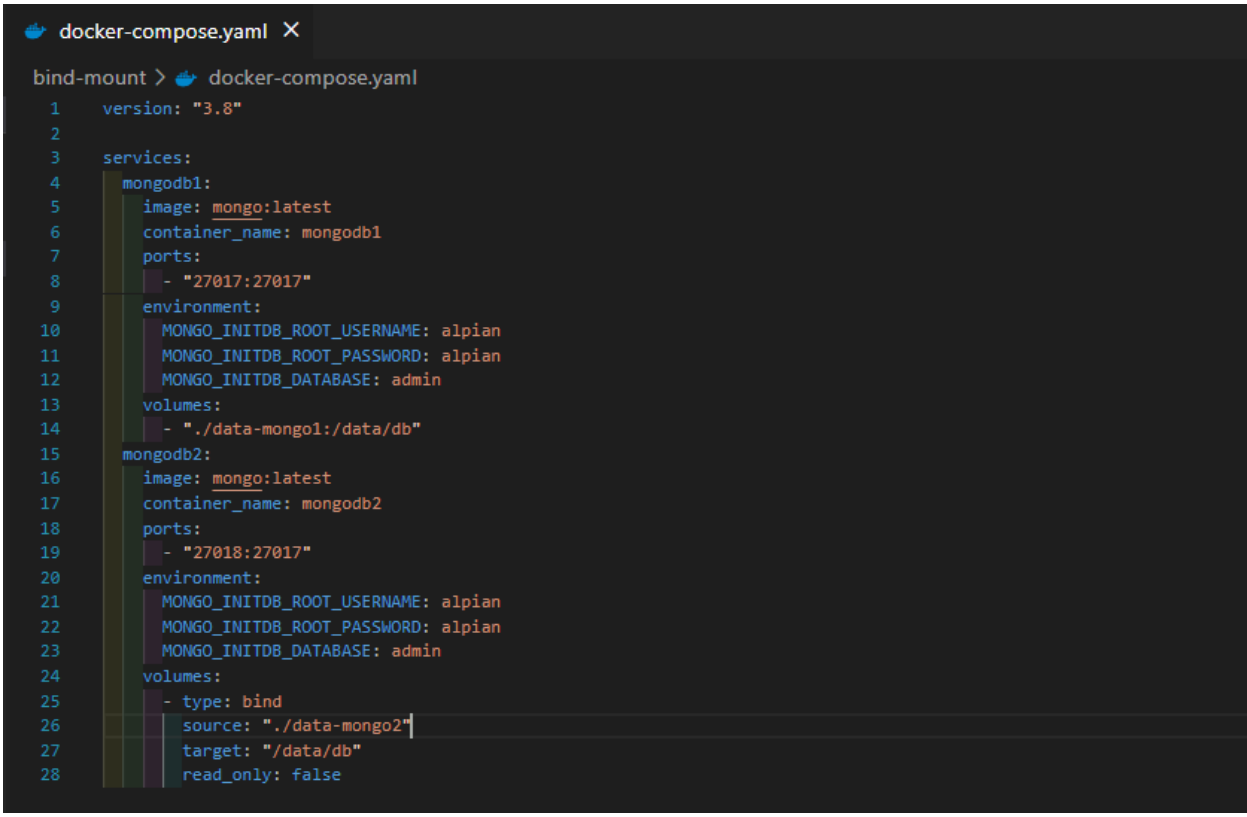
MODE adalah mode bind mount, ro untul readonly, rw untuk read write (default)

SOURCE bisa menggunakan relative path dengan diawali . (titik), atau absolute path.


PROBLEMS OUTPUT PORTS SQL CONSOLE GITLENS POSTMAN CONSOLE COMMENTS TERMINAL DEBUG CONSOLE
powershell
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/2
✓ Network bind-mount_default Created 0.1s
✓ Container mongodb1 Created 0.2s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\bind-mount> docker compose start
time="2024-11-09T23:04:26+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\bind-mount\\dock
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
✓ Container mongodb1 Started 1.0s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\bind-mount>
/wpu-resolusi#2218 needs reviewers
Connect
Live Share
Ln 14, Col 33
Spaces: 2
UTF-8
CRLF
Compose
Go Live


DOCKER COMPOSE
bind-mount
data-mongo1
.mongodb
diagnostic.data
journal
_mdb_catalog.wt
collection-0-11021954968786...
collection-2-11021954968786...
collection-4-11021954968786...
collection-7-11021954968786...
index-1-11021954968786799...
index-3-11021954968786799...
index-5-11021954968786799...
index-6-11021954968786799...
index-8-11021954968786799...
index-9-11021954968786799...
mongod.lock
sizeStorer.wt
storage.bson
WiredTiger
WiredTiger.lock
WiredTiger.turtle
WiredTiger.wt
WiredTigerHS.wt
docker-compose.yaml
services:
mongodb1:
image: mongo:latest
container_name: mongodb1
ports:
- "27017:27017"
environment:
MONGO_INITDB_ROOT_USERNAME: alpian
MONGO_INITDB_ROOT_PASSWORD: alpian
MONGO_INITDB_DATABASE: admin
volumes:
- "./data-mongo1:/data/db"
PROBLEMS OUTPUT PORTS SQL CONSOLE GITLENS
er-compose.yaml: the attribute `version` is ob
[+] Creating 2/2
✓ Network bind-mount_default Created
✓ Container mongodb1 Created
PS C:\Users\alpia\Documents\Materi dan Tugas K
time="2024-11-09T23:04:26+08:00" level=warning

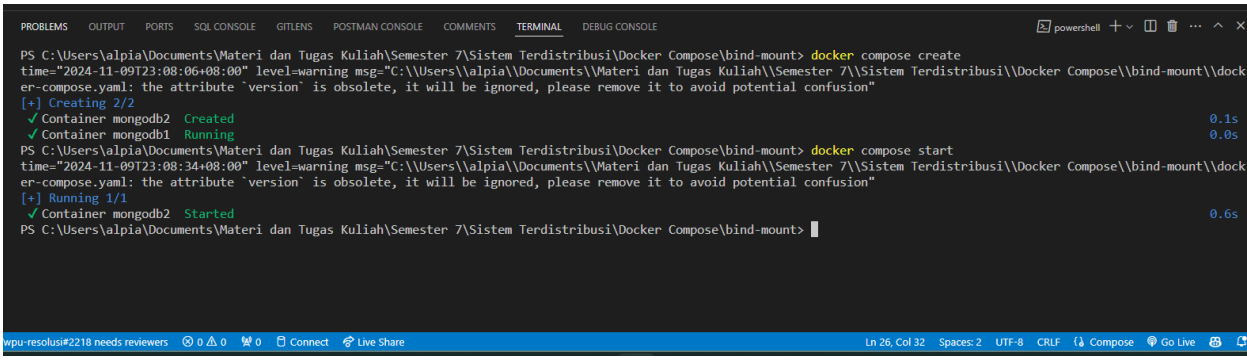
a. Bind Mount Long Syntax

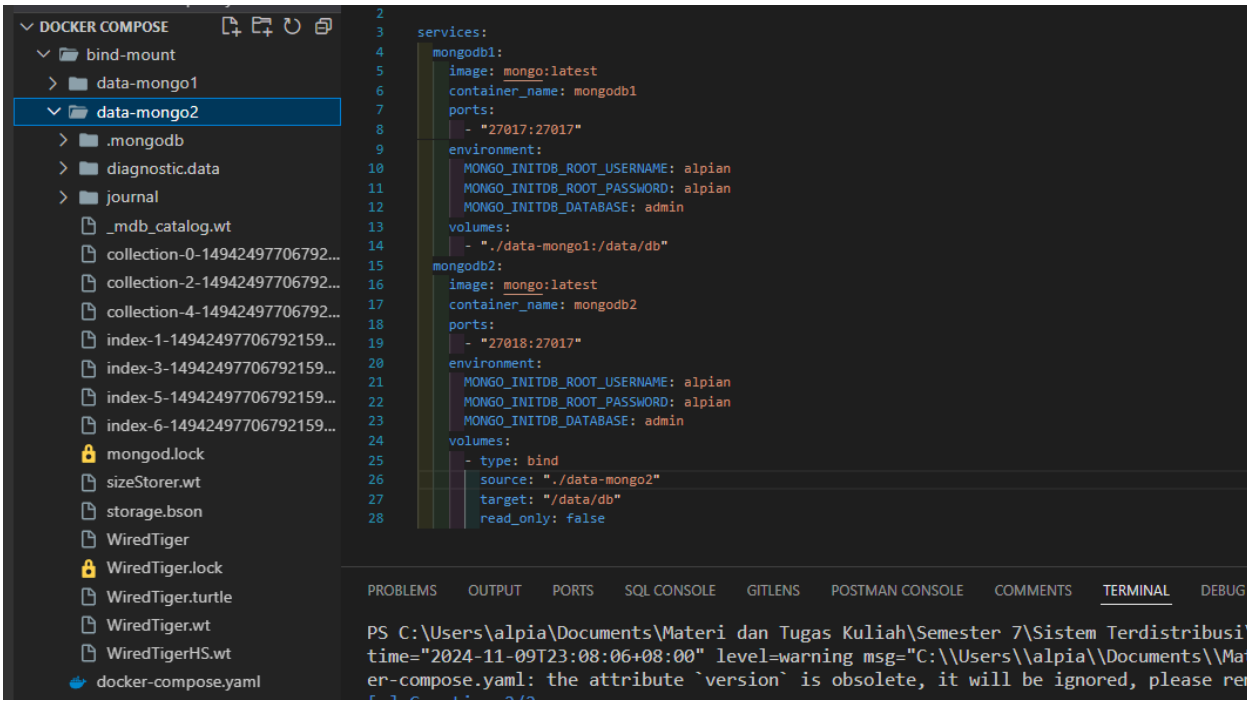


docker-compose.yaml

bind-mount > docker-compose.yaml

```yaml
version: "3.8"

services:
  mongodb1:
    image: mongo:latest
    container_name: mongodb1
    ports:
      - "27017:27017"
    environment:
      MONGO_INITDB_ROOT_USERNAME: alpian
      MONGO_INITDB_ROOT_PASSWORD: alpian
      MONGO_INITDB_DATABASE: admin
    volumes:
      - "./data-mongo1:/data/db"
  mongodb2:
    image: mongo:latest
    container_name: mongodb2
    ports:
      - "27018:27017"
    environment:
      MONGO_INITDB_ROOT_USERNAME: alpian
      MONGO_INITDB_ROOT_PASSWORD: alpian
      MONGO_INITDB_DATABASE: admin
    volumes:
      - type: bind
        source: "./data-mongo2"
        target: "/data/db"
        read_only: false
```



