```
Language(java)
Technology(JDBC)
Framework ( ORMTool----> Hibernate)
ORM ===> Do operation only in objects
Hibernate(Object<---> ROW)
1.SRO(Single Row Operation)
           a. save()/persist()
           b. get()/load()
           c. update(), saveOrUpdate()
           d. delete()
2.Bulk operation(working with more than one Row)
           a. HQL/JPQL
           b. NativeQuery
           c. Criterian API(mostly used by java developers)
Property
     hibernate.hbm2ddl.auto
           ddl=Data definition language(create /alter/drop in SQL)
           it has four possible value.
Those are:-
a. validate (default value)
b. create
c. update
d. create-drop
A. validate: - In this case hibernate creates no tables programmer has to create or
modify tables manually.
                   It is only default value.
             hibernate creates always new tables, if table exist then it will be
B. create:-
drop.
             It creates new table if table not exits else uses same tables.
C. update:-
                   This attribute is commonly used.
D. create-drop: This option is used for testing process not in development,
                        it creates a new table an performs operation and at last
table will be drop.
                        It is just for POC(proof of concept) or for learning
purpose.
Hibernate Persistence operation
Objects used in hibernate
     a. Configuration
     b. SessionFactory(heavy weight)
     c. Session
     d. Transaction
1.SRO(Single Row Operation)
           a. save()/persist()
           b. get()/load()
           c. update(), saveOrUpdate()
           d. delete()
a. save()/persist() => These methods are used to perform insert operation.
What is the difference between save() method and persist() method?
In Hibernate applications, save() method can be used to insert a record into the
```

Database table and it will return

Primary Key value of the inserted record.

This method is from hibernate api.

public Serializable save(Object obj)throws

HibernateException

In Hibernate applications, persist() method can be used to insert a record into database table and it will not return any value.

This method is from JPA specification.

public void persist(Object obj)throws HibernateException

Note:

@DynamicInsert(value=true) and @DynamicUpdate(value=true)

- => It is used to generate the dynamic query, based on the fields used in the Entity Object
- => If we want our table name and column name to be same as entity name and field name then no need to use

@Table(name='') and @Column(name='',length='')

=> During the creation of SessionFactory object, by refereing to mapping information hibernate will create

pregenerated sql queries for insert, update, delete, select by using all the fields of the entity class.

=> If we want to avoid that and if we want query to be generated based on the field injection we do on the entity class

then we need to use @DynamicInsert(value='true') or @DynamicUpdate(value='true')

Performing select operation in hibernate

b. get()/load()

What are the differences between get(-) method and load(-) method? Ans:

1. get() method can be used to retrive a record from database table if the record is existed.

If the required record is not existed then get() method will return null value.

public Object get(String class_Name, Serializable pk_Val)
public Object get(Class class_Type, Serializable pk_Val)

- 2. get() method is able to perform eager or early loading, that is, it will interact with database directly and it will retrive data and return to Hibernate application in the form of Object on the method call.
- 1. load() method can be used to retreive a record from database table if the record is existed.

If the required record is not existed then load() method will rise an Exception like HibernateException.

public Object load(String class_Name, Serializable pk_Val)
public Object load(Class class_Type, Serializable pk_val)

2.load() method will perform Lazy or late Loading , that is, when we access load() method then a duplicate object will be

created with the primary key value without interacting with database(proxy object).

When we use other properties of the Object then only it will fetch data from database table and return that data to

Java application.

update(), saveOrUpdate()

updating a record can be done in 3 ways

a. update total object(not prefered)

remember the object exists with the id and update the

entire object.

if the id doesn't exists it would return

'OptimisticLockException'.

- b. load the record and update(very useful for partial updation)
- c. load and modify the record without using update()[Synrhonization
 would exist b/w row and object]

saveOrUpdate()

This method call would first perform

a. select operation

if record found then it performs update operation, otherwise it performs insert operation

What is the difference between update() method and saveOrUpdate() method? Ans:

Where update(-) method will perform updation on a record in database table if the specified record is existed otherwise it will rise an Exception.

public void update(Object obj)throws HibernateException

Where saveOrUpdate(-) method will insert the specified record in database table if the specified record is not existed .

