Hibernate

내생각

스프링에서 filter처리를 해놓으면

우선 하이버네이트 session을 close 한후

getCurrentSession 할때 open을 하여 사용 한다 그렇기 그렇기 때문에 메모리 누수가 없다

어노테이션

|  |
| --- |
| @Entity 하이버네이트 객체  @Id 필드 아이디값이라고 선언  @GeneratedValue // @Id 일때 어떤식으로 아이디 처리할지..선언 값  @Column // 컬럼 명지정  @AttributeOverrite // 안쪽 컬럼명등 내용을 무시하고 오버라이딩한 값으로 셋팅  @Embeded // 임베드해는 쪽에서  임베드 당하는 클래스는 @Embededble?  @ElementCollection Embeded하는애들을 컬렉션 으로 넣을 때  @JoinTable 조인테이블 선언 안쪽에 조인컬럼 넣을수 있음 joinColumns….  @JoinColumn 조인컬럼 넣어준다.  @GenericGenerator // 제너릭 재너레이터  @CollectionId // 컬렉션의 아이디값 즉 인덱스값? 등을 넣어주는거 넣어줌에 따라 유일키 생성됨 |

Fetch

|  |
| --- |
| LAZY : 지연로딩  EAGER : 즉시로딩 |

팁

|  |
| --- |
| Fetch Mode가 Lazy 경우 지연로딩인경우  로딩이 안된상태에서 sesison을 닫으면 오류난다 못가져오니깐 |

조인

|  |
| --- |
| 조인을 걸게되면 자동으로 가운데 맵핑 테이블에 키값맵핑 테이블이 생긴다  (oneToOne은 안생김// 마스터 테이블에 컬럼이 생김 ) |

@oneToOne

@oneToMany

|  |
| --- |
| 1:N 조인을 걸게되면  가운데 키맵핑 테이블이 생긴다 master\_slave 테이블  @JoinTable로 @JoinColumn 처리로 수정가능 컬럼명  @JoinTable  name = “table name”  joincolumns = @JoinColumn,  inverseJoinColumns = @JoinColumn |

@ManyToOne

|  |
| --- |
| 1:N 에 N쪽에서 걸수 있다 |

Entity상속

|  |
| --- |
| 상속처리를 하게되면 DTYPE 컬럼에 값이 들어간다  DTYPE이름변경하고싶을 때  superclass에서 @DiscriminatorColumn을 지정하면된다  자식 class쪽에서는 @DiscriminatorValue 로 지정하면 그값이 들어간다 |
| 야무진 상속처리를 위해 @Inherithance를 사용하라  @Inheriteance 의 strategy를 SIngleTable로 |

스프링 셋팅

|  |
| --- |
| Gradle  compile group : 'org.hibernate', name:'hibernate-core', version : '4.3.6.Final' |
| dataSource  <bean id=*"dataSourceSpied"* class=*"org.springframework.jndi.JndiObjectFactoryBean"*>  <property name=*"jndiName"* value=*"jdbc/cbisOracle"*/>  <property name=*"resourceRef"* value=*"true"*></property>  </bean>  <bean id=*"dataSource"* class=*"net.sf.log4jdbc.Log4jdbcProxyDataSource"*>  <constructor-arg ref=*"dataSourceSpied"* />  <property name=*"logFormatter"*>  <bean class=*"net.sf.log4jdbc.tools.Log4JdbcCustomFormatter"*>  <property name=*"loggingType"* value=*"MULTI\_LINE"* />  <property name=*"sqlPrefix"* value=*"SQL:::"*/>  </bean>  </property>  </bean> |
| Hibernate  <?xml version=*"1.0"* encoding=*"utf-8"*?>  <beans xmlns=*"http://www.springframework.org/schema/beans"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xsi:schemaLocation=*"http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-2.5.xsd"*>  <!--  http://docs.jboss.org/hibernate/core/3.6/reference/en-US/html\_single/#persistent-classes-dynamicmodels  https://docs.jboss.org/hibernate/orm/3.3/reference/ko-KR/html\_single/  위링크 예제파일 잘나와있음  -->  <!-- <bean id="sessionFactory" class="org.springframework.orm.hibernate4.LocalSessionFactoryBean" lazy-init="false"> -->  <bean id=*"sessionFactory"* class=*"org.springframework.orm.hibernate4.LocalSessionFactoryBean"* lazy-init=*"false"*>  <property name=*"dataSource"* ref=*"dataSource"* />    <!-- <property name="configLocation"> -->  <!-- <value>classpath:/egovframework/hibernate/hibernate.cfg.xml</value> -->  <!-- </property> -->  <property name=*"mappingLocations"*>  <value>classpath:/hibernate/\*.hbm.xml</value>  </property>    <property name=*"hibernateProperties"*>  <props>  <!--hibernate dialect-START-->  <prop key=*"hibernate.dialect"*>org.hibernate.dialect.Oracle10gDialect</prop>  <!-- <prop key="hibernate.dialect">org.hibernate.dialect.MySQL5Dialect</prop> -->  <!--hibernate dialect-END-->  <prop key=*"hibernate.format\_sql"*>true</prop>  <prop key=*"hibernate.show\_sql"*>true</prop>  <prop key=*"hibernate.use\_sql\_comments"*>true</prop>  <prop key=*"hibernate.jdbc.batch\_size"*>5</prop>  </props>  </property>    <!-- <property name="annotatedClasses">  <list>  <value>egovframework.rte.rex.cgr.service.CategoryVO</value>  </list>  </property>  --> <property name=*"packagesToScan"*>  <list>  <value>com.omnicns.\*</value>  </list>  </property>  </bean>      <bean id=*"transactionManager"*  class=*"org.springframework.orm.hibernate4.HibernateTransactionManager"*>  <property name=*"sessionFactory"* ref=*"sessionFactory"* />  </bean>    </beans> |
| DAO  @Autowired  **public** SessionFactory sessionFactory; |
| Filter  <!-- 하이버네이트opensession getcurrentSession을위해.. -->  <filter>  <filter-name>OpenSessionInViewFilter</filter-name>  <filter-class>org.springframework.orm.hibernate4.support.OpenSessionInViewFilter</filter-class>  <init-param>  <param-name>sessionFactoryBeanName</param-name>  <param-value>sessionFactory</param-value>  </init-param>  </filter>  <filter-mapping>  <filter-name>OpenSessionInViewFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping> |
| 하이버네이트 validation off  <property name=*"hibernateProperties"*>  <props>  <!--hibernate dialect-START-->  <prop key=*"hibernate.dialect"*>org.hibernate.dialect.Oracle10gDialect</prop>  <!-- <prop key="hibernate.dialect">org.hibernate.dialect.MySQL5Dialect</prop> -->  <!--hibernate dialect-END-->  <prop key=*"hibernate.format\_sql"*>true</prop>  <prop key=*"hibernate.show\_sql"*>true</prop>  <prop key=*"hibernate.use\_sql\_comments"*>true</prop>  <prop key=*"hibernate.jdbc.batch\_size"*>50</prop>    <prop key=*"spring.jpa.properties.javax.persistence.validation.mode"*>false</prop>  <prop key=*"hibernate.query.startup\_check"*>false</prop>  <prop key=*"hibernate.validator.apply\_to\_ddl"*>false</prop>  <prop key=*"hibernate.validator.autoregister\_listeners"*>false</prop>  <prop key=*"javax.persistence.validation.mode"*>none</prop>    </props>  </property> |
| 스프링 시큐리트 보다 먼저 필터를 걸어줘야지  시큐리트에서 getCurrentSession을 할수 있다.  <!-- 하이버네이트opensession getcurrentSession을위해.. -->  <filter>  <filter-name>OpenSessionInViewFilter</filter-name>  <filter-class>org.springframework.orm.hibernate4.support.OpenSessionInViewFilter</filter-class>  <init-param>  <param-name>sessionFactoryBeanName</param-name>  <param-value>sessionFactory</param-value>  </init-param>  </filter>  <filter-mapping>  <filter-name>OpenSessionInViewFilter</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping>        <!-- 스프링 시큐어리티-->  <filter>  <filter-name>springSecurityFilterChain</filter-name>  <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>  </filter>  <filter-mapping>  <filter-name>springSecurityFilterChain</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping>    <!-- 사용자별 Content request,response,session 등 들고다니도록 하는것 -->  <filter>  <filter-name>Content</filter-name>  <filter-class>com.omnicns.web.content.ContentFilter</filter-class>  </filter>  <filter-mapping>  <filter-name>Content</filter-name>  <url-pattern>/\*</url-pattern>  </filter-mapping>    <!-- 스프링 세션 리스너 등록 -->  <listener>  <listener-class>org.springframework.security.web.session.HttpSessionEventPublisher</listener-class>  </listener> |

