**Difference between JPA, Hibernate and Spring Data JPA**

**JPA:**

persistence.xml:

<persistence xmlns="http://xmlns.jcp.org/xml/ns/persistence"

version="2.2">

<persistence-unit name="JPAUnit">

<provider>org.hibernate.jpa.HibernatePersistenceProvider</provider>

<class>com.example.model.Country</class>

<properties>

<property name="javax.persistence.jdbc.driver" value="org.h2.Driver"/>

<property name="javax.persistence.jdbc.url" value="jdbc:h2:mem:testdb"/>

<property name="javax.persistence.jdbc.user" value="sa"/>

<property name="javax.persistence.jdbc.password" value=""/>

<property name="hibernate.dialect" value="org.hibernate.dialect.H2Dialect"/>

<property name="hibernate.hbm2ddl.auto" value="create-drop"/>

</properties>

</persistence-unit>

</persistence>

Country.java:

package com.example.model;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "country")

public class Country {

@Id

private String code;

private String name;

// getters & setters

}

JPAService.java:

package com.example.service;

import com.example.model.Country;

import javax.persistence.EntityManager;

import javax.persistence.EntityManagerFactory;

import javax.persistence.Persistence;

import java.util.List;

public class JPAService {

private EntityManagerFactory emf =

Persistence.createEntityManagerFactory("JPAUnit");

public List<Country> getAllCountries() {

EntityManager em = emf.createEntityManager();

List<Country> list = em

.createQuery("SELECT c FROM Country c", Country.class)

.getResultList();

em.close();

return list;

}

}

**Hibernate:**

hibernate.cfg.xml:

<hibernate-configuration>

<session-factory>

<property name="hibernate.dialect">org.hibernate.dialect.H2Dialect</property>

<property name="hibernate.connection.driver\_class">org.h2.Driver</property>

<property name="hibernate.connection.url">jdbc:h2:mem:testdb</property>

<property name="hibernate.connection.username">sa</property>

<property name="hibernate.connection.password"></property>

<property name="hibernate.hbm2ddl.auto">create-drop</property>

<mapping class="com.example.model.Country"/>

</session-factory>

</hibernate-configuration>

Country.java:

package com.example.model;

import javax.persistence.\*;

@Entity

@Table(name = "country")

public class Country {

@Id

@Column(name = "co\_code")

private String code;

@Column(name = "co\_name")

private String name;

// getters & setters

}

HibernateUtil.java:

package com.example.util;

import org.hibernate.SessionFactory;

import org.hibernate.cfg.Configuration;

public class HibernateUtil {

private static final SessionFactory sf =

new Configuration().configure().buildSessionFactory();

public static SessionFactory getSessionFactory() {

return sf;

}

}

HibernateService.java:

package com.example.service;

import com.example.model.Country;

import com.example.util.HibernateUtil;

import org.hibernate.Session;

import org.hibernate.Transaction;

import java.util.List;

public class HibernateService {

public List<Country> getAllCountries() {

Session session = HibernateUtil.getSessionFactory().openSession();

Transaction tx = session.beginTransaction();

List<Country> list = session

.createQuery("FROM Country", Country.class)

.list();

tx.commit();

session.close();

return list;

}

}

**Spring Data JPA:**

application.properties:

spring.datasource.driverClassName=org.h2.Driver

spring.datasource.url=jdbc:h2:mem:testdb;DB\_CLOSE\_DELAY=-1

spring.datasource.username=sa

spring.datasource.password=

spring.jpa.hibernate.ddl-auto=create-drop

Country.java:

package com.example.model;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "country")

public class Country {

@Id

private String code;

private String name;

// getters & setters

}

CountryRepository.java:

package com.example.repository;

import com.example.model.Country;

import org.springframework.data.jpa.repository.JpaRepository;

public interface CountryRepository

extends JpaRepository<Country, String> {

// derive queries: List<Country> findByName(String name);

}

CountryService.java:

package com.example.service;

import com.example.model.Country;

import com.example.repository.CountryRepository;

import org.springframework.stereotype.Service;

import javax.transaction.Transactional;

import java.util.List;

@Service

public class CountryService {

private final CountryRepository repo;

public CountryService(CountryRepository repo) {

this.repo = repo;

}

@Transactional

public List<Country> getAllCountries() {

return repo.findAll();

}

}