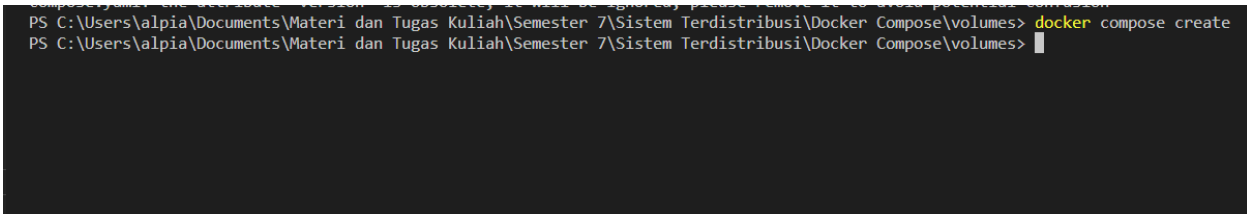


15. Volume

Docker Compose juga tidak hanya bisa digunakan untuk membuat container, tapi bisa juga digunakan untuk membuat volume

Kita bisa menggunakan attribute volumes pada konfigurasi file.

a. Volume



Pada bagian ini tidak membuat apapun, karena untuk docker compos mninmal memerlukan container terlebih dahulu, yang mana sebelumnya kita hanya ada

volumes, tetapi tidak ada apa-apa. Jadi, kita perlu menambahkan container yang menggunakan volumes.

b. Menggunakan volume short syntax

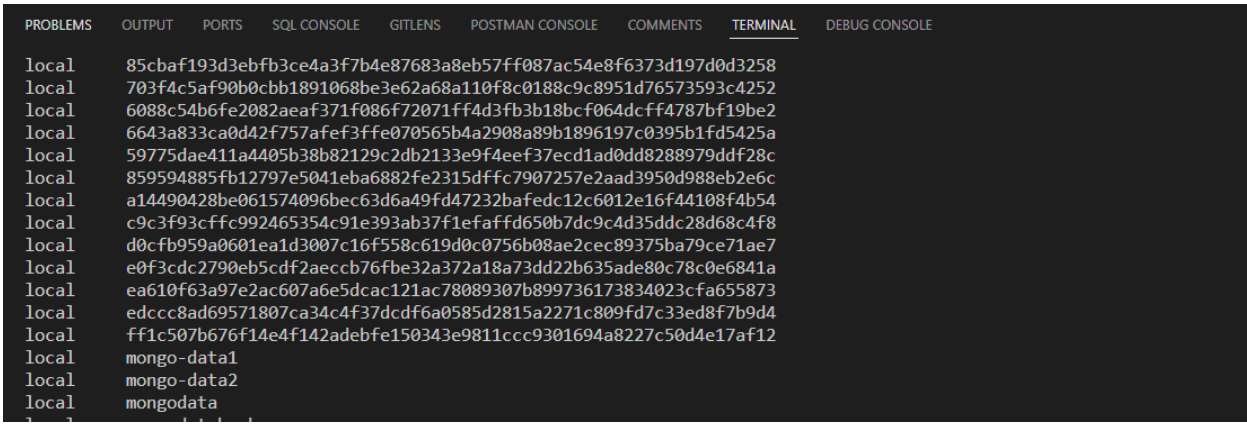
volumes > docker-compose.yaml

```yaml
version: '3.8'

services:
  mongodb1:
    image: mongo:latest
    container_name: mongodb1
    ports:
      - "27017:27017"
    environment:
      MONGO_INITDB_ROOT_USERNAME: alpian
      MONGO_INITDB_ROOT_PASSWORD: alpian
      MONGO_INITDB_DATABASE: admin
    volumes:
      - "mongo-data1:/data/db"
```

c. Menggunakan Volume Long Syntax

```yaml
  mongodb2:
    image: mongo:latest
    container_name: mongodb2
    ports:
      - "27018:27017"
    environment:
      MONGO_INITDB_ROOT_USERNAME: alpian
      MONGO_INITDB_ROOT_PASSWORD: alpian
      MONGO_INITDB_DATABASE: admin
    volumes:
      - type: volume
        source: mongo-data2
        target: "/data/db"
        read_only: false


volumes:
  mongo-data1:
    name: mongo-data1
  mongo-data2:
    name: mongo-data2
```

Cek Volume menggunakan “docker volume ls”

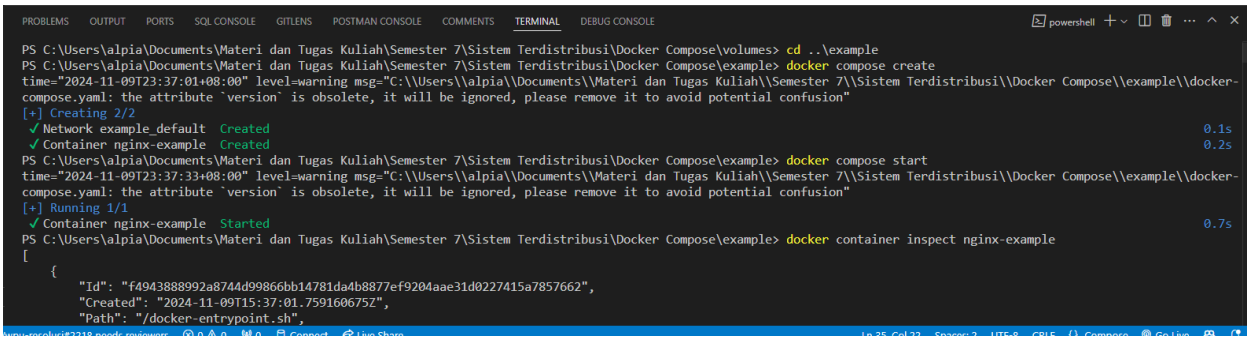


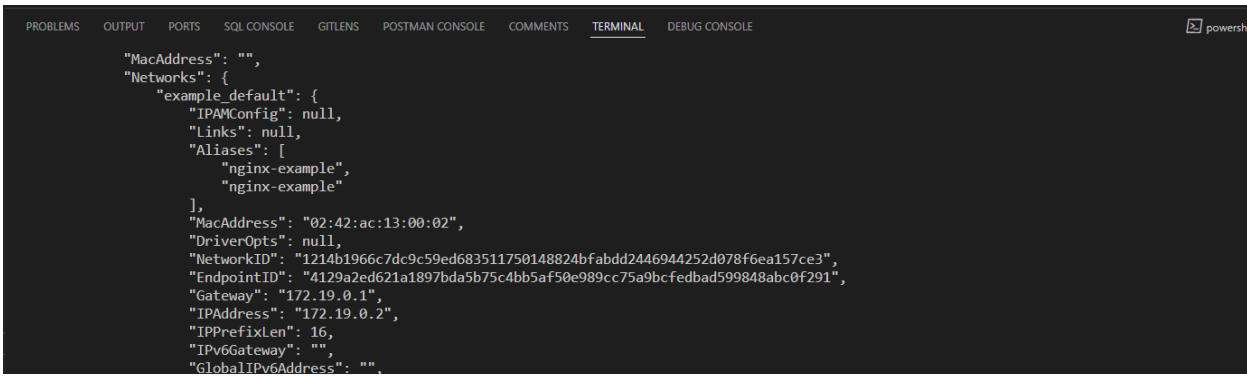
Terlihat bahwa sudah ada volume untuk mongo-data1 dan mongodata-2.

16. Network

Selain membuat Container dan Volume, kita juga bisa menggunakan Docker Compose untuk membuat Network secara otomatis.

a. Default Network





Pada Bagian Network, otomatis akan terkoneksi secara otomatis dengan network example_default.

b. Membuat Network

```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE

PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks> docker compose create
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks>
```

Sama dengan volume, bagian ini tidak membuat apapun, karena untuk docker compos mninmal memerlukan container terlebih dahulu.

