**JUNIT**

**Bevezetés a JUnit használatába**

Unit tesztelés

Mi nem Unit teszt

jUnit

Manuális ellenőrzés

Given, When, Then

Metódus tesztelés

Metódus tesztelés egybevonva

Maven integráció

POM.xml módosítás

IDE integráció

**Unit tesztelés ígéretei**

**Futtatás Mavennel**

**Tesztesetek életciklusa**

Test fixture

Test fixture külön interfészben

Izoláltság

Metódusok futtatásának sorrendje

Disabled

Conditional test execution

**Elnevezések**

**Assert**

Assert metódusok

AssertEquals lebegőpontos számok esetén

Assert metódusok több elem esetén

Egyszerre több Assert

Egymásba ágyazhatóság

Assert kollekciókon

Alternatív üzenet

Assume

**Kivételkezelés és timeout tesztelése**

Kivételkezelés

Timeout vizsgálata

Preemtively

**Egymásba ágyazás**

Belső osztályok

**Tagek és metaannotációk használata**

Tagek

Filterelés Mavennel

Metaannotációk

új Annotáció létrehozása

**Tesztesetek ismétlése**

Egyszerű ismétlés

Név módosítása

Repetitioninfo injektálás

**Paraméterezett tesztek**

Egyszerű értékekkel

Enum

Methodsource

CSV

Tempdir extension

**JUnit legjobb gyakorlatok**

F.I.R.S.T elvek

jUnit további tanácsok

Gyakori hibák jUnit használatakor

**AssertJ**

Egyszerű assert

Kollekciók

Tuple és szűrés

Üzenetek testreszabása

Soft Assertion

Assumption

pom.xml

**Mockito**

Külső függőségek

Unit tesztek csoportosítása

MOCKITO

Service Test mock objektummal

Dependency Injection

Visszatérési érték definiálása

Metódushívás ellenőrzése

Paraméter ellenőrzése

POM.XML

**Teszt lefedettség**

Jacoco

**JAVA SPRING BOOT MICROSERVICES**

**Bevezetés a Spring Framework és Spring Boot használatába**

Spring Framework céljai

Nagyvállalati alkalmazás -1.

Nagyvállalati alkalmazás -2. - további elvárások (6)

Spring Framework tulajdonságai -1.

Spring Framework tulajdonságai -2.

Applicaiton Context

Háromrétegű webes alkalmazás

**Bevezetés a Spring Boot használatába**

Spring Boot – 1.

Spring Boot – 2.

pom.xml

**Spring Beanek**

Controller komponensek

Controller annotációk

Controller

Service

Kapcsolatok

Függőség a Controllerben

**Konfiguráció Javaban**

Java konfigurációs példa

**Build és futtatás Mavennel**

**Unit és integrációs tesztek**

Unit tesztelés

Unit tesztelés függőséggel

Integrációs tesztek

Integrációs teszt példa

**Bevezetés a Docker használatába**

Docker

Archirtektúra

Docker felhasználási módja

További Docker komponensek

Docker forgalmak

Docker konténerek - parancsok

Műveletek image-ekkel

**Java alkalmazások Dockerrel**

Maven Plugin

Dockerfile – Maven plugin esetén

Parancsok

12Factor hivatkozás: Disposability

**Feltöltés Git repositoryba**

12Factor hivatkozás – one codebase, one application

**REST webszolgáltatások - GET művelet**

Restful webszolgáltatások tulajdonságai

Springbe hogyan kell rest webszogáltatásokat implementálni?

Controller

Architektúra

Lombok

Modelmapper

**GET műveletek paraméterezése**

1 - cél: Az Url-ben át lehessen adni egy prefix értéket

2 - cél: URL részletek kezelése

12FACTOR hivatkozás (=hogyan fejlesszünk alkalmazásokat) hogy érvényesül a RESTFUL API-knál?

Konkurencia

**REST webszolgáltatások POST és DELETE művelet**

Idempotencia

Controller POST művelettel

Controller Delete művelettel

**Státuszkódok és hibakezelés**

Kódok

Hogyan lehet Controllerből ilyen státusz kódokat visszaadni?

Hogyan lehet standard módon mindig ugyanazt a státuszkódot visszaadni?