If the specified record is existed in database table then it will update the record.

public void saveOrUpdate(Object obj)throws HibernateException

delete()

.....

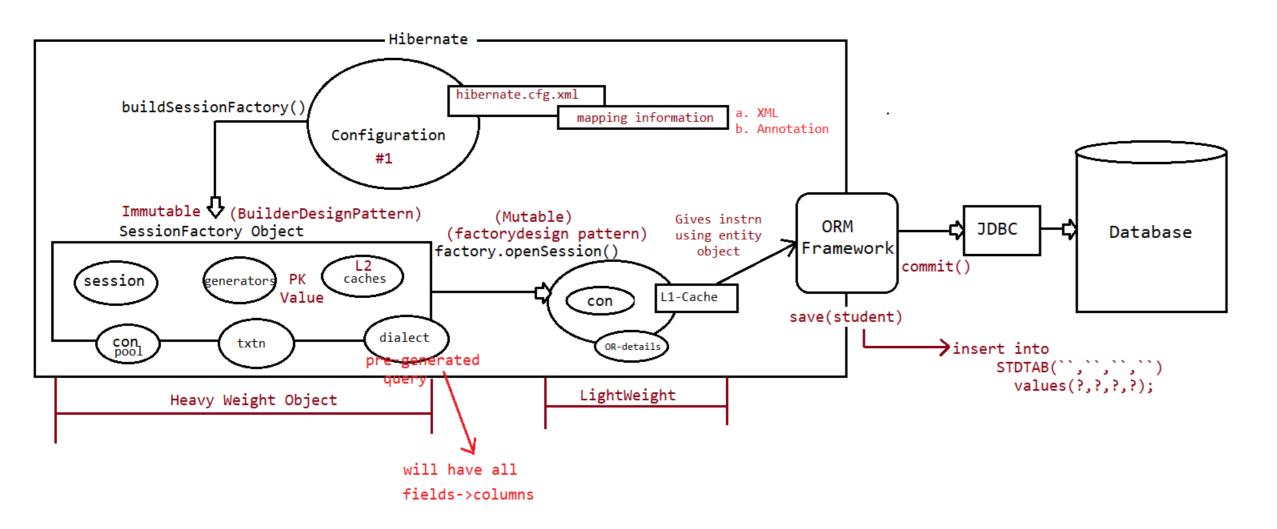
It can be done in 2 ways

a. perform deletion by supplied id value directly

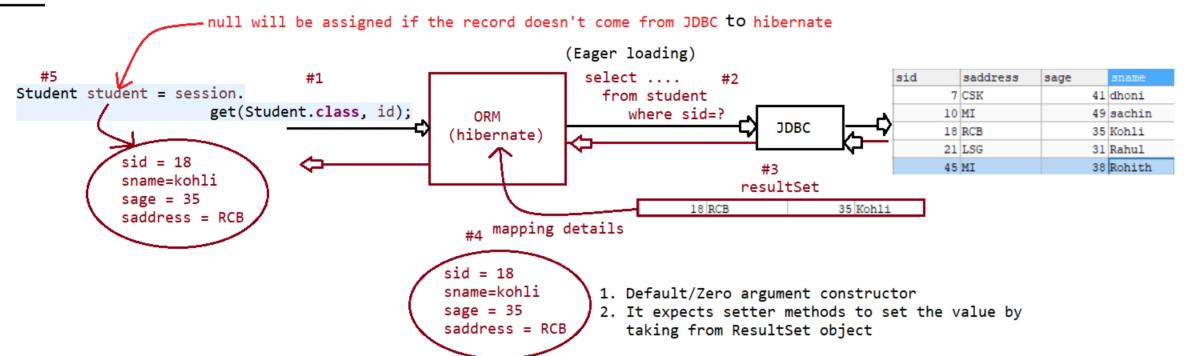
It generates sql select query, if the record exists it would generate delete query, otherwise it won't

