Case Study Title: Citizen and Passport Management System

SOLUTION

- Open Eclipse.
- Go to File > New > Maven Project.
- Choose Create a simple project (skip archetype selection) → Click Next.
- Group ID: com.example
  Artifact ID: CitizenPassportHibernate
  → Click Finish.

1. // pom.xml

```
<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
                             http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.example</groupId>
    <artifactId>CitizenPassportHibernate</artifactId>
    <version>1.0-SNAPSHOT</version>

    <dependencies>
        <!-- Hibernate Core -->
        <dependency>
            <groupId>org.hibernate.orm</groupId>
            <artifactId>hibernate-core</artifactId>
            <version>6.4.4.Final</version>
        </dependency>

        <!-- JPA API -->
        <dependency>
            <groupId>jakarta.persistence</groupId>
            <artifactId>jakarta.persistence-api</artifactId>
            <version>3.1.0</version>
        </dependency>

        <!-- MySQL JDBC Driver -->
        <dependency>
            <groupId>com.mysql</groupId>
            <artifactId>mysql-connector-j</artifactId>
            <version>8.3.0</version>
        </dependency>

        <!-- Logging -->
        <dependency>
            <groupId>org.jboss.logging</groupId>
            <artifactId>jboss-logging</artifactId>
            <version>3.5.0.Final</version>
        </dependency>
    </dependencies>
</project>
```

2. In MySQL Wrokbench :

```
CREATE DATABASE citizen_passport_db;

USE citizen_passport_db;
```

3. Create 3 packages in src/main/java
com.example.app
com.example.entity
com.example.util

4. Create class Passport----→ com.example.entity

```
package com.example.entity;

import jakarta.persistence.*;

@Entity
public class Passport {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    private String passportNumber;

    @OneToOne(mappedBy = "passport")
    private Citizen citizen;

    // Getters & Setters
    public int getId() { return id; }
    public void setId(int id) { this.id = id; }

    public String getPassportNumber() { return passportNumber; }
    public void setPassportNumber(String passportNumber) {
this.passportNumber = passportNumber; }

    public Citizen getCitizen() { return citizen; }
    public void setCitizen(Citizen citizen) { this.citizen = citizen; }
}
```

5. Create class Citizen ----→ com.example.entity

```
package com.example.entity;

import jakarta.persistence.*;

@Entity
public class Citizen {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    private String name;
```

```
    @OneToOne(cascade = CascadeType.ALL)
    @JoinColumn(name = "passport_id")
    private Passport passport;

    // Getters & Setters
    public int getId() { return id; }
    public void setId(int id) { this.id = id; }

    public String getName() { return name; }
    public void setName(String name) { this.name = name; }

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

6. In src/main/resources-→ create new file named hibernate.cfg.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
 "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
 "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory>
    <!-- DB Config -->
    <property
name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
    <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/citizen_passport_
db</property>
    <property name="hibernate.connection.username">root</property>
    <property
name="hibernate.connection.password">your_password</property>

    <!-- Hibernate Properties -->
    <property
name="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.format_sql">true</property>
    <property name="hibernate.hbm2ddl.auto">update</property>

    <!-- Mapping Classes -->
    <mapping class="com.example.entity.Citizen"/>
    <mapping class="com.example.entity.Passport"/>
  </session-factory>
</hibernate-configuration>
```

7. In package com.example.util: create class HibernateUtil

```
package com.example.util;

import org.hibernate.SessionFactory;
```

```
import org.hibernate.cfg.Configuration;

public class HibernateUtil {
   private static final SessionFactory sessionFactory;

   static {
      try {
         sessionFactory = new Configuration().configure().buildSessionFactory();
      } catch (Throwable ex) {
         throw new ExceptionInInitializerError(ex);
      }
   }

   public static SessionFactory getSessionFactory() {
      return sessionFactory;
   }
}
```

8. Create class App in package com.example.app

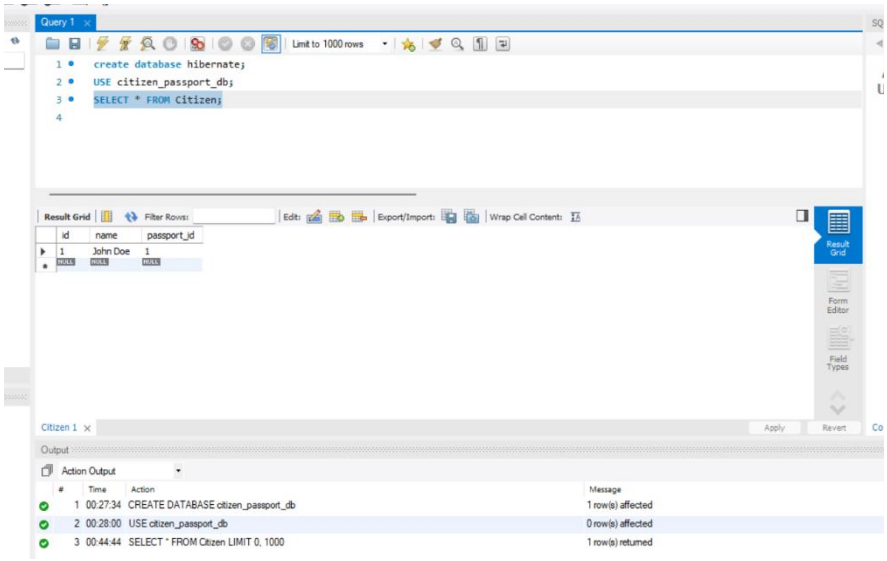
```
package com.example.app;

import com.example.entity.Citizen;
import com.example.entity.Passport;
import com.example.util.HibernateUtil;
import org.hibernate.Session;
import org.hibernate.Transaction;

public class App {
   public static void main(String[] args) {
      Session session = HibernateUtil.getSessionFactory().openSession();
      Transaction tx = session.beginTransaction();

      // Create Passport
      Passport passport = new Passport();
      passport.setPassportNumber("A1234567");

      // Create Citizen
      Citizen citizen = new Citizen();
      citizen.setName("John Doe");
      citizen.setPassport(passport);

      session.persist(citizen); // Will save both due to CascadeType.ALL

      tx.commit();
      session.close();

      System.out.println("Citizen and Passport saved successfully!");
   }
}
```

Right-click App.java → Run As → Java Application.

OUTPUT CONSOLE:

```
Hibernate:
    create table Citizen (
        id integer not null auto_increment,
        name varchar(255),
        passport_id integer,
        primary key (id)
    ) engine=InnoDB
Hibernate:
    create table Passport (
        id integer not null auto_increment,
        passportNumber varchar(255),
        primary key (id)
    ) engine=InnoDB
Hibernate:
    alter table Citizen
       drop index UK_rqf5e7eq8wwfdpflhg6u36pxv
Hibernate:
    alter table Citizen
       add constraint UK_rqf5e7eq8wwfdpflhg6u36pxv unique (passport_id)
Hibernate:
    alter table Citizen
       add constraint FKc915p6km4eh935l56vr4pckfs
       foreign key (passport_id)
       references Passport (id)
Hibernate:
    insert
    into
        Passport
        (passportNumber)
    values
        (?)
Hibernate:
    insert
    into
        Citizen
        (name, passport_id)
    values
        (?, ?)
Citizen and Passport saved successfully!
```

Now go to MySQL workbench
In quries : SELECT * FROM Citizen; ---→ execute



SELECT * FROM Passport; -→ execute