//조인예제 https://docs.jboss.org/hibernate/core/3.3/reference/ko-KR/html/queryhql.html

//http://www.mkyong.com/spring-security/spring-security-hibernate-annotation-example/

//http://javaonlineguide.net/2015/03/spring-4-and-hibernate-4-integration-example-project-in-eclipse-without-maven-support.html

//http://javacan.tistory.com/entry/105 생명주기

//<img src="http://cfs13.tistory.com/image/10/tistory/2008/12/08/20/54/493d0af9b086d"/>

//http://websystique.com/spring/spring4-hibernate4-mysql-maven-integration-example-using-annotations/

/\*

비영속 상태 - 퍼시스턴트 객체를 처음 만들었을 때의 상태. 데이터베이스 테이블에 관련 데이터가 없으며, 연관된 Session이 없다.

영속 상태 - 현재 활성화된 Session과 연결된 퍼시스턴트 객체. 이 상태의 퍼시스턴트 객체는 고유성을 가지며, 프로퍼티 값의 변경이 Session을 통해 자동으로 데이터베이스에 반영된다.

준영속 상태 - 영속 상태의 퍼시스턴트 객체가 Session과 연결이 끊기면 준영속 상태가 된다. Hibernate의 관리를 받지는 않지만, 영속 데이터를 갖고 있다.

코드로보면..

비영속 - SomObj obj = new SomObj();

Session s = sessionFactory.openSession();

Transaction tx = s.beginTransaction()...

영속 - s.save(obj), s.remove(obj)

tx.commit(); <-영속,DB반영 session.flush(); <-영속,DB반영

트랜젝션을 커밋하면 flush가 된다, jpql쿼리( session.createQuery("from BootCodeVO")) 실행될때도 플러시가 자동 호출된다.

Criteria 같은 객체지향 쿼리를 호출할떄도 플러시가 실행된다

session.close()

준영속 - obj.getId();..

em.detach(entity) 특정 엔티티만 준영속 상태로 전환

em.clear() 영속성 컨텍스트를 완전 초기화환다

em.close() 영속성 컨텍스트를 종료한다

객체가 고유성이 없는 경우(즉, 데이터베이스에 1대 1로 매핑되는 데이터가 존재하지 않는 경우) 비영속 상태에 해당한다.

예를 들어, 위 코드처럼 퍼시스턴트 객체를 새로 생성한 경우에 비영속 상태가 된다.

비영속 상태의 객체를 Session을 통해 저장하면(save() 또는 saveOrUpdate() 등을 사용)

객체는 영속 상태가 된다. 영속 상태의 객체는 활성화된 Session과 관련을 맺는다.

보통 영속 상태가 된다 하더라도 Session의 트랜잭션이 커밋되지 않으면 데이터베이스에 실제로 저장되지는 않는다.

일단, 객체가 영속 상태가 되면 퍼시스턴트 객체의 값 변화는 자동으로 데이터베이스에서 반영된다.

(이를 automatic dirty checking)이라 한다. Hibernate Session은 트랜잭션이 끝날 때 Session과 관련 있는 영속 상태 퍼시스턴트

객체의 프로퍼티가 변경되었는 지 확인한 후, 변경된 내용을 데이터베이스와 동기화한다.

Session이 종료되면 Session과 관련된 모든 영속 상태의 퍼스시턴트 객체는 준영속 상태가 된다. 준영속 상태의 퍼시스턴트 객체는

데이터베이스와 연결되어 있지 않기 때문에 프로퍼티 값을 변경하더라도 그 내역이 데이터베이스에 반영되지는 않는다. 준영속 상태의 퍼시스턴트 객체는 나중에 다시

다른 Session과 연결될 수 있으며, Session과 연결되면 다시 영속 상태로 바뀌게 된다. 다음의 코드는 준영속 상태로 된 객체가 다시 영속 상태로 변경되는

예를 보여주고 있다.

Session session1 = sessions.openSession();

...

Book book = (Book)session1.get(Book.class, new Integer(1)); // 영속 상태

...

session1.close(); // 이 시점에서 book은 준영속 상태로 변경

...

Session session2 = sessions.openSession();

...

session2.update(book); // 이 시점에서 다시 영속 상태로 변경

...

session2.close();

save(Object obj) - 새로운 객체를 (테이블에) 저장한다.

save(Object obj, Serializable id) - 새로운 객체를 (테이블에) 저장한다. 이때 식별값으로 두번째 인자로 전달받은 객체를 사용한다.

Object get(Class type, Serializable id) - 지정한 타입의 객체를 읽어온다. 두번째 인자는 객체를 읽어올 때 사용할 식별자이다.

