

Mysql container és docker webszerver létrehozása

Létre kell hozni egy mappát, az én esetemben application lesz a neve. Ezen belül 2 új mappát készítek, egy site és egy sql nevűt. Az előbbiben lesznek a webszerver adatai az utóbbiban pedig a mysql-el kapcsolatos fájlok.

A site mappában lesz 2 file

index.php

```
<h1>Table:</h1>
<?php
$dbhost = $_ENV["MYSQL_HOST"];
$dbuser = $_ENV["MYSQL_USER"];
$dbpass = $_ENV["MYSQL_PASSWORD"];
$dbdatabase = $_ENV["MYSQL_DATABASE"];
$conn = mysqli_connect($dbhost, $dbuser, $dbpass, $dbdatabase);

if(! $conn){
    die(mysqli_error($conn));
}

mysqli_select_db( $conn, 'db');
$sql = "SELECT * FROM student";
$ret = mysqli_query($conn, $sql);
echo "<table class='table table-striped'> <tr><th>id</th><th>name</th><th>neptuncode</th><th>age</th></tr>";
while($row = mysqli_fetch_array($ret)){
    echo "<tr>";
    echo "<td>".$row["id"]."</td>";
    echo "<td>".$row["name"]."</td>";
    echo "<td>".$row["neptuncode"]."</td>";
    echo "<td>".$row["age"]."</td>";
    echo "</tr>";
}
echo "</table>";
mysqli_close($conn);
?>
```

Docker file:

```
FROM php:7.2-apache

ENV MYSQL_HOST=
ENV MYSQL_USER=
ENV MYSQL_PASSWORD=
ENV MYSQL_DATABASE=

RUN docker-php-ext-install mysqli && docker-php-ext-enable mysqli

RUN apt-get update

EXPOSE 80

COPY . /var/www/html/
```

Miután ezekkel végeztem átléptem az sql mappába. Létrehoztam egy comm.sql file-t:

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

CREATE DATABASE IF NOT EXISTS `db` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
USE `db`;

CREATE TABLE `student` (
  `id` int NOT NULL,
  `name` varchar(255) DEFAULT NULL,
  `neptuncode` varchar(6) DEFAULT NULL,
  `age` int DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8_hungarian_ci;;

INSERT INTO student VALUES(2, 'Gipsz Jakab', 'ABC123', 55 );
INSERT INTO student VALUES(3, 'teszt', 'neptun', 99 );

COMMIT;
```

A fő mappán belül még két file-t kell létrehozni:

.env

```
MYSQL_USER=web_user
MYSQL_PASSWORD=passwd
MYSQL_ROOT_PASSWORD=passwd
MYSQL_DATABASE=db

MYSQL_HOST=database
```

compose.yaml:

```
version: "3.0"

services:
  webserver:
    build:
      context: ./site
      dockerfile: Dockerfile
    image: bead 1.0
    environment:
      - MYSQL_USER=${MYSQL_USER}
      - MYSQL_PASSWORD=${MYSQL_PASSWORD}
      - MYSQL_DATABASE=${MYSQL_DATABASE}
      - MYSQL_HOST=${MYSQL_HOST}
    ports:
      - 80:80
    networks:
      - web
    depends_on:
      database:
        condition: service_healthy
  database:
    image: mysql
    command: --default-authentication-plugin=mysql_native_password
    volumes:
      - ./sql:/docker-entrypoint-initdb.d
    environment:
      - MYSQL_USER=${MYSQL_USER}
      - MYSQL_PASSWORD=${MYSQL_PASSWORD}
      - MYSQL_DATABASE=${MYSQL_DATABASE}
      - MYSQL_HOST=${MYSQL_HOST}
    ports:
      - 3306:3306
    networks:
      - web
    healthcheck:
      test: ["CMD", "mysqladmin", "ping", "-h", "localhost"]
      timeout: 3s
      retries: 10
  adminer:
    image: adminer
    networks:
      - web
    ports:
      - 127.0.0.1: 8080:8080
    depends_on:
      database:
        condition: service_healthy
  networks:
    web:
      driver: bridge
```

Az elkészítések után parancssorban megnyitottam a mappát majd kiadtam a docker-compose up parancsot. Ez lefuttatja a compose.yaml fájlt ami létrehozza az összes szükséges konténert.