**Multiple employees may share same address**

**N:1**

N - number of employees

1 - single address

Java 9 hibernate 5

Create java project

Configure to maven

**Prepare database**

Create database manytoone;

CREATE TABLE `employee` (

`employee\_id` bigint(10) NOT NULL auto\_increment,

`firstname` varchar(50) default NULL,

`lastname` varchar(50) default NULL,

`cell\_phone` varchar(15) default NULL,

`address\_id` bigint(20) default NULL,

PRIMARY KEY (`employee\_id`),

KEY `FK\_employee` (`address\_id`),

CONSTRAINT `FK\_employee` FOREIGN KEY (`address\_id`) REFERENCES `address` (`address\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1

CREATE TABLE `address` (

`address\_id` bigint(20) NOT NULL auto\_increment,

`street` varchar(50) default NULL,

`city` varchar(50) default NULL,

`state` varchar(50) default NULL,

`country` varchar(50) default NULL,

PRIMARY KEY (`address\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1

**Maven dependencies**

<dependencies>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.2.6.Final</version>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>5.1.10</version>

</dependency>

</dependencies>

**Hibernate Config file**

<?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>

<!-- Database connection settings -->

<property name="connection.driver\_class">com.mysql.jdbc.Driver</property>

<property name="connection.url">jdbc:mysql://localhost:3306/manytoone</property>

<property name="connection.username">root</property>

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

<property name="connection.pool\_size">1</property>

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

<property name="current\_session\_context\_class">thread</property>

<property name="cache.provider\_class">org.hibernate.cache.NoCacheProvider</property>

<property name="show\_sql">true</property>

<property name="hbm2ddl.auto">validate</property>

<mapping class="com.worldclock.Address" />

<mapping class="com.worldclock.Employee" />

</session-factory>

</hibernate-configuration>

**Entity classes**

**1.Address**

package com.worldclock;

import java.util.Set;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.OneToMany;

import javax.persistence.Table;

@Entity

@Table(name = "address")

public class Address {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

@Column(name = "address\_id")

private Long addressId;

@Column(name = "street")

private String street;

@Column(name = "city")

private String city;

@Column(name = "state")

private String state;

@Column(name = "country")

private String country;

@OneToMany(mappedBy = "address")

private Set<Employee> employees;

public Long getAddressId() {

return addressId;

}

public void setAddressId(Long addressId) {

this.addressId = addressId;

}

public String getStreet() {

return street;

}

public void setStreet(String street) {

this.street = street;

}

public String getCity() {

return city;

}

public void setCity(String city) {

this.city = city;

}

public String getState() {

return state;

}

public void setState(String state) {

this.state = state;

}

public String getCountry() {

return country;

}

public void setCountry(String country) {

this.country = country;

}

public Set<Employee> getEmployees() {

return employees;

}

public void setEmployees(Set<Employee> employees) {

this.employees = employees;

}

}

**2.Employee**

package com.worldclock;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.ManyToOne;

import javax.persistence.Table;

@Entity

@Table(name="employee")

public class Employee {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

@Column(name="employee\_id")

private Long employeeId;

@Column(name="firstname")

private String firstname;

@Column(name="lastname")

private String lastname;

@Column(name="cell\_phone")

private String cellphone;

@ManyToOne

@JoinColumn(name="address\_id")

private Address address;

public Address getAddress() {

return address;

}

public void setAddress(Address address) {

this.address = address;

}

public Employee() {}

public Employee(String firstname, String lastname, String phone) {

this.firstname = firstname;

this.lastname = lastname;

this.cellphone = phone;

}

public Long getEmployeeId() {

return employeeId;

}

public void setEmployeeId(Long employeeId) {

this.employeeId = employeeId;

}

public String getFirstname() {

return firstname;

}

public void setFirstname(String firstname) {

this.firstname = firstname;

}

public String getLastname() {

return lastname;

}

public void setLastname(String lastname) {

this.lastname = lastname;

}

public String getCellphone() {

return cellphone;

}

public void setCellphone(String cellphone) {

this.cellphone = cellphone;

}

}

**Main class**

package com.worldclock;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.boot.Metadata;

import org.hibernate.boot.MetadataSources;

import org.hibernate.boot.registry.StandardServiceRegistry;

import org.hibernate.boot.registry.StandardServiceRegistryBuilder;

import org.hibernate.cfg.Configuration;

import org.hibernate.service.ServiceRegistry;

import org.hibernate.boot.registry.StandardServiceRegistryBuilder;

public class ManageEmployee {

private static SessionFactory sf;

private static StandardServiceRegistry standardServiceRegistry;

@SuppressWarnings("unchecked")

public static void main(String[] args) {

try {

standardServiceRegistry = new StandardServiceRegistryBuilder()

.configure()

.build();

// Create MetadataSources

MetadataSources metadataSources = new MetadataSources(standardServiceRegistry);

// Create Metadata

Metadata metadata = metadataSources.getMetadataBuilder().build();

// Create SessionFactory

sf = metadata.getSessionFactoryBuilder().build();

} catch (Throwable ex) {

System.err.println("Failed to create sessionFactory object." + ex);

throw new ExceptionInInitializerError(ex);

}

System.out.println("Hibernate One to Many Mapping Example Using Annotation ");

Session session = sf.openSession();

session.beginTransaction();

Address address = new Address();

address.setStreet("sindhiya street");

address.setCity("Gwalior");

address.setState("Madhya Pradesh");

address.setCountry("India");

session.save(address);

Employee e1 = new Employee("Ankit", "Sharma", "9999999999");

Employee e2 = new Employee("Ankit", "Kaushal", "3333333333");

e1.setAddress(address);

e2.setAddress(address);

session.save(e1);

session.save(e2);

session.getTransaction().commit();

session.close();

}

}

Add **--add-modules java.xml.bind** module in VM arguments

Run class which contains main method