Object load(Class type, Serializable id) - 지정한 타입의 객체를 읽어온다. 두번째 인자는 객체를 읽어올 때 사용할 식별자이다.

load(Object obj, Serializable id) - obj의 타입과 관련된 데이터를 읽어와 obj에 저장한다. 두번째 인자는 객체를 읽어올 때 사용할 식별자이다.

update(Object obj) - 객체의 값을 변경한다.

update(Object obj, Serializable id) - 지정한 식별자에 해당하는 테이블 행을 객체가 저장한 값으로 변경한다.

saveOrUpdate(Object obj) - 새로운 객체인 경우 (테이블에) 저장하고, 이미 존재하는 객체인 경우 (테이블의) 값을 새로운 값으로 변경한다.

delete(Object obj) - 객체를 (테이블에서) 삭제한다.

객체를 읽어오는 메소드가 get과 load 두가지가 있는데, 이 두 메소드는 다음과 같은 차이가 있다.

get : 읽어올 객체가 없는 경우 null을 리턴하며, 프록시 객체를 리턴하지 않는다.

load : 읽어올 객체가 없는 경우 예외를 발생하며, 프록시 객체를 리턴할 수 있다.

객체를 변경하는 방법은 크게 두가지가 있는데 하나는 Session의 트랜잭션 범위에서 영속 상태의 객체의 프로퍼티를 변경하는 것이며, 두번째 방법은 준영속 상태의 개체를 변경한 뒤 Session과 연관시키는 것이다. 다음은 두 방식의 차이점을 코드로 보여준다.

Session session1 = sessions.openSession();

Transaction tx1 = session1.beginTransaction();

Item item = (Item)session1.get(Item.class, new Integer(6));

item.setTitle(newTitle); // 영속 상태의 객체 프로퍼티 값 변경

tx1.commit(); // update() 메소드를 호출하지 않아도 영속 상태 객체의 변경 내용이 적용

session1.close();

item.setLowPrice(1000); // 준영속 상태의 객체 프로퍼티 값을 변경

Session session2 = sessions.openSession();

Transaction tx2 = session1.beginTransaction();

session2.update(item); // 준영속 상태를 영속 상태로 변경

tx2.commit();

session2.close();

객체를 삭제할 때에는 다음과 같이 get()이나 load()로 읽어온 뒤 delete() 메소드를 호출하면 된다.

Session session = sessions.openSession();

Transaction tx = session1.beginTransaction();

Item item = (Item)session.get(Item.class, new Integer(6));

session.delete(item);

tx.commit();

session.close();

\*/

|  |
| --- |
| **public** List<BootMenuVO> getMenu(){  Session session = openSession();  Criteria crit = session.createCriteria(BootMenuVO.**class**).add(Restrictions.*eq*("type", "M")).addOrder(Order.*asc*("right\_id"));  List<BootMenuVO> list = crit.list();  closeSession(session);  **return** list;  }  /\*http://www.mkyong.com/hibernate/hibernate-parameter-binding-examples/  String hql = "from Stock s where s.stockCode = :stockCode";  List result = session.createQuery(hql)  .setParameter("stockCode", "7277")  .list();  \*/  //http://www.journaldev.com/2954/hibernate-query-language-hql-example-tutorial  **public** List<BootCodeVO> selectCode(){  // Criteria crit = sessionFactory.getCurrentSession().createCriteria(BootCodeVO.class);  Session session = openSession();  Criteria crit = session.createCriteria(BootCodeVO.**class**);  List list = crit.list();  closeSession(session);  **return** list;  }  **public** List<BootCodeVO> selectCodeWhere(){  Session s = **this**.sessionFactory.openSession();  Criteria crit = s.createCriteria(BootCodeVO.**class**).add(Restrictions.*eq*("cd", "CODE100"));  // BootCodeVO = (BootCodeVO)crit.uniqueResult();  s.close();  **return** crit.list();  };  **public** List<BootCodeVO> selectCodeQuery1(){    /\*  \* Query query = session.createQuery("from Resource where emplteam = :team");  query.setParameter("team", "Team1");  List list = query.list();  //Get Employee with id  query = session.createQuery("from Employee where id= :id");  query.setLong("id", 3);  Employee emp = (Employee) query.uniqueResult();  System.out.println("Employee Name="+emp.getName()+", City="+emp.getAddress().getCity());    \*/  Session session = **this**.sessionFactory.openSession();  List<BootCodeVO> personList = session.createQuery("from BootCodeVO").list();  session.close();  **return** personList;  };  **public** List<BootCodeVO> selectCodeQuery2(){  /\*  Query q = sessionFactory.getCurrentSession().createQuery("from Contract c where :fullName = concat(c.firstname, ' ', c.lastname)");  q.setString("fullName", fullName);  return q.list();}  \*/  Session session = **this**.sessionFactory.openSession();  session.getNamedQuery("query");  List<BootCodeVO> personList = session.createSQLQuery("select \* from BootCodeVO").list();  session.close();  **return** personList;  };  **public** Long selectCodeQueryCnt(){  /\*  session session = getSessionFactory().beginSession();  Transaction transaction = null;    StringBuilder sql = new StringBuilder();    sql.append("SELECT DISTINCT B.COLUMN\_NAME ");  sql.append("FROM ALL\_CONSTRAINTS A, ALL\_CONS\_COLUMNS B, ARCTBL C ");  sql.append("WHERE A.TABLE\_NAME = :TABLE\_NAME AND A.CONSTRAINT\_TYPE = 'P' ");  sql.append("AND A.CONSTRAINT\_NAME = B.CONSTRAINT\_NAME ");  sql.append("AND C.TABLENAME=A.TABLE\_NAME AND A.TABLE\_NAME=B.TABLE\_NAME ");    try{    \*\*listOfValues = session.createSQLQuery(sql.toString()).setParameter("TABLE\_NAME", table).list();\*\*  //Change this to code below  String returnObjByHibernate = session.createSQLQuery(//Hibernate method)  transaction.commit;  }  catch (Exception e){  //Logic here  \*/  Session session = **this**.sessionFactory.openSession();  Long count = (Long)session.createQuery("select COUNT(\*) from BootCodeVO").uniqueResult();  session.close();  **return** count;  /\*  String hql = "select count(name) from Product";  Query query = session.createQuery(hql);  List listResult = query.list();  Number number = (Number) listResult.get(0);  System.out.println(number.intValue());  \*/  };  **public** Integer selectCodeCnt(){  Session s = **this**.sessionFactory.openSession();  Criteria crit = s.createCriteria(BootCodeVO.**class**);  Integer count = crit.list().size();  s.close();  **return** count;  }  **public** List<AreaVO> getArea() {  Session s = openSession();  Criteria crit = s.createCriteria(AreaVO.**class**);  crit.addOrder(Order.*asc*("order\_id1")).addOrder(Order.*asc*("order\_id2"));  List<AreaVO> list = crit.list();  closeSession(s);  **return** list;  }  **public** List<MsgClassVVO> getMsgClass() {  Session s = openSession();  Criteria crit = s.createCriteria(MsgClassVVO.**class**);  crit.addOrder(Order.*asc*("class1\_order")).addOrder(Order.*asc*("class2\_order"));  List<MsgClassVVO> list = crit.list();  closeSession(s);  **return** list;  }; |