Hogyan történik a hibakezelés?

Egy ExceptionHandler metódust hogyan kell megvalósítani?

Szabvány a hibakezelésre – RFC 7807

Problem

**Integrációs tesztelés**

Webes fajtái

WEB réteg tesztelése

Teljes alkalmazás tesztelése konténer nélkül

Teljes alkalmazás tesztelése konténerrel = RestTemplate

Mire való a RestTemplate és mit ellenőrzünk vele?

**Validáció**

**Intro**

*Mit jelent a validáció, és pl. mit ellenőrzünk vele?*

*SpringBoot milyen beépített támogatás tartalmaz erre? Mi a lényege?*

*Hogyan lehet validációt megadni? Milyen fajtái vannak? Példák!*

**Validáció Controlleren**

*Hogyan érjük el azt, hogy a Spring automatikusan vizsgálja meg a Dto paramétereket, amikor az adott metódus meghívásra kerül?*

*Hogyan küldünk vissza szabványos validáéciós hibaüzit?*

**Twelve-Factor App**

## ADATBÁZIS PROGRAMOZÁS JAVA NYELVEN - JPA

**Egyszerű mentés JPA-val**

pom.xml, persistence unit, persistence.xml

Entitások

Azonosítógenerálás

Mentés EntityManagerrel

**További egyszerű műveletek JPA-val**

Lekérdezés

Törlés azonosító alapján

**Architektúra és integrációs tesztelés**

Architektúra

DAO

Integrációs tesztelés

**Entitások konfigurálása**

Sémagenerálás

Generálás adatbázisba, Generálás szkriptbe

## Annotációk használata

Tábla konfigurálása

Oszlopok konfigurálása

Egyszerű típusú mezők, enumeration, Lob

Dátum és időkezelés

Tranziens mezők

**Elsődleges kulcs**

Elsődleges kulcs típusa

Azonosítógenerálás

Azonosítógenerálás táblával

Azonosítógenerálás szekvenciával

Azonosítógenerálás identity alapján

Összetett azonosítók

**Entitások életciklusa**

A JPA hogyan manageli az entitások állapotait

Persistence context fogalma

Lifecycle események

Flush metódus

**Többértékű attribútumok**

Egyszerű típus

Személyre szabás

Map perzisztálása

Lazy kapcsolat, lazy problémák és megoldás

**Kapcsolatok**

Milyen tulajdonság alapján lehet a kapcsolatokat kategorizálni?

Gyakrabban használt kombinációk

getReference

Cascade

Orphan removal

Sorrendezés

Szülő oldal

N+1 probléma

Lazy

Entity Graph

**Lekérdezések**

JPQL

A JPQL tulajdonságai

JPQL példa Path expression példán

Egyszerű lekérdezés

Lekérdezés streammel

Egy találat

Paraméterezett lekérdezés

Criteria API

MetaModel API

Lapozás

Definiálás named query-ként

Join

Objektum tömb

Projection query

Query timeout

**Haladó lekérdezések**

Milyen operátorok használhatóak a lekérdezésekben?

Aggregált lekérdezések

Allekérdezések

Beépített függvények

Beépített függvények

CASE

**Deklaratív tranzakciókezelés**

Hogyan lehet a tranzakciókezelést konfigurálni?

Propagáció, @Transactional

Izoláció

Visszagörgetési szabályok

Timeout

**Spring Data JPA**

Adatbáziskezelés evolúció Javaban

Spring data JPA előnyei

Spring Data JPA használatba vétele

JPAREPOSITORY

Entitás

Repository

H2, H2 elérés

**MariaDB**

vs. H2

MariaDb indítása Docker-rel

Inicializálás

12Factor ajánlások Backing Service-el kapcsolatban

**Integrációs tesztelés**

Mikor lesz már integrációs?

@DataJpaTest

@SpringBootTest

Tesztek H2 adatbázisban

Séma inicializáció

Adatfeltöltés

Tesztek egymásra hatása

**Alkalmazás futtatása Dockerben MariaDB-vel**

Architektúra

Link vs. network

Parancsok

**Séma inicializálás Flyway-jel**

Elvárások

Flyway- Spring Boottal integráció

**Swagger UI**