c. Menggunakan Network

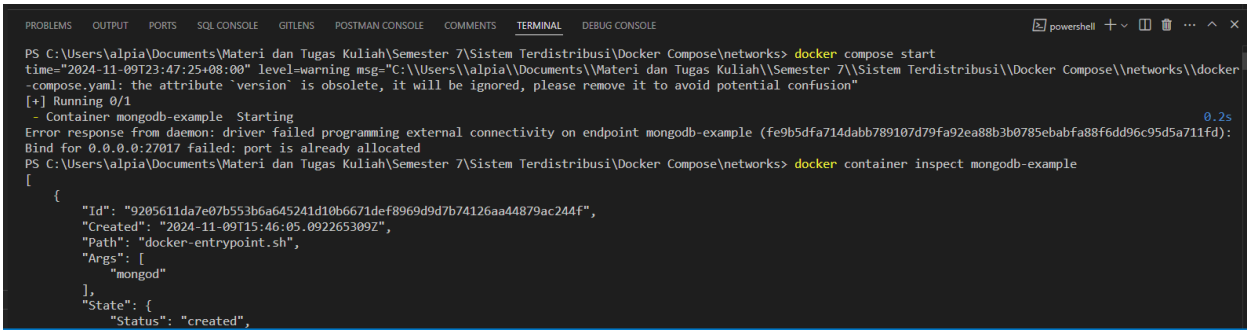
docker-compose.yaml

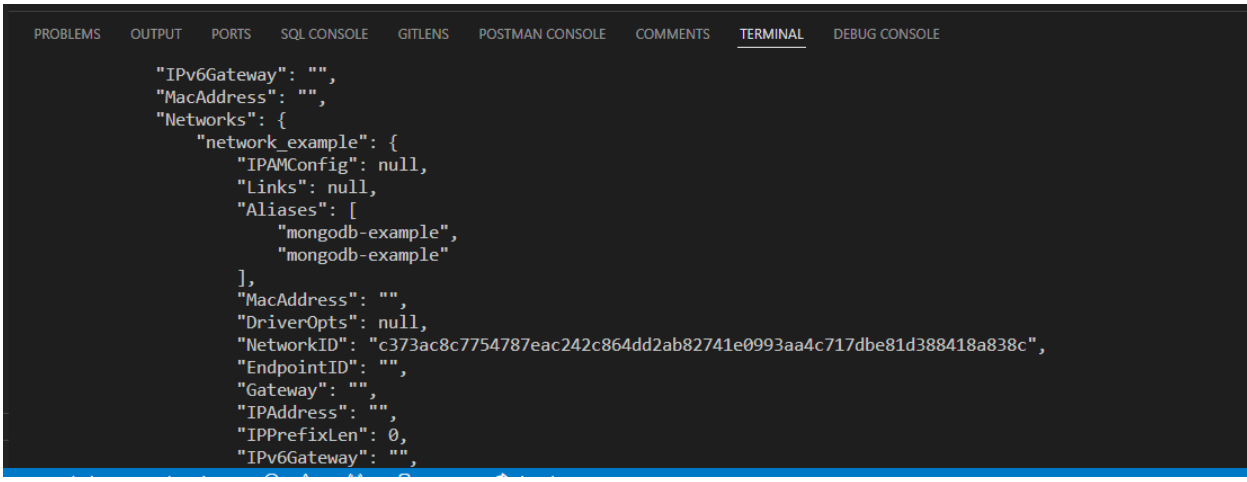
networks > docker-compose.yaml

```yaml
version: "3.8"

services:
  mongodb-example:
    image: mongo:latest
    container_name: mongodb-example
    ports:
      - "27017:27017"
    environment:
      MONGO_INITDB_ROOT_USERNAME: alpian
      MONGO_INITDB_ROOT_PASSWORD: alpian
      MONGO_INITDB_DATABASE: admin
    networks:
      - network_example

networks:
  network_example:
    name: network_example
    driver: bridge
```

```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE  powershell

PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks> docker compose create
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks> docker compose create
time="2024-11-09T23:46:04+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\networks\\docker
-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/2
 ✓ Network network_example    Created                                                                    0.1s
 ✓ Container mongodb-example  Created                                                                    0.1s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks>
```

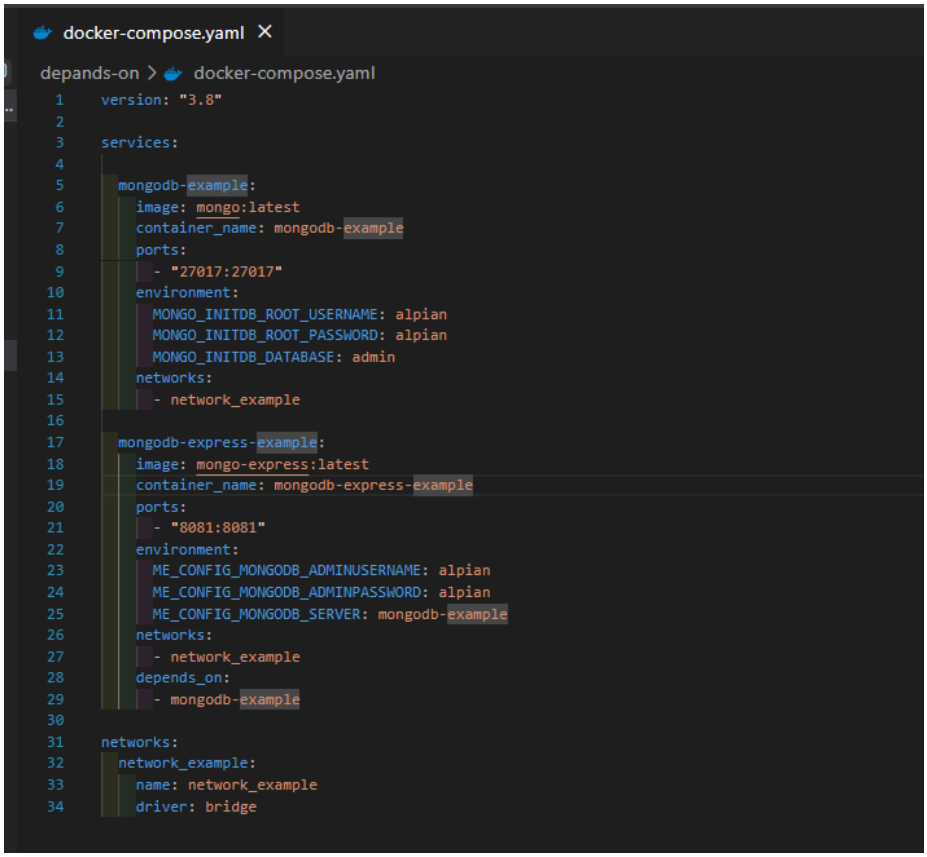
Akan membuat Networknya yaitu network_example (sesuai dengan nama yang dibuat), bukan lagi network_default.

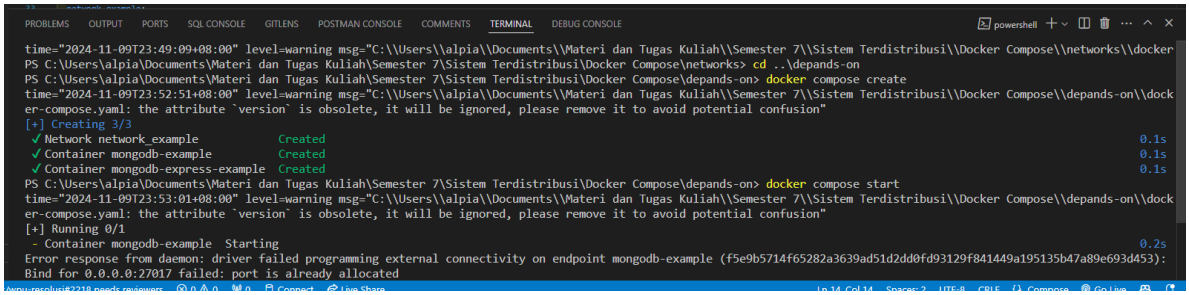

PROBLEMS OUTPUT PORTS SQL CONSOLE GITLENS POSTMAN CONSOLE COMMENTS TERMINAL DEBUG CONSOLE
powershell
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks> docker compose start
time="2024-11-09T23:47:25+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\networks\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 0/1
- Container mongodb-example Starting
0.2s
Error response from daemon: driver failed programming external connectivity on endpoint mongodb-example (fe9b5dfa714dabb789107d79fa92ea88b3b0785ebabfa88f6dd96c95d5a711fd): Bind for 0.0.0.0:27017 failed: port is already allocated
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\networks> docker container inspect mongodb-example
[
{
"Id": "9205611da7e07b553b6a645241d10b6671def8969d9d7b74126aa44879ac244f",
"Created": "2024-11-09T15:46:05.0922653092",
"Path": "docker-entrypoint.sh",
"Args": [
"mongod"
],
"State": {
"Status": "created",


PROBLEMS OUTPUT PORTS SQL CONSOLE GITLENS POSTMAN CONSOLE COMMENTS TERMINAL DEBUG CONSOLE
"IPv6Gateway": "",
"MacAddress": "",
"Networks": {
"network_example": {
"IPAMConfig": null,
"Links": null,
"Aliases": [
"mongodb-example",
"mongodb-example"
],
"MacAddress": "",
"DriverOpts": null,
"NetworkID": "c373ac8c7754787eac242c864dd2ab82741e0993aa4c717dbe81d388418a838c",
"EndpointID": "",
"Gateway": "",
"IPAddress": "",
"IPPrefixLen": 0,
"IPv6Gateway": "",

17. Depends On

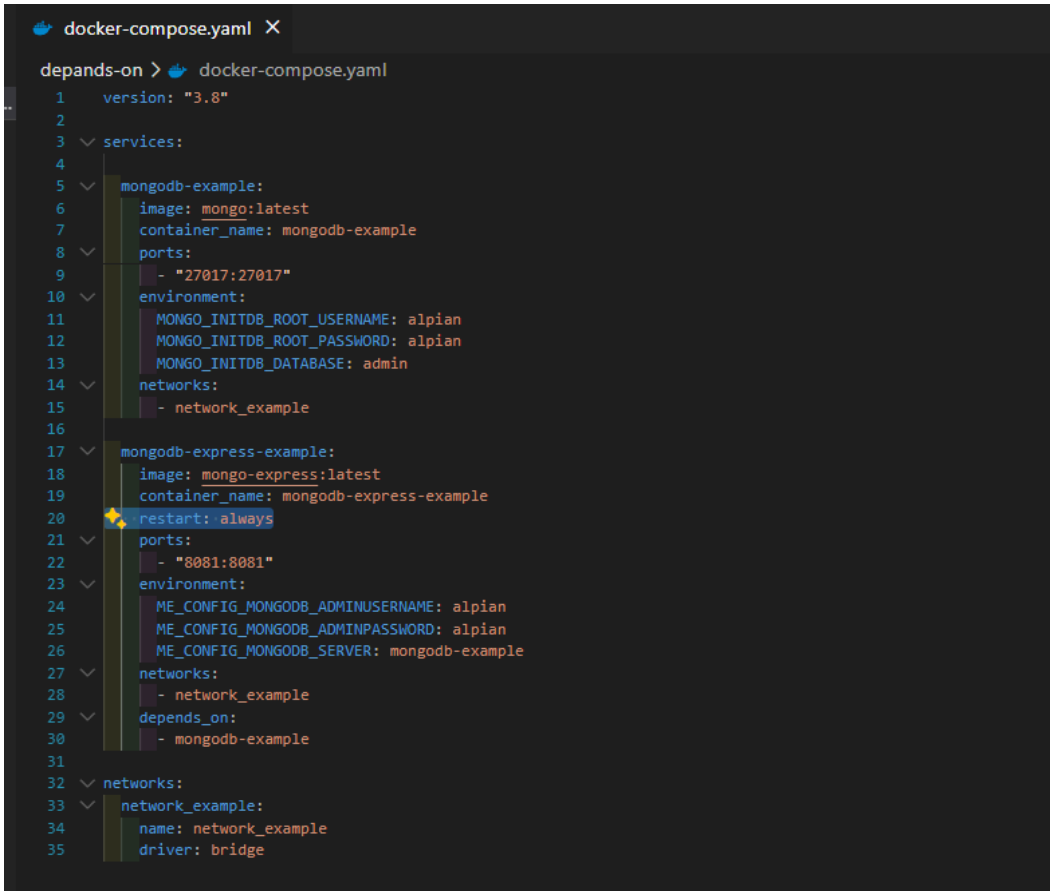
Saat membuat file Docker Compose dengan banyak container, sering kali ada container yang membutuhkan container lain untuk berjalan terlebih dahulu, atau dengan kata lain, memerlukan urutan tertentu dalam proses startup. Secara default, Docker Compose menjalankan semua container secara bersamaan tanpa urutan yang pasti. Untuk mengatur urutan ini, kita dapat menggunakan atribut `depends_on`. Dengan `depends_on`, kita dapat menentukan bahwa sebuah container hanya akan berjalan jika container lain telah berjalan terlebih dahulu. Atribut ini memungkinkan kita mencantumkan satu atau lebih container lain yang diperlukan sebagai dependensi, memastikan container berjalan dalam urutan yang dibutuhkan.