|  |
| --- |
| @RequestMapping("/test")  **public** List<MsgClassVVO> test(HttpServletRequest request,HttpServletResponse reponse, ModelMap model) **throws** Exception {    /\*  Type[] tipos = {IntegerType.INSTANCE, IntegerType.INSTANCE};  Integer[] values = {1, 2};  criteria.add(Restrictions.sqlRestriction("SELECT ({alias}.id \* {alias}.code \* "+ ? + " \* " + ? + ")  as number from city HAVING number < 10", values, tipos ));  \*/  // Projections.sqlProjection(sql, columnAliases, types);    // Projection p = Projections.sqlProjection("substr({alias}.class2,1,2) as CLASS2", new String [] {"CLASS2"}, new Type [] {StringType.INSTANCE} );  // Projection pg = Projections.sqlGroupProjection("sum({alias}.class2,1,2) as rowCount","this\_.discriminatorColumn having rowCount>0",new String[]{"rowCount"},Helper.HIBERNATE\_INTEGER\_INSTANCE)  // Projection p = Projections.sqlProjection("substr({alias}.class3,1,2) as class3\_short", new String[]{"value"}, new Type[]{new LongType()}))  //http://stackoverflow.com/questions/29482880/how-to-pass-value-to-sqlrestriction-of-hibernate  //https://docs.jboss.org/hibernate/orm/3.3/reference/ko-KR/html/querycriteria.html      //http://stackoverflow.com/questions/7773072/using-database-functions-to-transform-columns-in-hibernate-criteria  ProjectionList plist = Projections.*projectionList*()  .add(Projections.*sqlProjection*("substr({alias}.class2,1,2) as class2", **new** String [] {"class2"}, **new** Type [] {StringType.***INSTANCE***} ))  .add(Projections.*groupProperty*("{alias}.class2"));    Criterion where = Restrictions.*sqlRestriction*("substr({alias}.class1, 1, 2)=?", "자연", StringType.***INSTANCE***);  Session s = sessionFactory.openSession();  Criteria crit = s.createCriteria(MsgClassVVO.**class**).add(where);  //crit.createAlias("products","p");  //crit.setProjection(Projections.distinct(Projections.property("class1")));  //crit.setProjection(Projections.sqlGroupProjection("substr({alias}.class2,1,2) as class2", "substr({alias}.class2,1,2) ", new String[] { "class2" }, new Type[] { StandardBasicTypes.STRING }));  //crit.createAlias("class1",)  crit.setProjection(plist);  List<MsgClassVVO> list = crit.list();  s.close();  **return** list;  } |

Function

http://docs.jboss.org/hibernate/orm/3.5/reference/en/html\_single/

|  |
| --- |
| Hibernate Criteria Query examples  Hibernate supports a feature that helps us to build queries dynamically. The org.hibernate.Criteria is the interface used for this purpose. Criteria is useful when we are supposed to apply a multiple complicated search criteria on the queries.   For a Criteria query, a Criterion Object should be constructed (i.e) a call to createCriteria() passing the class of the objects we want the query to return.  Let us go the package **com.demo.CriteriaExamples** and execute the class **CriteriaExampleExecutor.java  Criteria Example:** Following is an example that will all the instances of the class Employee. **Criteria criteria = session.createCriteria(Employee.class);** List critList = criteria.list(); for(Object objList: critList) System.out.println("Employee Name is "+ ((Employee)objList).getEmpName()); **Limiting the number of rows returned:** This is particularly useful in pagination. **Criteria criteria = session.createCriteria(Employee.class).setMaxResults(2);** critList = criteria.list(); System.out.println("Number of rows returned "+ critList.size()); for(Object objList: critList) System.out.println("Employee Name is "+ ((Employee)objList).getEmpName()); **Narrowing the result set:** Restrictions class provides factory methods for built-in Criterion types.In our example we have used some of the factory methods like Restrictions.like(),Restrictions.between(), Restrictions.isNotNull().  **Criteria criteria = session.createCriteria(Employee.class).add(Restrictions.like("empName", "Ana%"));** critList = criteria.list(); for(Object objList: critList) System.out.println("Employee Name is "+ ((Employee)objList).getEmpName()); **Ordering the Result Set:**  The class Order provides the factoy methods asc() and desc() to order objects based on any particular property. **Criteria criteria = session.createCriteria(Employee.class).addOrder(Order.asc("empName"))** **.addOrder(Order.desc("salary"));** critList = criteria.list(); for(Object objList: critList) System.out.println("Employee Name is "+ ((Employee)objList).getEmpName() + " Employee Salary is " + ((Employee)objList).getSalary()); **Logical and Comparison Operators available in Restrictions Class:** All regular SQL comparison operators are available in the Restrictions class.The methods Restrictions.or() and Restrictions.and() will serve the Logical operators. **Criteria criteria = session.createCriteria(Employee.class).add(Restrictions.or(** **Restrictions.gt("salary", new Integer(23000)), Restrictions.ge("age", new Integer(30))));** critList = criteria.list(); for(Object objList: critList) System.out.println("Employee Name salary is more than 23000 or whose age is greater than or equal to 30 is "+ ((Employee)objList).getEmpName());  **Aggregate Functions():** Projections class serves the factory methods which can be used as the aggregate functions as follows. **critList = session.createCriteria(Employee.class).setProjection(Projections.max("salary")).list();** System.out.println("Maximum salary is "+ critList.get(0)); **critList = session.createCriteria(Employee.class).setProjection(Projections.min("salary")).list();** System.out.println("Minimum salary is "+ critList.get(0)); **critList = session.createCriteria(Employee.class).setProjection(Projections.rowCount()).list();** System.out.println("Total employees "+ critList.get(0)); **critList = session.createCriteria(Employee.class).setProjection(Projections.avg("age")).list();** System.out.println("Average employee age "+ critList.get(0)); **critList = session.createCriteria(Employee.class).setProjection(Projections.sum("salary")).list();** System.out.println("Sum of all the salaries "+ critList.get(0)); Also, Projections class will help us to select exactly the objects or the properties of objects we need in the query result as follows. **session.createCriteria(Employee.class).setProjection(Projections.projectionList().add(Projections.id()) .add(Projections.property("empName")) .add(Projections.property("age")) );** |

