

```
1 import model.Address;
2 import model.Cohort;
3 import model.Department;
4 import model.Teacher;
5 import org.hibernate.Session;
6 import org.hibernate.SessionFactory;
7 import org.hibernate.Transaction;
8 import org.hibernate.cfg.Configuration;
9
10 import java.util.ArrayList;
11 import java.util.HashSet;
12 import java.util.Set;
13
14 public class App {
15     public static void main(String[] args) {
16         //      manyToOne();
17         //      oneToMany();
18         //      oneToOne();
19         manyToMany();
20
21     }
22
23
24 public static void manyToMany(){
25     SessionFactory factory = new Configuration().
configure().buildSessionFactory();
26     Session session = factory.openSession();
27     Transaction t = session.beginTransaction();
28
29     Cohort class1 = new Cohort("Java Developer", "14
weeks");
30     Cohort class2 = new Cohort("FullStack Developer"
, "7 weeks");
31     Cohort class3 = new Cohort("Python Developer", "
12 weeks");
32
33     session.persist(class1);
34     session.persist(class2);
35     session.persist(class3);
36
37     Set<Cohort> classSet1 = new HashSet<>();
```

```
38     classSet1.add(class1);
39     classSet1.add(class2);
40     classSet1.add(class3);
41
42     Set<Cohort> classSet2 = new HashSet<>();
43     classSet2.add(class1);
44     classSet2.add(class2);
45     classSet2.add(class3);
46
47     Set<Cohort> classSet3 = new HashSet<>();
48     classSet3.add(class1);
49     classSet3.add(class2);
50     classSet3.add(class3);
51
52     Teacher t1 = new Teacher("100", "Haseeb",
classSet1);
53     Teacher t2 = new Teacher("200", "jenny",
classSet2);
54     Teacher t3 = new Teacher("200", "Charlie",
classSet3);
55     session.persist(t1);
56     session.persist(t2);
57     session.persist(t3);
58
59     t.commit();
60
61
62 }
63     public static void manyToOne() {
64         SessionFactory factory = new Configuration().
configure().buildSessionFactory();
65         Session session = factory.openSession();
66         Transaction transaction = session.
beginTransaction();
67
68         Department dept1 = new Department("IT");
69         Department dept2 = new Department("HR");
70
71         Teacher t1 = new Teacher("1000", "MHaseeb",
dept1);
72         Teacher t2 = new Teacher("2220", "Shahparan"
```

```
72 , dept1);
73     Teacher t3 = new Teacher("3000", "James",
    dept1);
74     Teacher t4 = new Teacher("40000", "Joseph",
    dept2);
75
76     session.persist(dept1);
77     session.persist(dept2);
78
79     session.persist(t1);
80     session.persist(t2);
81     session.persist(t3);
82     session.persist(t4);
83     transaction.commit();
84 }
85
86     public static void oneToMany() {
87         SessionFactory factory = new Configuration
    ().configure().buildSessionFactory();
88         Session session = factory.openSession();
89         Transaction t = session.beginTransaction();
90
91         Teacher t1 = new Teacher("1000", "MHaseeb");
92         Teacher t2 = new Teacher("2220", "Shahparan"
    );
93         Teacher t3 = new Teacher("3000", "James");
94         Teacher t4 = new Teacher("40000", "Joseph");
95         Teacher t5 = new Teacher("200", "Ali");
96
97         ArrayList<Teacher> teacherArrayList = new
    ArrayList<>();
98         teacherArrayList.add(t1);
99         teacherArrayList.add(t2);
100        teacherArrayList.add(t3);
101        teacherArrayList.add(t4);
102        teacherArrayList.add(t5);
103
104        session.persist(t1);
105        session.persist(t2);
106        session.persist(t3);
107        session.persist(t4);
```

```
108         session.persist(t5);
109
110         Department department = new Department();
111         department.setDeptName("Development");
112         department.setTeacherList(teacherArrayList);
113
114         session.persist(department);
115         t.commit();
116     }
117
118     public static void oneToOne(){
119         System.out.println("Maven + Hibernate + SQL
120         One to One Mapping Annotations");
121
122         SessionFactory factory = new Configuration
123         ().configure().buildSessionFactory();
124         Session session = factory.openSession();
125         Transaction t = session.beginTransaction();
126         Address a1 = new Address("27th street", "NYC
127         ", "NY", 11103);
128         Address a2 = new Address("28th street", "
129         Buffalo", "NY", 15803);
130
131         Teacher t1 = new Teacher("1000", "MHaseeb");
132         Teacher t2 = new Teacher("2220", "Shahparan"
133         );
134
135         t1.setAddress(a1);
136         t2.setAddress(a2);
137
138         session.persist(a1);
139         session.persist(t1);
140         session.persist(a2);
141         session.persist(t2);
142
143         t.commit();
144     }
145 }
```

```
1 package model;
2
3 import jakarta.persistence.*;
4
5 @Entity
6 @Table(name="cohort")
7 public class Cohort {
8     @Id
9     @GeneratedValue(strategy = GenerationType.