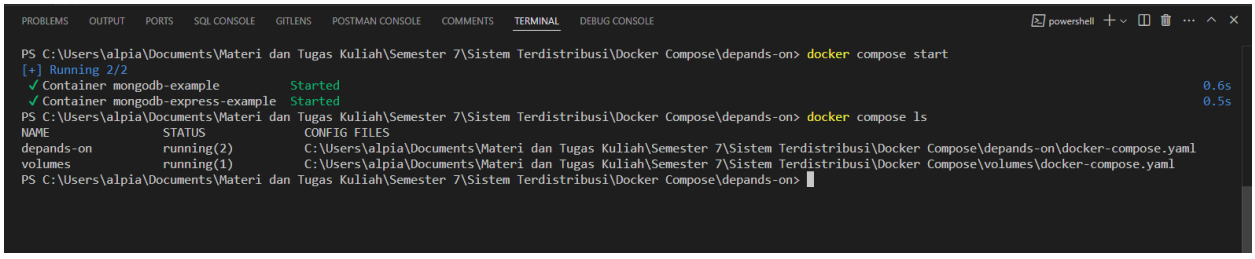




18. Restart

Secara default, saat sebuah container mati, Docker tidak akan menjalankannya lagi, sehingga kita harus memulainya secara manual. Untuk memastikan container selalu berjalan, kita dapat menambahkan atribut `restart` di konfigurasi Docker Compose, dengan beberapa opsi: `no` (tidak pernah restart, nilai default), `always` (selalu restart jika container berhenti, kecuali dihentikan secara manual; container akan restart saat Docker di-restart), `on-failure` (restart hanya jika terjadi error yang menyebabkan container exit dengan kode non-zero), dan `unless-stopped` (selalu restart container, kecuali dihentikan secara manual). Pengaturan ini membantu menjaga container tetap aktif atau me-restart otomatis saat terjadi masalah.

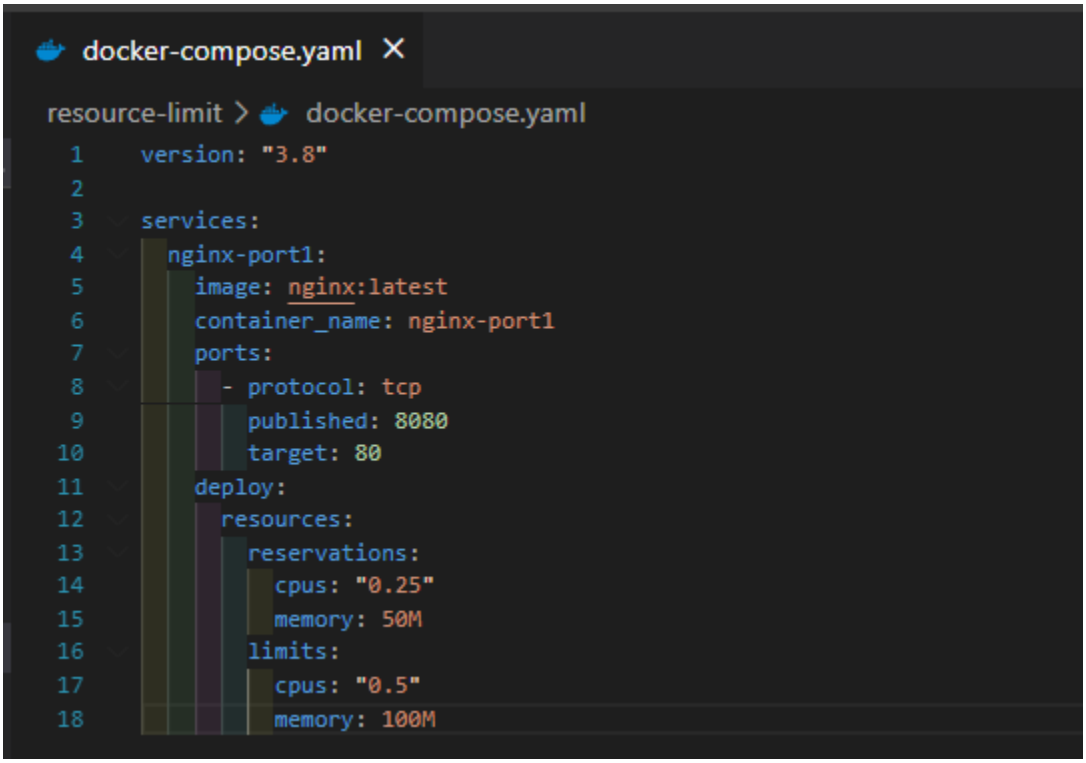




19. Resource Limit

Kita dapat menggunakan file konfigurasi Docker Compose untuk mengatur batasan sumber daya (resource limit) untuk CPU dan memori dari setiap container yang dibuat. Hal ini dapat dilakukan dengan menambahkan atribut `deploy` dan di dalamnya menggunakan atribut `resources`. Di dalam `resources`, kita dapat menentukan dua hal penting: `reservations` dan `limits`. Atribut `reservations` digunakan untuk memastikan bahwa sejumlah sumber daya tertentu dijamin dapat digunakan oleh container, sementara `limits` menetapkan batas maksimal sumber daya yang dapat digunakan oleh container. Namun, perlu diingat bahwa batas ini tidak menjamin eksklusivitas, karena sumber daya ini bisa saja diperebutkan dengan container lain jika ada persaingan dalam penggunaan sumber daya.

a. Resource Limit



b. Docker container stats

Jalankan terlebih dahulu.

```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE

PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\resource-limit> docker compose create
time="2024-11-10T07:51:46+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\resource-limit\\
docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/2
 ✓ Network resource-limit_default  Created                                                                    0.2s
 ✓ Container nginx-port1           Created                                                                    0.1s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\resource-limit> docker compose start
time="2024-11-10T07:51:55+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\resource-limit\\
docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
 ✓ Container nginx-port1  Started                                                                             0.8s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\resource-limit>
```

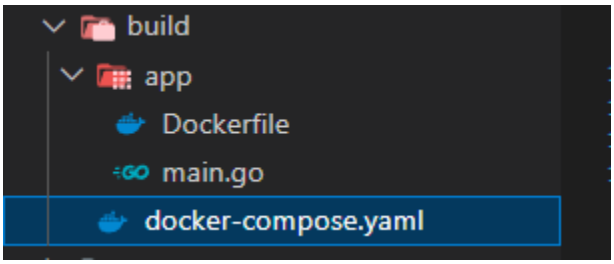
Hasil cek menggunakan “docker container stats”

```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE

CONTAINER ID   NAME          CPU %     MEM USAGE / LIMIT     MEM %     NET I/O         BLOCK I/O   PIDS
d4bdeab6e0a5   nginx-port1   0.00%     13.02MiB / 100MiB     13.02%    1.57kB / 0B     0B / 0B     13
9ff83ba6550f   mongodb2      0.74%     131.9MiB / 7.439GiB   1.73%     2.16kB / 291B   0B / 0B     43
```

20. Dockerfile

a. Struktur Folder

Membuat folder dengan struktur seperti ini.