|  |
| --- |
| Criteria and Aggregation We saw [aggregate functions with HQL](http://learningviacode.blogspot.com/2012/12/hql-and-aggregrate-functions.html). Criteria too has methods that provide for using aggregate functions. Consider the below code that simply fetches the count of all rows.  **Count**  **public** **static** void getCount() {  **final** Session session = sessionFactory.openSession();  Criteria criteria = session.createCriteria(Entity.class);  criteria.setProjection(Projections.rowCount());  **Integer** count = (**Integer**) criteria.list().get(0);    //The count value is returned as Integer  **System**.out.println("No of entities is " + count);  }  The output of the method is :  **select**  **count(\*) as y0\_**  **from**  **ENTITY this\_**  No of entities is 4  The rowCount() method returns an instance of **RowCountProjection**which appends the count fragment to the generated SQL. Its the first time, I have seen the \* used in an SQL query by Hibernate.  This is from the **RowCountProjection**class:  **public** **class** RowCountProjection **extends** SimpleProjection {  //...  **public** **String** toSqlString(Criteria criteria, **int** position, CriteriaQuery criteriaQuery)  **throws** HibernateException {  **return** **new** StringBuffer()  .append("count(\*) as y")  .append(position)  .append('\_')  .toString();  }  //...  }  I decided to try out the queries I used in the HQL example earlier:  Query q = session.createQuery("select count(e) from Entity e where e.master.id = ? ");  The same using Criteria would be:  **public** **static** void getCount2() {  **final** Session session = sessionFactory.openSession();  Criteria criteria = session.createCriteria(Entity.class);  criteria.add(Restrictions.eq("master.id", 2L));  criteria.setProjection(Projections.count("name"));  **Integer** count = (**Integer**) criteria.list().get(0);  **System**.out.println("No of entities is " + count);  }  The result is :  select  count(this\_.NAME) as y0\_  from  ENTITY this\_  where  this\_.MASTER\_ID=?  No of entities is 2  In case we need to ensure that the records are distinct:  criteria.setProjection(Projections.countDistinct("name"));  **Max and Min**  For max and min I had executed a combined query earlier:  "select max(e.id), min(e.id) from Entity e where e.master.id = ? "  **public** **static** void getMaxMin() {  //("select max(e.id), min(e.id) from Entity e where e.master.id = ? "    **final** Session session = sessionFactory.openSession();  Criteria criteria = session.createCriteria(Entity.class);  criteria.add(Restrictions.eq("master.id", 2L));  criteria.setProjection(  Projections.projectionList()  .add(Projections.max("id"))  .add(Projections.min("id")  )  );  **Object**[] result = (**Object**[]) criteria.list().get(0);  **System**.out.println("Max id is " + result[0] + " and min id is " + result[1]);  }  **public** **static** void getMaxMin() {  //("select max(e.id), min(e.id) from Entity e where e.master.id = ? "    **final** Session session = sessionFactory.openSession();  Criteria criteria = session.createCriteria(Entity.class);  criteria.add(Restrictions.eq("master.id", 2L));  criteria.setProjection(  Projections.projectionList()  .add(Projections.max("id"))  .add(Projections.min("id")  )  );  **Object**[] result = (**Object**[]) criteria.list().get(0);  **System**.out.println("Max id is " + result[0] + " and min id is " + result[1]);  }  The max() and min() method both return instances of **AggregateProjection**. The query generated is :  **select**  **max(this\_.ID) as y0\_,**  **min(this\_.ID) as y1\_**  **from**  **ENTITY this\_**  **where**  **this\_.MASTER\_ID=?**  Max id is 5 and min id is 4  **Avg and Sum**  The code for the aggregate methods is :  **public** **static** void getAvgSum() {  //select avg(c.id), sum(c.id) from Child c where c.id is not null    **final** Session session = sessionFactory.openSession();  Criteria criteria = session.createCriteria(Child.class);  criteria.add(Restrictions.isNotNull("id"));  criteria.setProjection(  Projections.projectionList()  .add(Projections.avg("id"))  .add(Projections.sum("id")  )  );  **Object**[] result = (**Object**[]) criteria.list().get(0);  **System**.out.println("Avg id is " + result[0] + " and sum of id is " + result[1]);  }  The generated SQL is :  **select**  **avg(this\_.ID) as y0\_,**  **sum(this\_.ID) as y1\_**  **from**  **CHILD\_ENTITY this\_**  **where**  **this\_.ID is not null**  Avg id is 3.5 and sum of id is 21 |

insert하고싶은데 update될때..

|  |
| --- |
| **public** Code addMsg(MsgWaitVO msgWait) {  hibernate.flushClearBeginTransaction();  msgWait.setCreate\_id(securityMng.getSecurityUser().getOperator\_id());  hibernate.save(msgWait);  **for** (MsgCompany c :msgWait.getCompanys()) {  c.setWait\_id(msgWait.getWait\_id());  }  **for** (MsgAreaVO c :msgWait.getAreas()) {  c.setWait\_id(msgWait.getWait\_id());  }  hibernate.flushClearCommit();  **return** Code.***SUCCESS***;  }  @Entity  @Table(name = "TB\_MSG\_WAIT")  @SequenceGenerator(  name = "CBIS\_SEQ\_GENERATOR",  sequenceName = "CBIS\_SEQ",  initialValue = 1, allocationSize = 1  )  **public** @Data **class** MsgWaitVO **extends** DefaultVO {  **private** **static** **final** **long** ***serialVersionUID*** = 3957786793896721440L;  @Expose @Column(name="APPROVE\_DATE") String approve\_date ;//승인날짜 CHAR  @Expose @Column(name="APPROVE\_ID") Integer approve\_id ;//승인ID NUMBER  @Expose @Column(name="APPROVE\_TIME") String approve\_time ;//승인시간 CHAR  @Expose @Column(name="CALLBACK\_DATA") String callback\_data ;//콜백내용 VARCHAR2  @Expose @Column(name="CALLBACK\_TYPE") String callback\_type ;//콜백종류 CHAR  @Expose @Column(name="CH\_ID") String ch\_id ;// VARCHAR2  @Expose @Column(name="CLASS\_ID") Integer class\_id ;//정보분류ID NUMBER  @Expose @Column(name="CONTENTS\_ALL") String contents\_all ;//전체 컨텐츠내용 VARCHAR2  @Expose @Column(name="CONTENTS\_KTF") String contents\_ktf ;//KTF용 컨텐츠내용 VARCHAR2  @Expose @Column(name="CONTENTS\_KT\_LTE") String contents\_kt\_lte ;//KT LTE VARCHAR2  @Expose @Column(name="CONTENTS\_LGT") String contents\_lgt ;//LGT용 컨텐츠내용 VARCHAR2  @Expose @Column(name="CONTENTS\_LGU\_LTE") String contents\_lgu\_lte ;//LG U+ LTE VARCHAR2  @Expose @Column(name="CONTENTS\_SKT") String contents\_skt ;//SKT용 컨텐츠내용 VARCHAR2  @Expose @Column(name="CONTENTS\_SKT\_LTE") String contents\_skt\_lte ;//SKT LTE VARCHAR2  @Expose @Column(name="CREATE\_DATE") String create\_date ;//등록날짜 CHAR  @Expose @Column(name="CREATE\_ID") Integer create\_id ;//등록자ID NUMBER  @Expose @Column(name="CREATE\_TIME") String create\_time ;//등록시간 CHAR  @Expose @Column(name="GRADE\_ID") Integer grade\_id ;//정보등급ID NUMBER  @Expose @Column(name="STATUS") String status ;//상태 VARCHAR2  @Expose @Column(name="TEST\_CH") String test\_ch ;//테스트송출 CHAR  @Expose @Column(name="WAIT\_DATE") String wait\_date ;//예약날짜 CHAR  @GeneratedValue(strategy = GenerationType.***SEQUENCE***, generator = "CBIS\_SEQ\_GENERATOR")  @Expose @Id @Column(name="WAIT\_ID") Integer wait\_id ;//예약ID NUMBER  @Expose @Column(name="WAIT\_TIME") String wait\_time ;//예약시간 CHAR    @Expose  @OneToMany(cascade=CascadeType.***ALL***)  @JoinColumn(name="WAIT\_ID")  List<MsgCompany> companys = **new** ArrayList<MsgCompany>();    @Expose  @OneToMany(cascade=CascadeType.***ALL***)  @JoinColumn(name="WAIT\_ID")  List<MsgAreaVO> areas = **new** ArrayList<MsgAreaVO>(); |