IDENTITY)
10     private int cohortId;
11     private String cohortName;
12     private String duration;
13
14     public Cohort(String cohortName, String duration
) {
15         this.cohortName = cohortName;
16         this.duration = duration;
17     }
18
19     public Cohort() {
20
21     }
22
23     public int getCohortId() {
24         return cohortId;
25     }
26
27     public void setCohortId(int cohortId) {
28         this.cohortId = cohortId;
29     }
30
31     public String getCohortName() {
32         return cohortName;
33     }
34
35     public void setCohortName(String cohortName) {
36         this.cohortName = cohortName;
37     }
38
39     public String getDuration() {
```

```
40         return duration;
41     }
42
43     public void setDuration(String duration) {
44         this.duration = duration;
45     }
46 }
47
```

```
1 package model;
2
3 import jakarta.persistence.*;
4
5 import java.io.Serial;
6 import java.io.Serializable;
7
8 @Entity
9 @Table
10 public class Address implements Serializable {
11     @Serial
12     private static final long serialVersionUID = 1L;
13     @Id
14     @GeneratedValue(strategy = GenerationType.
IDENTITY)
15     private int addressId;
16     private String street;
17     private String city;
18     private String state;
19     private int zipCode;
20     public Address(){}
21
22     public Address(String street, String city, String
state, int zipCode) {
23         this.street = street;
24         this.city = city;
25         this.state = state;
26         this.zipCode = zipCode;
27     }
28
29     public int getAddressId() {
30         return addressId;
31     }
32
33     public void setAddressId(int addressId) {
34         this.addressId = addressId;
35     }
36
37     public String getStreet() {
38         return street;
39     }
```

```
40
41     public void setStreet(String street) {
42         this.street = street;
43     }
44
45     public String getCity() {
46         return city;
47     }
48
49     public void setCity(String city) {
50         this.city = city;
51     }
52
53     public String getState() {
54         return state;
55     }
56
57     public void setState(String state) {
58         this.state = state;
59     }
60
61     public int getZipCode() {
62         return zipCode;
63     }
64
65     public void setZipCode(int zipCode) {
66         this.zipCode = zipCode;
67     }
68 }
69
```



```
1 package model;
2
3 import java.io.Serial;
4 import java.io.Serializable;
5 import java.util.Set;
6
7 import jakarta.persistence.*;
8
9 @Entity
10 @Table
11 public class Teacher implements Serializable {
12     @Serial
13     private static final long serialVersionUID = 1L;
14     @Id
15     @GeneratedValue(strategy = GenerationType.
        IDENTITY)
16     private int teacherId;
17     private String salary;
18     private String teacherName;
19     // @ManyToOne
20     // private Department department;
21
22     @OneToOne(cascade = CascadeType.ALL)
23     private Address address;
24     public Address getAddress(){
25         return address;
26     }
27
28     public void setAddress(Address address) {
29         this.address = address;
30     }
31     @ManyToMany(targetEntity = Cohort.class)
32     private Set<Cohort> cohort;
33
34     public Set<Cohort> getCohort() {
35         return cohort;
36     }
37
38     public void setCohort(Set<Cohort> cohort) {
39         this.cohort = cohort;
40     }
```

```
41
42     public Teacher(String salary, String teacherName
43     , Set<Cohort> cohort) {
44         this.salary = salary;
45         this.teacherName = teacherName;
46         this.cohort = cohort;
47     }
48     public Teacher(int teacherId, String salary,
49     String teacherName) {
50         super();
51         this.teacherId = teacherId;
52         this.salary = salary;
53         this.teacherName = teacherName;
54     }
55     public Teacher() {
56     }
57
58     public Teacher(String salary, String teacherName
59     , Department department){
60         this.salary=salary;
61         this.teacherName =teacherName;
62         //      this.department=department;
63     }
64     public Teacher(String salary, String teacherName
65     ){
66         this.salary=salary;
67         this.teacherName =teacherName;
68     }
69     //      public Department getDepartment(){
70     //          return department;
71     //      }
72     public int getTeacherId(){
73         return teacherId;
74     }
75     public void setTeacherId(int teacherId) {
76         this.teacherId = teacherId;
77     }
78     public String getSalary() {
```

```
78         return salary;
79     }
80     public void setSalary(String salary) {
81         this.salary = salary;
82     }
83     public String getTeacherName() {
84         return teacherName;
85     }
86     public void setTeacherName(String teacherName) {
87         this.teacherName = teacherName; }
88
89
90 }
91
```

```
1 package model;
2 import jakarta.persistence.*;
3 import java.io.Serial;
4 import java.io.Serializable;
5 import java.util.List;
6
7 @Entity
8 @Table
9 public class Department implements Serializable {
10     @Serial
11     private static final long serialVersionUID=1L;
12
13     @Id
14     @GeneratedValue(strategy = GenerationType.
IDENTITY)
15     private int deptId;
16     private String deptName;
17     @OneToMany(targetEntity= Teacher.class, cascade
= {CascadeType.ALL})
18     private List<Teacher> teacherList;
19     public Department(int deptId, String deptName){
20         super();
21         this.deptId=deptId;
22         this.deptName=deptName;
23     }
24
25     public List<Teacher> getTeacherList() {
26         return teacherList;
27     }
28     public void setTeacherList(List<Teacher>
teacherList) {
29         this.teacherList = teacherList;
30     }
31
32
33     public Department(){}
34
35     public Department(String deptName){
36         this.deptName=deptName;
37     }
38
```

```
39     public int getDeptId(){
40         return deptId;
41     }
42
43     public void setDeptId(int deptId) {
44         this.deptId = deptId;
45     }
46
47     public String getDeptName() {
48         return deptName;
49     }
50
51     public void setDeptName(String deptName) {
52         this.deptName = deptName;
53     }
54 }
55
```