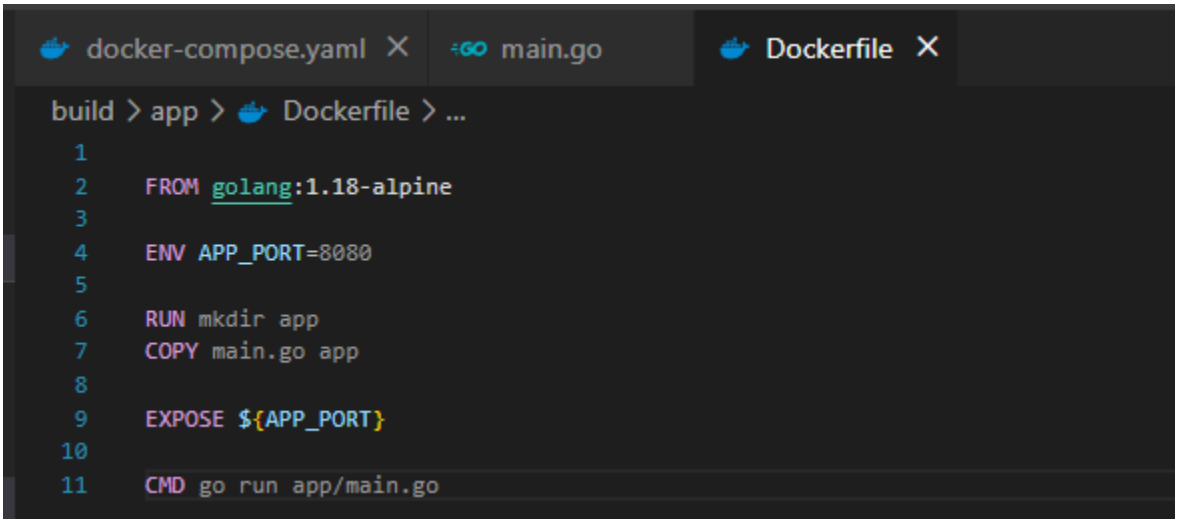
Isi dari docker-compose.yaml

docker-compose.yaml | main.go | Dockerfile

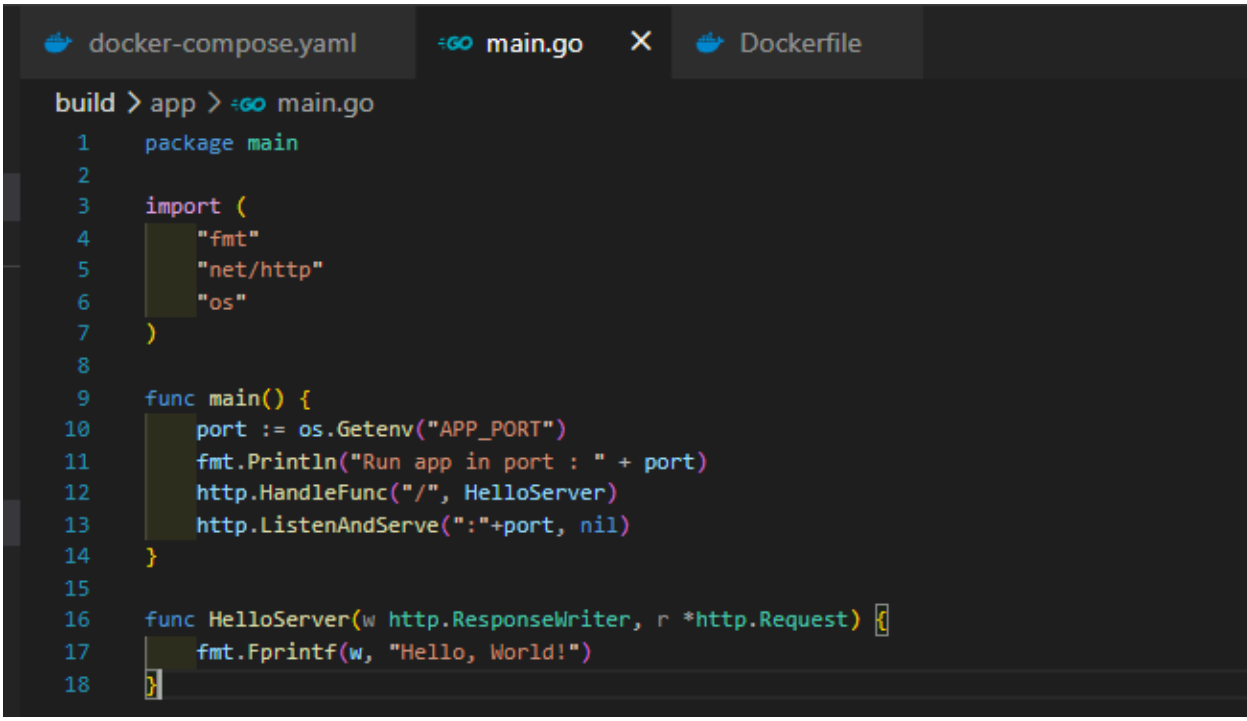
build > docker-compose.yaml

```yaml
version: "3.8"

services:
  app:
    container_name: app
    build:
      context: "./app"
      dockerfile: Dockerfile
    image: "app-golang:1.0.0"
    environment:
      - "APP_PORT=8080"
    ports:
      - "8080:8080"
```

Isi dari app/Dockerfile

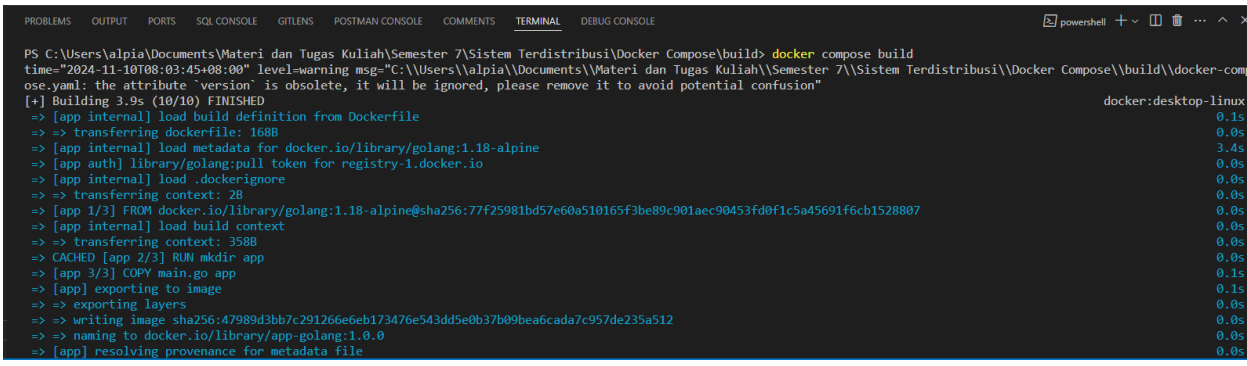


Isi dari app/main.go



b. Build

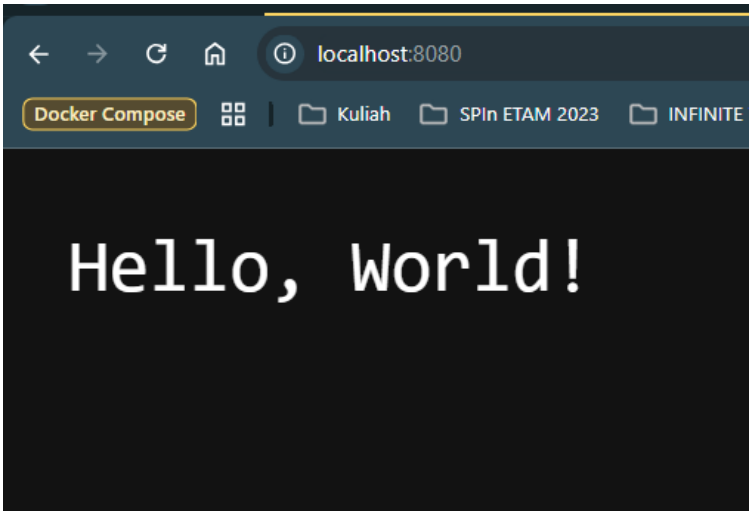
Ketika kita menggunakan perintah `docker compose start`, Docker Compose akan secara otomatis melakukan build terlebih dahulu jika image yang diperlukan belum terbuat. Namun, jika kita hanya ingin melakukan build image saja tanpa membuat container, kita bisa menggunakan perintah `docker compose build`. Perintah ini hanya akan membangun image sesuai dengan konfigurasi di file `docker-compose.yaml` tanpa menjalankan container, yang berguna ketika kita ingin memperbarui atau membangun image terlebih dahulu sebelum memulai container.



Hasil cek menggunakan “docker image ls”

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker image ls
REPOSITORY                TAG       IMAGE ID       CREATED          SIZE
app-golang                1.0.0     47989d3bb7c2   19 seconds ago   330MB
secondary-db              latest    fbda7dea498c   7 days ago       419MB
primary-db                latest    873b127ed483   7 days ago       419MB
roymunduss/multi          latest    980d47849356   3 weeks ago      14.1MB
roymunduss/entrypoint     latest    eb7a0da61e66   3 weeks ago      330MB
roymunduss/health         latest    3bc4d36018bf   3 weeks ago      332MB
roymunduss/arg            latest    743b3d023dcb   3 weeks ago      330MB
roymunduss/user           latest    d4ab9cac7b44   3 weeks ago      330MB
roymunduss/workdir        latest    4a0b1609940c   3 weeks ago      330MB
roymunduss/volume         latest    003bc7a30005   3 weeks ago      330MB
roymunduss/env            latest    5d46f4b0ed82   3 weeks ago      330MB
roymunduss/expose         latest    899acffcb42e   3 weeks ago      330MB
roymunduss/ignore         latest    9c3a40eccb35   3 weeks ago      7.8MB
roymunduss/copy           latest    89fe45c3d723   3 weeks ago      7.8MB
roymunduss/add            latest    b9587e02ca46   3 weeks ago      7.8MB
roymundus/run             latest    c009f86465e5   3 weeks ago      7.8MB
roymunduss/command        latest    6e2f1bf742bc   3 weeks ago      7.8MB
```

```
PROBLEMS  OUTPUT  PORTS  SQL CONSOLE  GITLENS  POSTMAN CONSOLE  COMMENTS  TERMINAL  DEBUG CONSOLE                powershell

PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker compose create
[+] Creating 2/2
 ✓ Network build_default  Created
 ✓ Container app          Created
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker compose start
[+] Running 0/1
 - Container app  Starting
Error response from daemon: driver failed programming external connectivity on endpoint app (d6cca91083fa7045830d5131c4f4d5ce3626f28b16c824f3dae4ca634cb599d3): Bind for 0.0
.0:8080 failed: port is already allocated
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker container ls
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS                     NAMES
d4bdeab6e0a5   nginx:latest   "/docker-entrypoint.…"   16 minutes ago   Up 16 minutes   0.0.0.0:8080->80/tcp      nginx-port1
9ff83ba6550f   mongo:latest   "docker-entrypoint.s…"   9 hours ago      Up 9 hours      0.0.0.0:27018->27017/tcp  mongodb2
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker stop d4bdeab6e0a5
d4bdeab6e0a5
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker compose start
[+] Running 1/1
 ✓ Container app  Started
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker compose ps
NAME      IMAGE                COMMAND                  SERVICE   CREATED              STATUS              PORTS
app       app-golang:1.0.0     "/bin/sh -c 'go run …"   app       About a minute ago   Up 12 seconds       0.0.0.0:8080->8080/tcp
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build>
```



c. Menghapus Image

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker compose down
[+] Running 2/2
 ✓ Container app          Removed
 ✓ Network build_default  Removed
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker image ls
REPOSITORY               TAG       IMAGE ID       CREATED          SIZE
app-golang               1.0.0     47989d3bb7c2   20 minutes ago   330MB
secondary-db             latest    fbda7dea498c   7 days ago       419MB
primary-db               latest    873b127ed483   7 days ago       419MB
roymunduss/multi         latest    980d47849356   3 weeks ago      14.1MB
roymunduss/entrypoint    latest    eb7a0da61e66   3 weeks ago      330MB
```

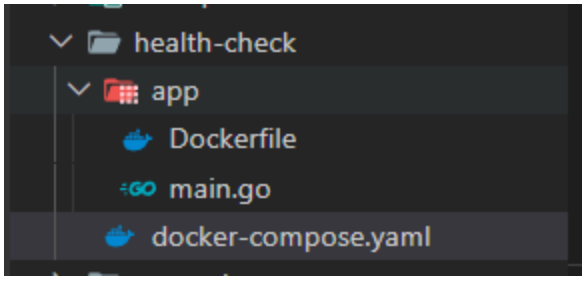
Untuk menghapus image, kita harus menghapus manual dengan menuliskan nama imagenya, seperti ini:

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker image rm app-golang:1.0.0
Untagged: app-golang:1.0.0
Deleted: sha256:47989d3bb7c291266e6eb173476e543dd5e0b37b09bea6cada7c957de235a512
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\build> docker image ls
REPOSITORY               TAG       IMAGE ID       CREATED          SIZE
secondary-db             latest    fbda7dea498c   7 days ago       419MB
primary-db               latest    873b127ed483   7 days ago       419MB
roymunduss/multi         latest    980d47849356   3 weeks ago      14.1MB
roymunduss/entrypoint    latest    eb7a0da61e66   3 weeks ago      330MB
roymunduss/health        latest    3bc4d36018bf   3 weeks ago      332MB
roymunduss/arg           latest    743b3d023dcb   3 weeks ago      330MB
roymunduss/user          latest    d4ab9cac7b44   3 weeks ago      330MB
roymunduss/workdir       latest    4a0b1609940c   3 weeks ago      330MB
roymunduss/volume        latest    003bc7a30005   3 weeks ago      330MB
roymunduss/env           latest    5d46f4b0ed82   3 weeks ago      330MB
roymunduss/expose        latest    899acffcb42e   3 weeks ago      330MB
roymunduss/ignore        latest    9c3a40eccb35   3 weeks ago      7.8MB
roymunduss/copy          latest    89fe45c3d723   3 weeks ago      7.8MB
roymunduss/add           latest    b9587e02ca46   3 weeks ago      7.8MB
roymunduss/label         latest    a391ac87bce5   3 weeks ago      7.8MB
```

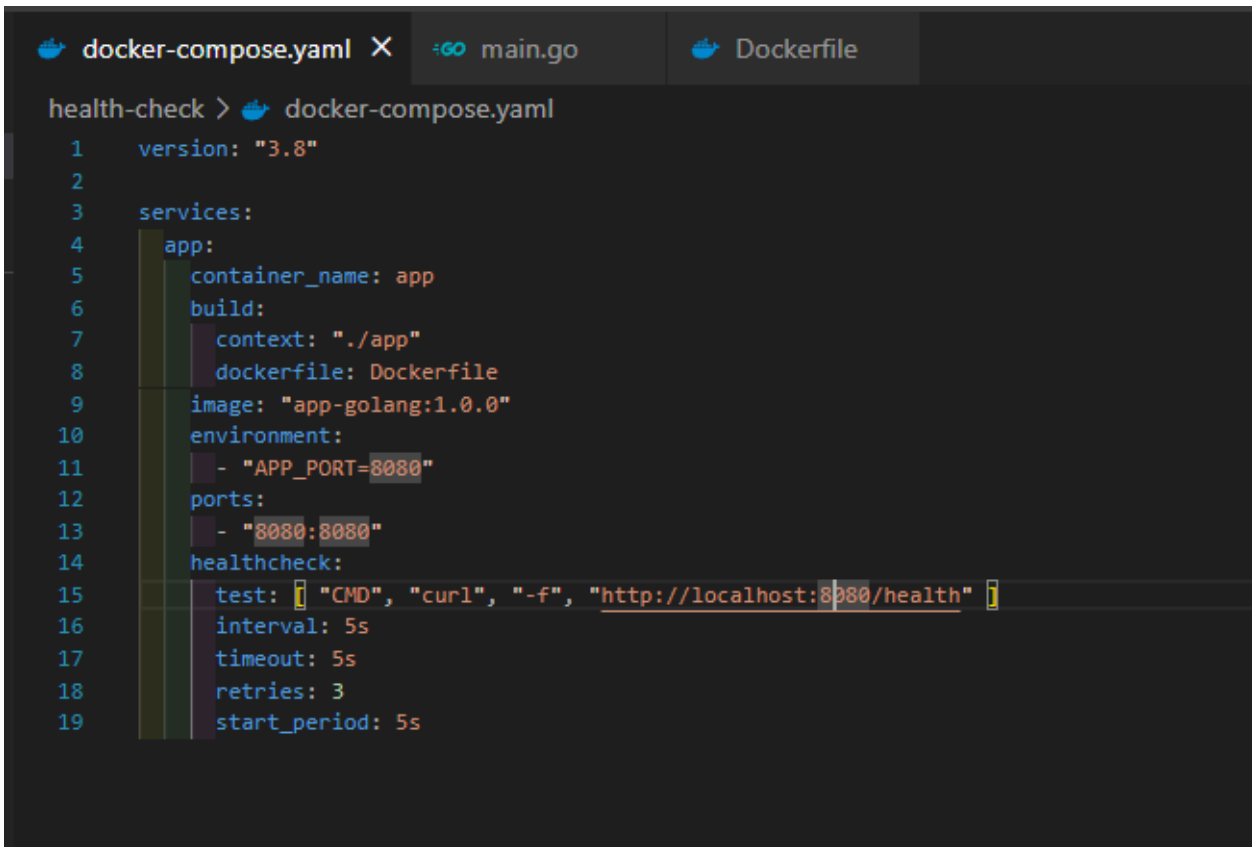
21. Health Check

Kita pernah membahas tentang **Container Health Check** pada materi Dockerfile. Secara default, container yang dibuat—baik secara manual maupun menggunakan Docker Compose—akan selalu menggunakan health check yang didefinisikan di Dockerfile. Namun, jika kita ingin mengubah health check tersebut, hal itu bisa dilakukan. Kita dapat mengonfigurasi health check baru di file `docker-compose.yaml` pada atribut `healthcheck` di bagian `services`. Dengan menambahkan pengaturan health check di Docker Compose, kita dapat menyesuaikan bagaimana Docker memeriksa kesehatan container, seperti menentukan interval, timeout, dan perintah yang digunakan untuk memeriksa status container.

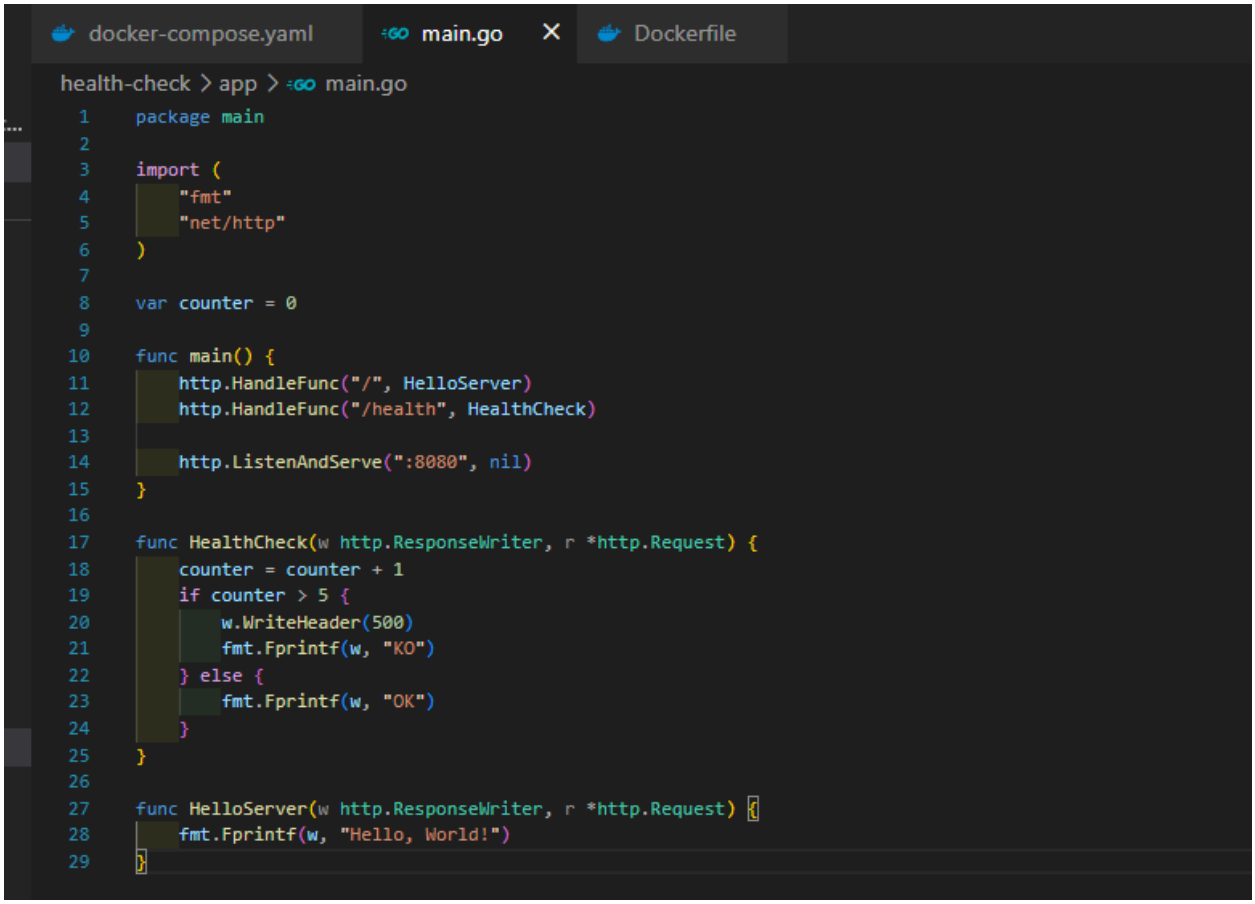
Buat struktur file seperti ini:



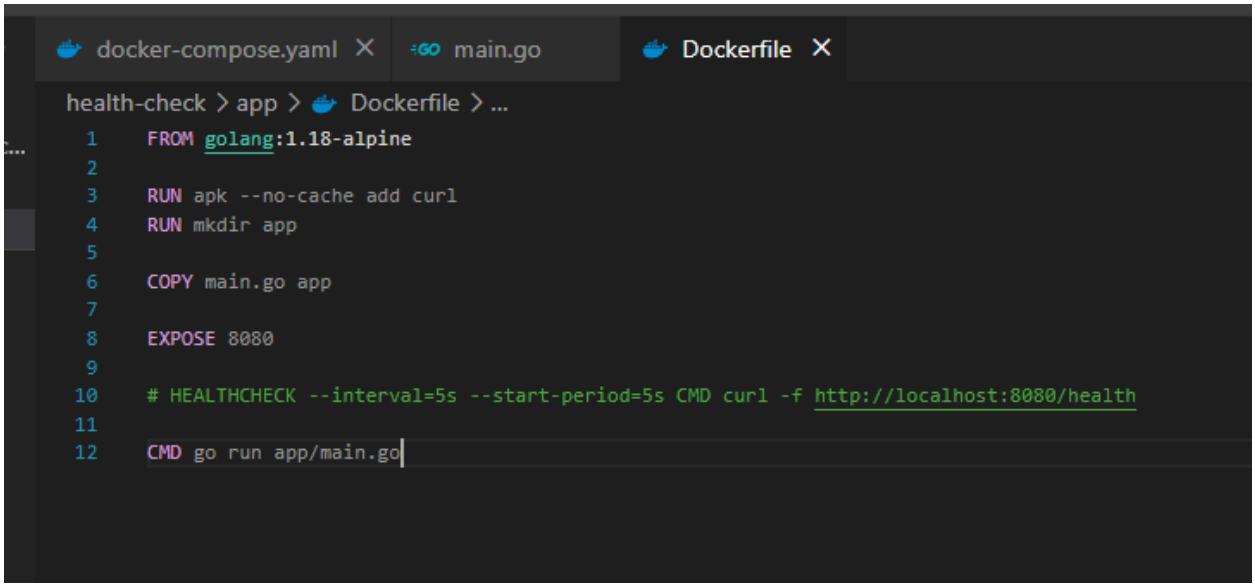
Isi dari docker-compose.yaml



Isi dari app/main.go



Isi dari app/Dockerfile



Proses Build, Create dan Start.

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker compose build
time="2024-11-10T08:36:43+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\health-check\\dock
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Building 3.3s (11/11) FINISHED                                                                                                                  docker:desktop-linux
 => [app internal] load build definition from Dockerfile                                                                                                          0.0s
 => => transferring dockerfile: 259B                                                                                                                              0.0s
 => [app internal] load metadata for docker.io/library/golang:1.18-alpine                                                                                         3.0s
 => [app auth] library/golang:pull token for registry-1.docker.io                                                                                                 0.0s
 => [app internal] load .dockerignore                                                                                                                             0.0s
 => => transferring context: 2B                                                                                                                                   0.0s
 => [app 1/4] FROM docker.io/library/golang:1.18-alpine@sha256:77f25981bd57e60a510165f3be89c901aec90453fd0f1c5a45691f6cb1528807                                   0.0s
 => [app internal] load build context                                                                                                                             0.0s
 => => transferring context: 531B                                                                                                                                 0.0s
 => CACHED [app 2/4] RUN apk --no-cache add curl                                                                                                                  0.0s
 => CACHED [app 3/4] RUN mkdir app                                                                                                                                0.0s
 => CACHED [app 4/4] COPY main.go app                                                                                                                             0.0s
 => [app] exporting to image                                                                                                                                      0.0s
 => => exporting layers                                                                                                                                           0.0s
 => => writing image sha256:6d0168c443a963bf59c536a0e130c42a4c8a7a87396a389a02898c74f3a3aed4                                                                      0.0s
 => => naming to docker.io/library/app-golang:1.0.0                                                                                                               0.0s
 => [app] resolving provenance for metadata file                                                                                                                  0.0s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker compose create
time="2024-11-10T08:36:58+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\health-check\\dock
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/1
 ✓ Network health-check_default  Created                                                                                                                          0.1s
 ✓ Container app                 Created                                                                                                                          0.1s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker compose start
time="2024-11-10T08:37:03+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\health-check\\dock
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
 ✓ Container app  Started                                                                                                                                         0.5s
```