수동으로 FetchMode변경 Join인경우

|  |
| --- |
| @Entity  @Table(name = "TB\_OPERATOR")  **public** @Data **class** LoginUserVO **extends** DefaultVO **implements** UserDetails{  **private** **static** **final** **long** ***serialVersionUID*** = -162352325348994087L;  …  @OneToOne(fetch=FetchType.***EAGER***)  @JoinColumn(name="CARD\_NO",referencedColumnName="CARD\_NO", insertable=**false**, updatable=**false**)  LoginCheckCardJoinVO checkCard; |
| **public** @Data **class** LoginCheckCardJoinVO **extends** DefaultVO {  …  @OneToMany(fetch=FetchType.***LAZY***)  @JoinColumn(name="CARD\_ID",referencedColumnName="CARD\_ID")  List<LoginCheckCodeVO> codes = **new** ArrayList<LoginCheckCodeVO>(); |
| @Entity  @Table(name = "TB\_CHECK\_CODE")  **public** @Data **class** LoginCheckCodeVO **extends** DefaultVO {  **private** **static** **final** **long** ***serialVersionUID*** = -9220842008341369522L;  @Expose @Id @Column(name="CODE") Integer code ;//???? NUMBER  @Expose @Id @Column(name="CODE\_ID") Integer code\_id ;//??ID NUMBER  @Expose @Id @Column(name="CARD\_ID") Integer card\_id ;//??ID NUMBER  } |
| Criteria crit = hibernate.createCriteria(LoginUserVO.**class**).add(Restrictions.*eq*("login\_id", id));  crit.setFetchMode("checkCard.codes", FetchMode.***JOIN***);  LoginUserVO l = (LoginUserVO) crit.uniqueResult();  **return** l; |
| 복합적으로 날라가긴하지만 그래도 join처리잘됨  select this\_.OPERATOR\_ID as OPERATOR\_ID1\_16\_2\_, this\_.AREA as AREA2\_16\_2\_,  this\_.CAN\_LOGIN as CAN\_LOGIN3\_16\_2\_, this\_.CARD\_NO as CARD\_NO4\_16\_2\_, this\_.CREATE\_DATE as  CREATE\_DATE5\_16\_2\_, this\_.CREATE\_ID as CREATE\_ID6\_16\_2\_, this\_.CREATE\_TIME as CREATE\_TIME7\_16\_2\_,  this\_.DELETED as DELETED8\_16\_2\_, this\_.EMAIL as EMAIL9\_16\_2\_, this\_.INDV\_KEY as INDV\_KEY19\_16\_2\_,  this\_.LAST\_LOGIN\_DATE as LAST\_LOGIN\_DATE20\_16\_2\_, this\_.LAST\_LOGIN\_IP as LAST\_LOGIN\_IP21\_16\_2\_,  this\_.LAST\_LOGIN\_SESSION as LAST\_LOGIN\_SESSIO22\_16\_2\_, this\_.LAST\_PWD\_UPDATE as LAST\_PWD\_UPDATE23\_16\_2\_,  this\_.LOCKED as LOCKED10\_16\_2\_, this\_.LOGIN\_FAIL as LOGIN\_FAIL24\_16\_2\_, this\_.LOGIN\_ID as LOGIN\_ID11\_16\_2\_,  this\_.LOGIN\_PW as LOGIN\_PW12\_16\_2\_, this\_.NAME as NAME13\_16\_2\_, this\_.OP\_LEVEL as OP\_LEVEL14\_16\_2\_,  this\_.PART as PART15\_16\_2\_, this\_.PHONE1 as PHONE16\_16\_2\_, this\_.PHONE2 as PHONE17\_16\_2\_, this\_.POSITION  as POSITION18\_16\_2\_, logincheck2\_.CARD\_NO as CARD\_NO1\_5\_0\_, logincheck2\_.CARD\_ID as CARD\_ID2\_5\_0\_,  codes3\_.CARD\_ID as CARD\_ID3\_5\_4\_, codes3\_.CODE\_ID as CODE\_ID1\_6\_4\_, codes3\_.CODE as CODE2\_6\_4\_,  codes3\_.CARD\_ID as CARD\_ID3\_6\_4\_, codes3\_.CODE\_ID as CODE\_ID1\_6\_1\_, codes3\_.CODE as CODE2\_6\_1\_,  codes3\_.CARD\_ID as CARD\_ID3\_6\_1\_ from TB\_OPERATOR this\_ left outer join TB\_CHECK\_CARD logincheck2\_  on this\_.CARD\_NO=logincheck2\_.CARD\_NO left outer join TB\_CHECK\_CODE codes3\_ on logincheck2\_.CARD\_ID=codes3\_.CARD\_ID  where this\_.LOGIN\_ID='ag0001' |

