**MariaDB**

**valódi alkalmazáshoz; mindig explicit el kell indítani; Docker konténerben tegyük: imageből, környezeti változók megadásával(database, user, passworld), konténer portjának kivezetése localhost portjára; inicializálás: pom.xml JDBC driver felvétele és application propertiesben a kapcsolódási paraméterek megadása**

A **H2 demo alkalmazásokhoz jó vagy integrációs teszekhez,** de valódi alkalmazás esetén komolyabb adatbázis kell

**MariaDb indítása Docker-rel**

*Hogy kapcsolódni tudjunk hozzá, el kell indítani; tegyük ezt egy Docker konténerben:*

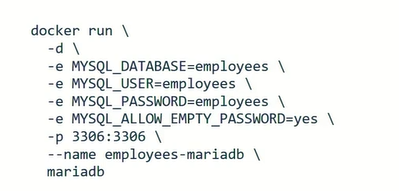
**már létező mariadb imageből** indulunk ki, ha nincs a gépünkön, leszedi a dockerhubról

**környezeti változóként meg lehet adni a paramétereket**, hogy hogyan induljon el:database, user, password

root felhasználóval üres jelszós belépés legyen engedélyezett

**-p: MariaDb konténeren belüli 3306os portját hozza ki a hostgép 3306os portjára, így sajátgépes heidisqllel hozzá tudunk kapcsolódni a konténerben futó adatbázishoz**

--name: konténer elnevezése

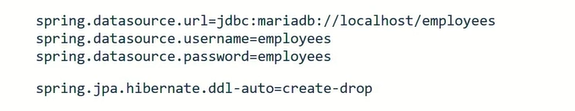


**Inicializálás**

*Alkalmazás módosítása, hogy ehhez a már konténerben futó adatbázishoz hozzácsatlakozzon*

**pom.xml: fel kell venni a JDBC drivert**

**applicaiton.properties:** itt adjuk meg, **hova kapcsolódjon** az alkalmazás:

****

**12Factor ajánlások Backing Service-el kapcsolatban**

**(bs def: minden az alkalmazásból elérhető külső erőforrás; adatbázis és egyéb példák; konténerben futó szolgáltatások; microservice rnyezetben az alkalmazások egymás backing service-ei; telepítés automatikusan, eszközzel, Infrastructure as Code; filerendszer az nem bs; embeddable háttérszolgáltatások és korlátaik; leállás esetén újracsatlakozás; egyéb komponensek pl Circuit bracker-> leállás esetén temp a hozzáférés megszűntetése-> túlterheltség megakadályozása)**

**backing service: összes külső erőforrás, amit az alkalmazásunkból el akarunk érni;** leggyakoribb az adatbázis; de ezen túl még üzenetküldő middlewarek, directory és email szerverek, elosztott cache, Big Data eszközök, stb.

**minden olyan külön konténerben futó szolgáltatás, amit az alkalmazás igénybe vesz;** **microservice környezetben,** ahol sok kis alkalmazás kapcsolódik össze ott **az alkalmazások egymás backing service-ei**

ezek **telepítése automatikusan történjen valamilyen eszköz segítségével,** amivel **meg lehet adni az infrastruktúrát kódosan(Infrastrucutre as Code**); ez alapján jöjjön létre és telepítődjön fel; ilyen eszköz pl. : Ansible, Chef, Puppet **környezeti változóként adjuk meg; így a legegyszerűbb paramétert bejuttatni a konténerbe**

az alkalmazás tudjon hozzáférni a backingekhez; az **elérést** általában meg kell adni; ezt lehetőleg ?

**filerendszer nem tekinthető backendservicenek; hiányzik belőle** a magas rendelkezésre állás, a tranzakcionalitás; fileok tárolására is valami erre alkalmas backing servicet kell alkalmaznunk

**Léteznek ún. embeddable háttérszolgáltatások; pl H2 adatbázis, ami az alkalmazás részét képezi; éles környezetben nem jó,** arra használjunk külön konténert; ekkor független az alkalmazástól, külön telepíthető, külön managelhető, újraindítható, deployolható

Amennyiben **leáll egy backing** és újraindítjuk**, az alkalmazás automatikusan tudjon újrakapcsolódni** hozzá

A microservice architektúráknál sok egyéb komponens létezik; pl **Circuit braker** eszköz: ha észleli, hogy egy **backing service leáll, akkor megszűnteti egy időre a hozzáférést;** e egyfajta biztosíték és azt a problémát küszöböli ki, hogyha nem vennénk le a serviceről a leállás után a felgyülemlő forgalmat, lehet nem tudna újra indulni a túlterheltség végett

**MariaDB – gyakorlat**

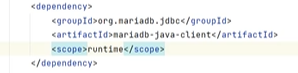
**docker run --name mariadbtest -e MYSQL\_DATABASE=movies -e MYSQL\_USER=movies -e MYSQL\_PASSWORD=movies -e MYSQL\_ALLOW\_EMPTY\_PASSWORD=yes -p 3305:3306 -d mariadb**

1. Módosítom az alkalmazást, hogy H2 helyett MariaDbn fusson

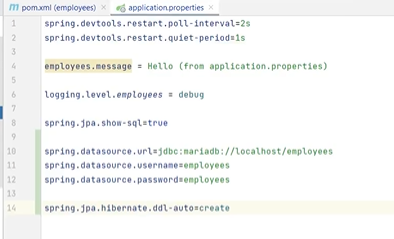
MDb elindítása Docker konténerben: terminal:

**docker run --name exam-mariadb -e MYSQL\_DATABASE= exam -e MYSQL\_USER= exam -e MYSQL\_PASSWORD= exam -e MYSQL\_ALLOW\_EMPTY\_PASSWORD=yes -p 3305:3306 -d mariadb**

1. pom.xml: h2 kivétele, és mariadb driver felvétele: runtime scoppeal(nem akarok az alkalmazásból hivatkozni a jdbc dirver osztályaira)



1. applicaiton properties:



1. springbootos alkalmazás elindítása; http requestek:

GET(még semmi), POST 2új alkalmazott, GET

1. **IDEA-ban új adatbázis kapcsolat létrehozása:IDEA jobb gördítő sáv: Database + jel -> Data Sorce ->MariaDb; user password és database megadás; Test Connection, ok; jobb oldalt megjelenik az adatbázis; középen megjelenik egy konzol SQL utik kiadására(A dokcer környezetben futó adatbázis user interfacet ideán keresztül kezelem)**

select \* from employees

**Kérdések**

*Mikor jó a H2, és mikor kell komolyabb adatbázis?*

*Hogy indítunk el mariadb adatbázis docker konténerben?*

*Hogy konfiguráljuk a springes alkalmazásunkat a mariadb használatához?*

*Mik a 12Factor ajánlásai Backing Service-el kapcsolatban?*