✅ Case Study Title: Citizen and Passport Management System

pom.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
        <modelVersion>4.0.0</modelVersion>
        <parent>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-starter-parent</artifactId>
                <version>3.5.4</version>
                <relativePath/> <!-- lookup parent from repository -->
        </parent>
        <groupId>com.example</groupId>
        <artifactId>citizen-passport</artifactId>
        <version>0.0.1-SNAPSHOT</version>
        <name>citizen-passport</name>
        <description>Demo project for Spring Boot Hibernate</description>
        <url/>
        <licenses>
                <license/>
        </licenses>
        <developers>
                <developer/>
        </developers>
        <scm>
                <connection/>
                <developerConnection/>
                <tag/>
                <url/>
        </scm>
        <properties>
                <java.version>21</java.version>
        </properties>
        <dependencies>
                <dependency>
        <groupId>org.hibernate.orm</groupId>
        <artifactId>hibernate-core</artifactId>
        <version>6.3.1.Final</version>
    </dependency>
    <dependency>
       <groupId>jakarta.persistence</groupId>
       <artifactId>jakarta.persistence-api</artifactId>
```

```xml
            <version>3.1.0</version>
        </dependency>
                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-data-jpa</artifactId>
                </dependency>
                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-web</artifactId>
                </dependency>
                <dependency>
                        <groupId>com.mysql</groupId>
                        <artifactId>mysql-connector-j</artifactId>
                        <scope>runtime</scope>
                </dependency>
                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-test</artifactId>
                        <scope>test</scope>
                </dependency>
                <dependency>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-api</artifactId>
        <version>2.0.9</version>
        </dependency>
        <dependency>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-simple</artifactId>
        <version>2.0.9</version>
        </dependency>
            </dependencies>
            <build>
                    <plugins>
                            <plugin>
                                    <groupId>org.springframework.boot</groupId>
                                    <artifactId>spring-boot-maven-plugin</artifactId>
                            </plugin>
                            <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.11.0</version>
        <configuration>
            <source>21</source>
            <target>21</target>
        </configuration>
        </plugin>
                            </plugins>
            </build>
</project>
```

Hibernate.cfg.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
    <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/citizenpassport_db</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password">root</property>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
    <property name="hibernate.hbm2ddl.auto">update</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.format_sql">true</property>
    <mapping class="com.example.entity.Citizen"/>
    <mapping class="com.example.entity.Passport"/>
  </session-factory>
</hibernate-configuration>
```

Citizen.java

```java
package com.example.entity;
import jakarta.persistence.*;
@Entity
public class Citizen {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String name;
    @OneToOne(cascade = CascadeType.ALL)
    @JoinColumn(name = "passport_id", referencedColumnName = "id")
    private Passport passport;
    public Citizen() {}
    public Citizen(String name, Passport passport) {
        this.name = name;
        this.passport = passport;
    }
        public Long getId() {
                return id;
        }
        public void setId(Long id) {
                this.id = id;
        }
        public String getName() {
                return name;
        }
        public void setName(String name) {
                this.name = name;
```

```java
        }
        public Passport getPassport() {
                return passport;
        }
        public void setPassport(Passport passport) {
                this.passport = passport;
        }
}
```

Passport.java

```java
package com.example.entity;
import jakarta.persistence.*;
@Entity
public class Passport {
        @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String passportNumber;
    public Passport() {}
    public Passport(String passportNumber) {
       this.passportNumber = passportNumber;
    }
        public Long getId() {
                return id;
        }
        public void setId(Long id) {
                this.id = id;
        }
        public String getPassportNumber() {
                return passportNumber;
        }
        public void setPassportNumber(String passportNumber) {
                this.passportNumber = passportNumber;
        }
}
```

HibernateUtil.java

```java
package com.example.util;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;
public class HibernateUtil {
        private static final SessionFactory sessionFactory;
  static {
    try {
```

```java
        sessionFactory = new Configuration().configure().buildSessionFactory();
    } catch (Throwable ex) {
        System.err.println("SessionFactory creation failed." + ex);
        throw new ExceptionInInitializerError(ex);
    }
  }
  public static SessionFactory getSessionFactory() {
      return sessionFactory;
  }
}
```

(App.java) CitizenPassportApplication.java

```java
package com.example.app;
import org.hibernate.Session;
import org.hibernate.Transaction;
import com.example.entity.Citizen;
import com.example.entity.Passport;
import com.example.util.HibernateUtil;
public class CitizenPassportApplication {
        public static void main(String[] args) {
    Passport passport = new Passport("P123456");
    Citizen citizen = new Citizen("John Doe", passport);
    Session session = HibernateUtil.getSessionFactory().openSession();
    Transaction tx = null;
    try {
        tx = session.beginTransaction();
        session.save(citizen);
        tx.commit();
        System.out.println("Citizen and Passport saved successfully.");
        Citizen retrieved = session.get(Citizen.class, citizen.getId());
        System.out.println("Retrieved: " + retrieved.getName() +
                ", Passport: " + retrieved.getPassport().getPassportNumber());
    } catch (Exception e) {
        if (tx != null) tx.rollback();
        e.printStackTrace();
    } finally {
        session.close();
        HibernateUtil.getSessionFactory().close();
    }  }
}
```