Entity 전처리 후처리

|  |
| --- |
| // @PrePersist  // void preInsert() {  // if ( getCan\_login() == null ) { setCan\_login( "Y" ); }  // }  // @PrePersist  // void preInsert() {  // if(null==create\_date){  // create\_date = DateUtil.getDate("yyyyMMdd");  // }  // if(null==create\_time){  // create\_time = DateUtil.getDate("HHmmss");  // }  // if(null==can\_login){  // can\_login="Y";  // }  // if(null==deleted){  // deleted="N";  // }  // if(null==locked){  // locked="N";  // }  // } |

NULL값 insert 및 update안하도록 하는것

|  |
| --- |
| //@DynamicInsert(value=true)  //@DynamicUpdate(value=true)  //@SelectBeforeUpdate  //@org.hibernate.annotations.Entity(  // dynamicInsert = true, dynamicUpdate = true  //) |

And (or) 처리

|  |
| --- |
| <http://stackoverflow.com/questions/8126589/hibernate-criteria-restrictions-and-or-combination>  (((A='X') and (B in('X',Y))) or ((A='Y') and (B='Z')))  Criteria criteria = getSession().createCriteria(clazz);  Criterion rest1= Restrictions.and(Restrictions.eq(A, "X"),  Restrictions.in("B", Arrays.asList("X",Y)));  Criterion rest2= Restrictions.and(Restrictions.eq(A, "Y"),  Restrictions.eq(B, "Z"));  criteria.add(Restrictions.or(rest1, rest2));  <http://stackoverflow.com/questions/57484/how-do-you-or-criteria-together-when-using-a-criteria-query-with-hibernate>  /Expression : (c1 AND c2) OR (c3)  Criteria criteria = session.createCriteria(Employee.class);  Criterion c1 = Restrictions.like("name", "%e%");  Criterion c2 = Restrictions.ge("salary", 10000.00);  Criterion c3 = Restrictions.like("name", "%YYY%");  Criterion c4 = Restrictions.or(Restrictions.and(c1, c2), c3);  criteria.add(c4);  //Same thing can be done for (c1 OR c2) AND c3, or any complex expression.  Criteria query = getCriteria("ENTITY\_NAME");  query.add(Restrictions.ne("column Name", current \_value));  Disjunction disjunction = Restrictions.disjunction();  if (param\_1 != null)  disjunction.add(Restrictions.or(Restrictions.eq("column Name", param1)));  if (param\_2 != null)  disjunction.add(Restrictions.or(Restrictions.eq("column Name", param\_2)));  if (param\_3 != null)  disjunction.add(Restrictions.or(Restrictions.eq("column Name", param\_3)));  if (param\_4 != null && param\_5 != null)  disjunction.add(Restrictions.or(Restrictions.and(Restrictions.eq("column Name", param\_4 ), Restrictions.eq("column Name", param\_5 ))));  if (disjunction.conditions() != null && disjunction.conditions().iterator().hasNext())  query.add(Restrictions.and(disjunction));  return query.list(); |

## [Session으로 작업하기](http://jjaeko.tistory.com/62) http://jjaeko.tistory.com/62

[Hibernate](http://jjaeko.tistory.com/category/Hibernate) 2008.03.02 02:49

참고 : 래퍼런스 10장

***객체 영속화 하기(저장 하기)***  
새로 생성된 객체는 Session.save(), Session.persist(), Session.saveOrUpdate() 메서드를 통해 영속화 할 수 있습니다. 연관된 객체가 있을 경우 cascade 설정에 따라 같이 영속화 될 수 있습니다.

Member member = new Member();  
member.setName("jjaeko");  
session.save(member);

save()는 식별 값을 리턴하며 persist()는 식별값을 리턴하지 않는 차이점이 있습니다.  
saveOrUpdate()는 자동적인 상태 검출 에서 다루겠습니다.  
  
  
***객체 로드하기***  
데이터베이스에 저장된 객체는 Session.get(), Session.load() 메서드를 이용해 로드할 수 있습니다.

Member member = (Member)session.get(Member.class, 1);  
Member member2 = (Member)session.load(Member.class, 2);

Session.get() 메서드는 즉시 객체를 가져오는 반면 Session.load() 메서드는 인자로 전달된 식별 값을 가지는 프록시 객체를 리턴하는 차이점이 있습니다. 리턴된 프록시 객체는 사용하는 시점에서 비로소 저장된 값을 가져 옵니다. 만약 데이터베이스에 식별 값에 해당하는 객체가 없다면 ObjectNotFoundException이 던져 집니다. 따라서 객체가 존재하는지 확신하지 않는다면 Session.get() 메서드를 사용해야 합니다. Session.get() 메서드는 객체가 존재하지 않을경우 null을 리턴 합니다.  
  
  
***영속 객체 변경하기***  
영속 객체를 변경하고 이를 반영하기 위해 별도의 메서드를 호출할 필요가 없습니다. 더티체킹 기능이 있기 때문인데 객체의 변경을 알아채고 자동으로 동기화 하는 기능입니다.

Member member = (Member)session.get(Member.class, 1);  
member.setName("째코");

동기화 시점은 컴밋 되거나 flush() 될 때 입니다. 만약 컴밋이나 flush()가 되지 않는다면 동기화도 되지 않습니다. flush()에 대해서는 즉시 동기화 하기 에서 다루겠습니다.  
  
  
***준영속 객체 변경하기***  
Session이 종료되면 영속 객체는 준영속 객체로 변경 됩니다. 준영속 객체는 Session.update(), Session.merge(), Session.replicate() 메서드를 통해 재첨부 될 수 있습니다.  
  
Session.update() 메서드를 이용한 예  
Session이 종료된 후 준영속 객체를 새로운 Session의 update() 메서드를 통해 영속 상태로 바꿀수 있습니다. update() 메서드를 호출하고 컴밋이나 flush()가 되지 않는다면 Update 쿼리가 수행되지 않습니다. 그리고 객체의 변경 유무를 떠나 Update 쿼리가 실행되는데 이를 방지하려면 select-before-update 속성 값을 true로 설정하면 됩니다.

Member member = (Member)session.get(Member.class, 1);  
session.close();                     // Session 종료  
member.setName("김재호"); // Session 종료후 값 변경  
session2.update(member);   // 새로운 Session

Session.merge() 메서드를 이용한 예  
update() 메서드를 이용할 때 한가지 문제점이 있습니다. 새로운 Session내에 update()하려는 준영속 객체와 동일한 식별 값을 갖는 영속 객체가 존재한다면 NonUniqueObjectException이 던져 집니다.

Member member = (Member)session.get(Member.class, 1);  
session.close();                       
member.setName("김재호");  
Member member2 = (Member)session.get(Member.class, 1);  
session2.update(member);   // 예외 발생

이 경우 merge() 메서드를 사용 합니다. merge() 메서드는 전달된 준영속 객체의 값을 동일한 식별 값을 갖는 객체에 적용한뒤 리턴 합니다. 그리고 전달된 준영속 객체는 계속 준영속 상태로 남아 있습니다.

