Today’s Objectives

1. Revise with Category 1 <----> \* Product
2. Cascading
3. Orphan removal
4. JPA Fetch Types n LazyInitializationException
5. Uni dir one to one (User 1--- >1 Address)
6. Uni dir many to many (Tag \*------>\* Restaurant)
7. Components (value types) – Embeddables
8. JPA Inheritance

0

**Project Tip**

Either create one - many n many - to one bi dir association

OR

Simply create **many 🡪 one** uni dir association (more optimized since requires lesser no of queries, since many side : owning side)

Easier to configure – uni dir

Use case of bi dir association - cascading

1. **Revise**

How will you establish , a bi dir Category 1 <----> \* Product ?

In Category class

….+

Private List<Product> Products=new ArrayList<>();

In Product class

….+

Private Category myCategory;

**Solve**

Add Product to the Category

i/p – Category id , Product details

Dev steps in Product DAO implementation class

1. Get Category by its id
2. If not null -> Category exists
3. Establish bi-dir association between pojos

Eg – Category1.addProduct(Product1);

1. Tx.commit();
2. What happens in DB ? – rec inserted in Products table with FK assigned !

**Gavin King’s suggestion**

**Add helper methods** in the Parent entity – to add / remove a child entity -instead of writing this code in DAO layer.

In Category POJO –

Public void addProduct(Product p)

{

this.Products.add(p);

p.setMyCategory(this);

}

public void removeProduct(Product p)

{

this.Products.remove(p);

p.setMyCategory(null);

}

1. How to **automatically persist** food item , after its added in the association (i.e in the List , in Restaurant) ?

Refer- "day11-data\day11\_help\readmes\Regarding Cascading in JPA n Hibernate.txt"

1. **Solve objective**

**Remove specified food item from the specified restaurant**

i/p – food item id , restaurant id

o/p - mesg

DB action – child rec should be deleted from food items table

Any problem – Instead of deleting post rec , hibernate simply assigns restaurant id=null (update)

Cause – orphan removal is not specified yet (default value – false)

Solution – Add in @OneToMany ,

orphanRemoval=true

1. **Solve objective**

**Get Restaurant details , by its name**

i/p **Restaurant** name

o/p **Restaurant** details or exception in case of invalid name

1. **Solve objective**

**Get Restaurant n food item details, by its name**

i/p **Restaurant** name

o/p **Restaurant** + food item details or exception in case of invalid name

Any problems n solution?

Problem – LazyInitializationException , different solutions

(change fetchType –eager | access size of the collection within session scope | join fetch)

**6.Solve objective**

**Establish bi dir association - one to one (User 1<--- >1 Address)**

**Entity Layer –**

**User entity – inverse**

**State – firstName, …..+**

**@OneToOne(mappedBy=” owner”,cascade=CascadeType.ALL,orphanRemoval=true)**

**Private Address myAddress;**

**Address –owning side**

**Line1 , line2….+**

**@OneToOne**

**@JoinColumn(name=”user\_id”)**

**Private User owner;**

**Establish uni dir association - one to one (User 1--- >1 Address)**

**In User class –**

**@OneToOne(cascade=ALL,orphanRemoval=true)**

**@JoinColumn(name=”addr\_id”)**

**Private Address myAddress;**

**In Address class –**

**None !**

**Solve .’**

**What will be Address class definition ?**

Public class Address extends BaseEntity{…}

**What will you add in User class ?**

**…..**

**+**

**What will you add in Address class ?**

**Adr details**

**Solve –**

**Assign address to the existing user**

**i/p –adr details + user id**

**o/p – mesg**

**DB – rec gets inserted in address table .**

**Users updated – adr id.**

**Solve –lab work !**

1. **Display user details + address details**

**i/p – user id**

**o/p - user details + address details**

**Hint – join fetch**

1. **Display user details**

**i/p – user id**

**o/p –only user details**

1. **Display last names of all users from the specified city**

**i/p – city name**

**o/p – list of last names**

**JPQL –**

1. **Uni dir many to many (Tag \*------>\* Restaurant)**

What will you add in Tag class ?

@ManyToMany - mandatory

@JoinTable(name , joinColumns,inverseJoinColumns)

Private Set< **Restaurant** > restaurants=new HashSet<>();

What will you add in **Restaurant** class ?

NONE !

Solve

1. Create a tag

i/p – tag name

DB – rec inserted in tags table (session.persist(nesTag))

1. Add specified restaurant to the specified tag

i/p – tag id , post id

DB – rec should be inserted in join table – with FKs assigned .

1. **Components (value types) – Embeddables | Elements**

User HAS-A Adhar Card : demo of entity n value type (composite)

User HAS-A Language Skills : string (multiple languages)

- collection of basic value type

1. JPA Inheritance

* Single table inheritance strategy