**Create Entity Classes:**

**// Person.java**

**import javax.persistence.\*;**

**import java.util.List;**

**@Entity**

**public class Person {**

**@Id**

**@GeneratedValue(strategy = GenerationType.IDENTITY)**

**private Long id;**

**private String name;**

**@OneToMany(mappedBy = "person", fetch = FetchType.LAZY)**

**private List<Phone> phones;**

**// getters and setters**

**}**

**// Phone.java**

**import javax.persistence.\*;**

**@Entity**

**public class Phone {**

**@Id**

**@GeneratedValue(strategy = GenerationType.IDENTITY)**

**private Long id;**

**private String number;**

**@ManyToOne(fetch = FetchType.LAZY)**

**@JoinColumn(name = "person\_id")**

**private Person person;**

**// getters and setters**

**}**

**Configure Hibernate:**

<!-- hibernate.cfg.xml -->

<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<!-- JDBC Database connection settings -->

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

<property name="hibernate.connection.url">jdbc:mysql://localhost:3306/your\_database</property>

<property name="hibernate.connection.username">your\_username</property>

<property name="hibernate.connection.password">your\_password</property>

<!-- JDBC connection pool settings -->

<property name="hibernate.c3p0.min\_size">5</property>

<property name="hibernate.c3p0.max\_size">20</property>

<property name="hibernate.c3p0.timeout">300</property>

<property name="hibernate.c3p0.max\_statements">50</property>

<property name="hibernate.c3p0.idle\_test\_period">3000</property>

<!-- Specify dialect -->

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

<!-- Echo all executed SQL to stdout -->

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

<!-- Drop and re-create the database schema on startup -->

<property name="hibernate.hbm2ddl.auto">update</property>

<!-- Mention annotated class -->

<mapping class="Person"/>

<mapping class="Phone"/>

</session-factory>

</hibernate-configuration>

**Test Lazy Loading:**

public class HibernateTest {

public static void main(String[] args) {

SessionFactory factory = new Configuration().configure("hibernate.cfg.xml").buildSessionFactory();

Session session = factory.getCurrentSession();

try {

session.beginTransaction();

// Create and save a Person with phones

Person person = new Person();

person.setName("John");

Phone phone1 = new Phone();

phone1.setNumber("123-456-7890");

phone1.setPerson(person);

Phone phone2 = new Phone();

phone2.setNumber("987-654-3210");

phone2.setPerson(person);

person.setPhones(Arrays.asList(phone1, phone2));

session.save(person);

session.getTransaction().commit();

// Close the session

session.close();

// Open a new session for retrieval

session = factory.getCurrentSession();

session.beginTransaction();

// Load the Person with lazy-loaded phones

Person loadedPerson = session.get(Person.class, person.getId());

// At this point, phones collection is not loaded from the database yet

// Accessing the phones collection triggers the lazy loading

List<Phone> phones = loadedPerson.getPhones();

for (Phone phone : phones) {

System.out.println("Phone Number: " + phone.getNumber());

}

session.getTransaction().commit();

} finally {

factory.close();

}

}

}