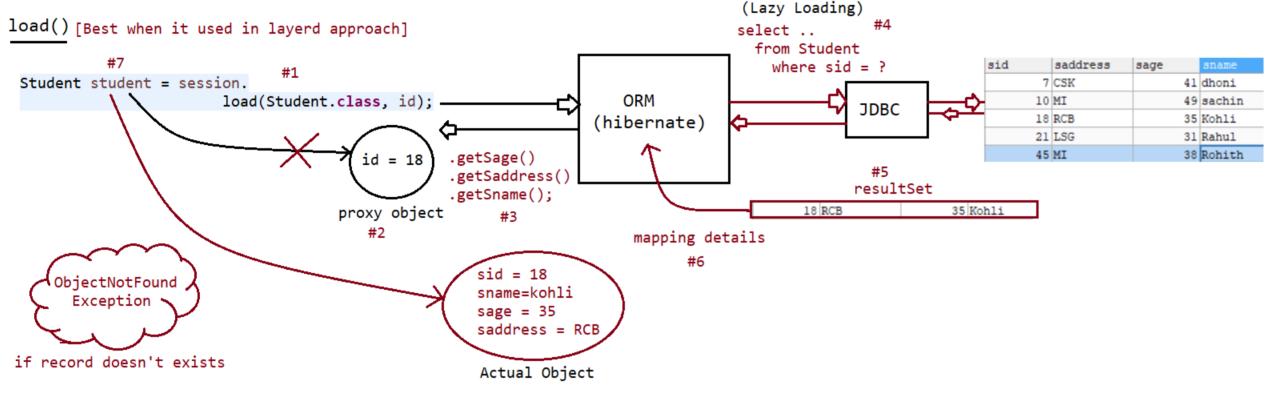
generate delete query and it won't throw any Exception.

b. load and delete the record[prefered]

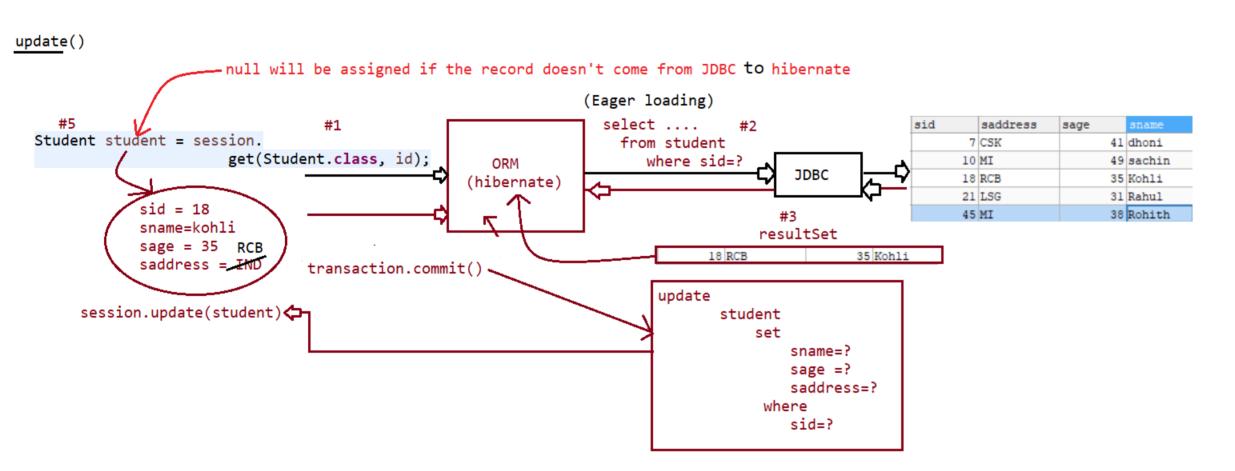


get() [Best when we use in standlone projects]





 Default/Zero argument constructor
 It expects setter methods to set the value by taking from ResultSet object



updating the record with respect to transaction