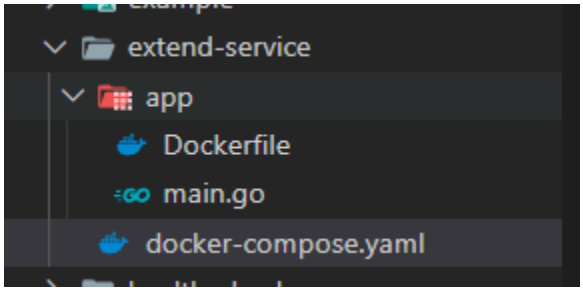
Health-check.

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker compose ps
time="2024-11-10T08:37:15+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\health-check\\dock
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME      IMAGE              COMMAND                  SERVICE   CREATED          STATUS                    PORTS                     NAMES
app       app-golang:1.0.0   "/bin/sh -c 'go run …"   app       16 seconds ago   Up 12 seconds (healthy)   0.0.0.0:8080->8080/tcp
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker container ls
CONTAINER ID   IMAGE              COMMAND                  CREATED          STATUS                    PORTS                     NAMES
d2fd745b0a18   app-golang:1.0.0   "/bin/sh -c 'go run …"   34 seconds ago   Up 30 seconds (healthy)   0.0.0.0:8080->8080/tcp    app
9ff83ba6550f   mongo:latest       "docker-entrypoint.s…"   9 hours ago      Up 9 hours                0.0.0.0:27018->27017/tcp  mongodb2
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker container ls
CONTAINER ID   IMAGE              COMMAND                  CREATED          STATUS                      PORTS                     NAMES
d2fd745b0a18   app-golang:1.0.0   "/bin/sh -c 'go run …"   54 seconds ago   Up 50 seconds (unhealthy)   0.0.0.0:8080->8080/tcp    app
9ff83ba6550f   mongo:latest       "docker-entrypoint.s…"   9 hours ago      Up 9 hours                  0.0.0.0:27018->27017/tcp  mongodb2
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker compose down
time="2024-11-10T08:38:14+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\health-check\\dock
er-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
 ✓ Container app               Removed                                                                                                                            0.7s
 ✓ Network health-check_default  Removed                                                                                                                          0.3s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check> docker image rm app-golang:1.0.0
Untagged: app-golang:1.0.0
Deleted: sha256:6d0168c443a963bf59c536a0e130c42a4c8a7a87396a389a02898c74f3a3aed4
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\health-check>
```

22. Extend Service

Saat membuat aplikasi menggunakan Docker, kadang kita ingin menjalankan aplikasi tersebut ke beberapa server baik itu di local laptop, di server development, atau server production. Kadang ada kalanya beberapa hal berbeda, misal konfigurasi misalnya pada kasus ini, mau tidak mau kita harus membuat banyak file konfigurasi Docker Compose, misal untuk di local, di development dan di production.

Buat Struktur file seperti ini:



a. Konfigurasi Utama

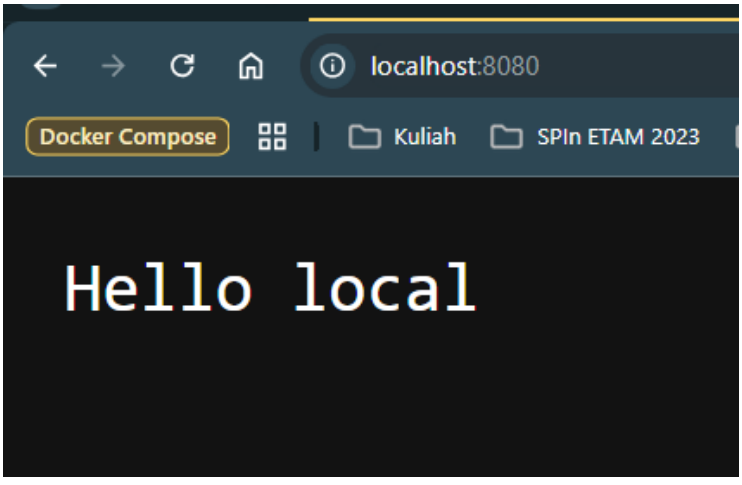
docker-compose.yaml | main.go | Dockerfile

extend-service > docker-compose.yaml

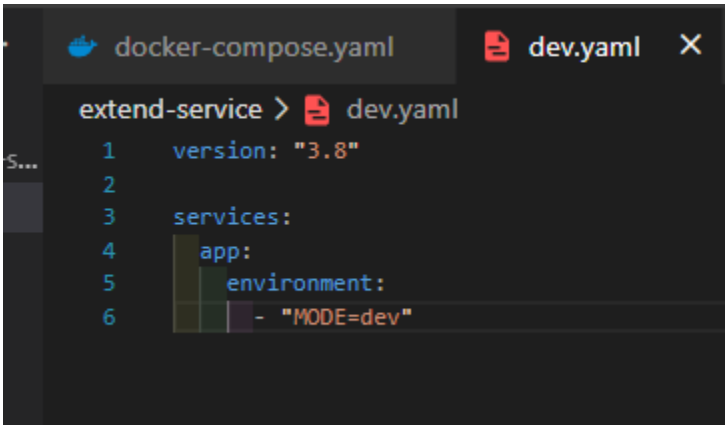
```yaml
version: "3.8"

services:
  app:
    container_name: app
    build:
      context: "./app"
      dockerfile: Dockerfile
    image: "app-golang:1.0.0"
    environment:
      - "APP_PORT=8080"
      - "MODE=local"
    ports:
      - "8080:8080"
```

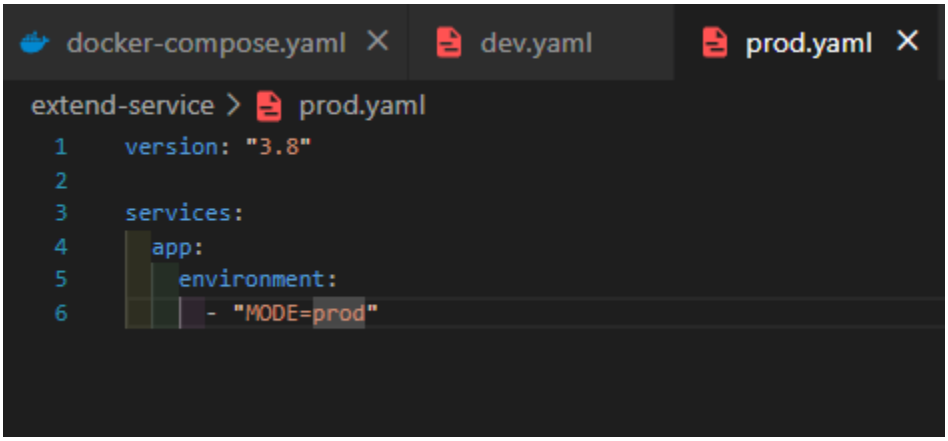
Build dan start.

```
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service> docker compose create
time="2024-11-10T08:46:46+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\do
cker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
 ! app Warning pull access denied for app-golang, repository does not exist or may require 'docker login': denied: requested access to the resource is d...  4.3s
[+] Building 2.9s (10/10) FINISHED                                                                                                    docker:desktop-linux
 => [app internal] load build definition from Dockerfile                                                                                               0.0s
 => => transferring dockerfile: 182B                                                                                                                   0.0s
 => [app internal] load metadata for docker.io/library/golang:1.18-alpine                                                                              2.2s
 => [app auth] library/golang:pull token for registry-1.docker.io                                                                                      0.0s
 => [app internal] load .dockerignore                                                                                                                  0.0s
 => => transferring context: 2B                                                                                                                        0.0s
 => CACHED [app 1/3] FROM docker.io/library/golang:1.18-alpine@sha256:77f25981bd57e60a510165f3be89c901aec90453fd0f1c5a45691f6cb1528807                 0.0s
 => [app internal] load build context                                                                                                                  0.0s
 => => transferring context: 415B                                                                                                                      0.0s
 => [app 2/3] RUN mkdir app                                                                                                                            0.4s
 => [app 3/3] COPY main.go app                                                                                                                         0.1s
 => [app] exporting to image                                                                                                                           0.1s
 => => exporting layers                                                                                                                                0.1s
 => => writing image sha256:40ba61b06835084e85f10d09126bb459e98793e84ec25938e2147aa053e9dd64                                                           0.0s
 => => naming to docker.io/library/app-golang:1.0.0                                                                                                    0.0s
 => [app] resolving provenance for metadata file                                                                                                       0.0s
[+] Creating 2/2
 ✓ Network extend-service_default  Created                                                                                                             0.1s
 ✓ Container app                   Created                                                                                                             0.1s
```



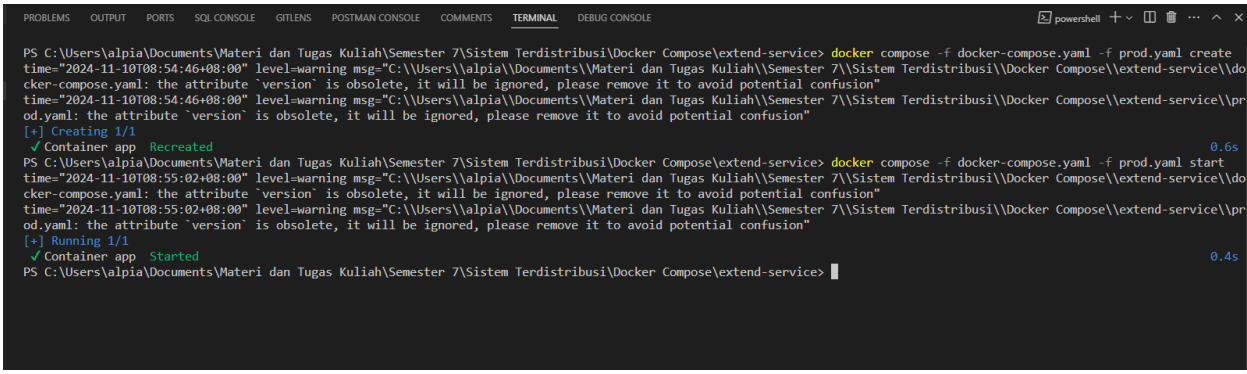
b. Konfigarasi Dev dan Prod

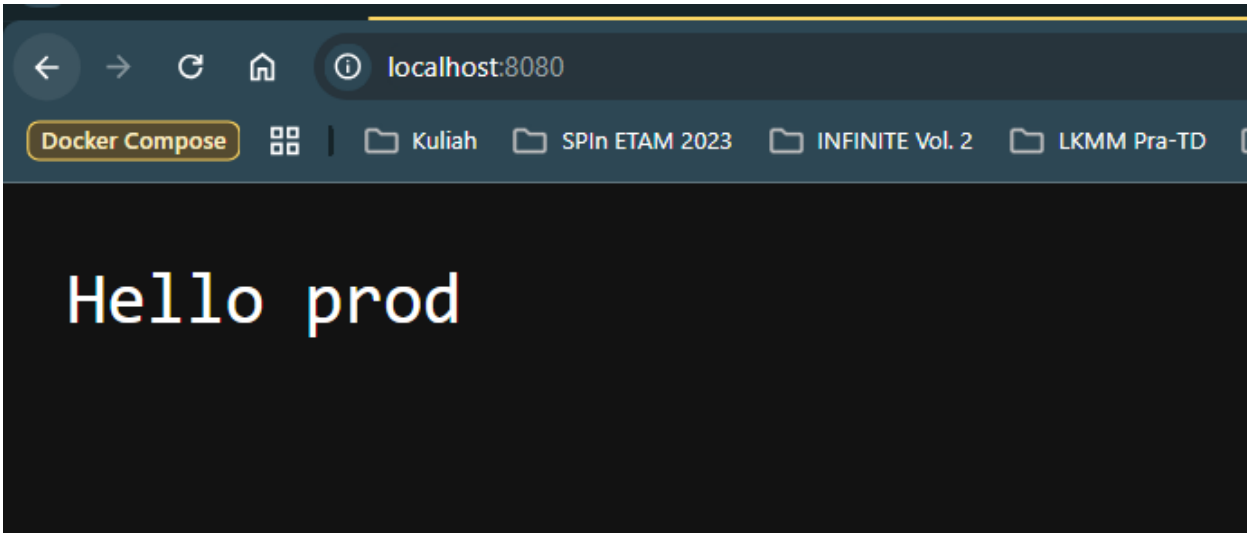


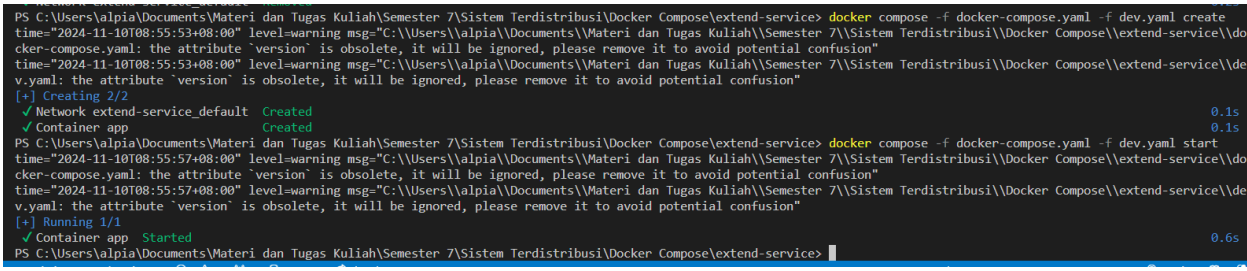


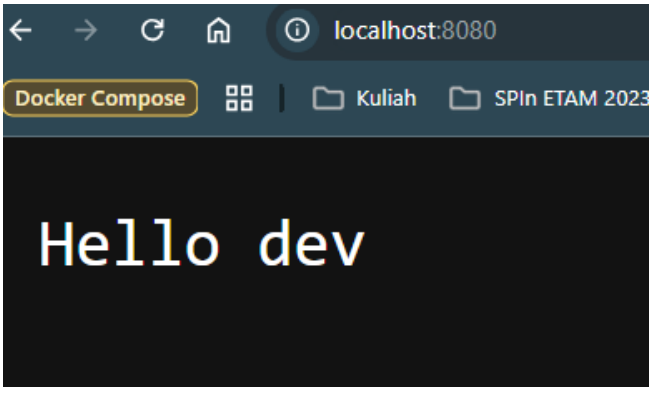
c. Extend Service

Docker Compose memiliki fitur bernama **extend service**, yang memungkinkan kita untuk menggabungkan beberapa file konfigurasi sekaligus. Fitur ini berguna untuk membuat konfigurasi umum yang dapat digunakan di berbagai environment, serta file konfigurasi khusus untuk setiap environment, misalnya untuk development, staging, atau production. Dengan cara ini, kita dapat memisahkan konfigurasi umum dan konfigurasi spesifik environment, membuatnya lebih mudah untuk dikelola. Saat menjalankan Docker Compose, kita bisa menggunakan perintah `-f namafile.yaml` untuk menentukan nama file konfigurasi yang ingin digunakan, jika file tersebut bukan `docker-compose.yaml` yang default.


PROBLEMS OUTPUT PORTS SQL CONSOLE GITLENS POSTMAN CONSOLE COMMENTS TERMINAL DEBUG CONSOLE
powershell
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service> docker compose -f docker-compose.yaml -f prod.yaml create
time="2024-11-10T08:54:46+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
time="2024-11-10T08:54:46+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\prod.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 1/1
✓ Container app Recreated 0.6s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service> docker compose -f docker-compose.yaml -f prod.yaml start
time="2024-11-10T08:55:02+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
time="2024-11-10T08:55:02+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\prod.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
✓ Container app Started 0.4s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service>


localhost:8080
Docker Compose
Kuliah
SPIn ETAM 2023
INFINITE Vol. 2
LKMM Pra-TD
Hello prod


PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service> docker compose -f docker-compose.yaml -f dev.yaml create
time="2024-11-10T08:55:53+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
time="2024-11-10T08:55:53+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\dev.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 2/2
✓ Network extend-service_default Created 0.1s
✓ Container app Created 0.1s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service> docker compose -f docker-compose.yaml -f dev.yaml start
time="2024-11-10T08:55:57+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
time="2024-11-10T08:55:57+08:00" level=warning msg="C:\\Users\\alpia\\Documents\\Materi dan Tugas Kuliah\\Semester 7\\Sistem Terdistribusi\\Docker Compose\\extend-service\\dev.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/1
✓ Container app Started 0.6s
PS C:\Users\alpia\Documents\Materi dan Tugas Kuliah\Semester 7\Sistem Terdistribusi\Docker Compose\extend-service>


localhost:8080
Docker Compose
Kuliah
SPIn ETAM 2023
Hello dev