Member member = (Member)session.get(Member.class, 1);  
session.close();                       
member.setName("김재호");  
Member member2 = (Member)session.get(Member.class, 1);  
Member member3 = session2.merge(member);   // member2와 member3은 같은 객체

위 예제에서는 merge() 메서드를 호출하기 전 이미 동일한 식별 값을 갖는 영속 객체가 존재 합니다. 만약 동일한 식별 값을 갖는 영속 객체가 존재하지 않을 경우 알아서 로드후 변경을 적용 합니다.  
 ***자동적인 상태 검출***  
대상 객체가 새로운 객체인지 아니면 준영속 객체인지에 따라 save() 메서드나 update() 메서드를 구분하여 호출 해야 합니다. 하지만 Session.saveOrUpdate() 메서드는 객체의 상태를 파악하고 알아서 작업을 수행 합니다. Session.saveOrUpdate() 는 대략 이런 시나리오에 의해 사용됩니다.

1. 첫번째 Session에서 객체 로드  
2. 로드된 객체가 UI로 전달  
3. UI에서 객체 변경  
4. 두번째 Session에서 update()

saveOrUpdate() 메서드의 행동은 이렇습니다.

* 대상 객체가 현재 Session 내에 영속화 되어 있을 경우 아무 작업도 하지 않습니다.
* Session 내에 전달된 준영속 객체와 동일한 식별 값을 갖는 객체가 존재한다면 예외가 던져 집니다.
* 대상 객체가 식별자 프로퍼티를 갖고 있지 않다면 save() 합니다.
* 대상 객체가 식별 값을 갖고 있지 않다면 save() 합니다.
* version 값이 없다면 save() 합니다.
* 그 밖의 경우 update() 합니다.

***영속 객체 삭제하기***  
Session.delete() 메서드를 이용해 데이터베이스에 저장된 객체를 삭제할 수 있습니다. 객체를 삭제하더라도 어플리케이션 에서는 여전히 준영속 상태로 객체를 참조할 수 있습니다.  
  
  
***서로 다른 데이터베이스 사이에 객체 복사하기***  
Session.replicate() 메서드를 이용해 서로 다른 데이터베이스 사이에 객체를 복사할 수 있습니다.  
(아래 예제 코드에서 트랜잭션 코드는 생략 했습니다.)

Member member = session1.get(Member.class, 1);  
session1.close();  
session2.replicate(member, ReplicationMode.IGNORE);

replicate() 메서드의 두번째 인자는 대상 데이터베이스에 동일한 식별 값을 갖는 객체가 이미 있을 경우 처리 방법을 결정 합니다. 다음 4가지 옵션이 있습니다.

ReplicationMode.IGNORE : 실행을 무시합니다.  
ReplicationMode.OVERWRITE : 덮어 씌웁니다.  
ReplicationMode.EXCEPTION : 예외를 발생 시킵니다.  
ReplicationMode.LASTED\_VERSION : 마지막 버전 값을 갖는 객체를 선택 합니다.

그리고 세가지 주의 사항이 있습니다.

첫번째로 복사하려는 객체는 반드시 준영속 객체여야 합니다.  
두번째로 복사하려는 객체가 연관 객체가 있을 경우 cascade속성 값에 replicate가 지정되어 있어야 합니다.  
세번째로 복사하려는 객체의 연관 객체가 프록시 객체라면 반드시 연관 객체들을 가져온 상태여야 합니다. 왜냐면 자신이 속해 있던 Session이 이미 종료되었기 때문에 연관 객체를 불러오지 못하고 대상 데이터베이스에 복사하지 못하기 때문 입니다.

***즉시 동기화 하기***  
Session내에 존재하는 영속 객체는 기본적으로 다음 세가지 시점에서 데이터베이스와 동기화 됩니다.

org.hibernate.Transaction.commit() 호출  
Session.flush() 호출  
몇몇 쿼리가 실행되기 전

마지막 특정 쿼리가 실행되기 전의 경우에 대해 알아보겠습니다.

Member member = new Member();  
session.save(member);  
List memberList = session.createCriteria(Member.class).list();

새로 생성된 member 객체를 저장했지만 데이터베이스와 동기화되지 않은 상태입니다. 이 때 list() 메서드로 저장된 객체들을 조회할 경우 앞서 저장했던 member 객체는 어떻게 될까요? 아직 저장되지 않았기 때문에 누락될 것이지만 Hibernate는 이 경우 올바른 데이터 보장을 위해 member 객체를 동기화 합니다.  
또 다른 예외 사항으로 native 식별자 생성기를 사용하는 객체를 저장할 경우 식별 값을 알기 위해 즉시 동기화 합니다. FlushMode에 대해서는 테스트 결과 뜻대로 되지 않아서 다루지 않겠습니다.

## [openSession()과 currentSession()의 close()](http://jjaeko.tistory.com/63)

[Hibernate](http://jjaeko.tistory.com/category/Hibernate) 2008.03.04 01:24

openSession()은 항상 새로운 트랜잭션의 Session을 돌려주며 currentSession은 현재 트랜잭션의 Session을 돌려준다는 사실은 누구나 이미 알고 있는 사실 입니다.  
  
그런데...  
  
테스트중 이미 세션이 닫혀 있다는 예외가 났습니다.  
평소와 다름 없이 테스트 했을 뿐인데 이런 예외가 발생하니 좀 당황 스러웠습니다.  
문제의 코드는 바로 이것...

HibernateUtil.commit();  
HibernateUtil.closeSession();

두 메서드 모두 currentSession으로 얻어진 Session으로 작업을 수행 합니다.  
  
검색을 해보니 Session.close() 메서드는 openSession() 으로 얻어졌을 때와 currentSession() 으로 얻어졌을 때 차이가 있다는 내용을 발견 했습니다. openSession() 으로 얻어진 Session은 commit을 하더라도 close()가 되지 않지만 currentSession() 으로 얻어진 Session은 commit될 때 자동으로 close()가 된다는 내용 이었습니다.  
  
저만 몰랐나 봅니다 -0-

Re: [KSUG] Re: Hibernate getCurrentSession() 시 에러 발생



더 자세한 설정 정보가 필요할 듯 싶은데요.

혹시 웹프로젝트라면 OpenSessionInViewFilter 와 같은 Session 을 시작하고 종료하는 필터를 등록하셨나요?

JPA 로 구성하였다면, OpenEntityManagerInViewFilter 를 사용하시구요.

2014년 6월 13일 오전 12:10, Sungju Bong <[bongs...@gmail.com](javascript:)>님이 작성:

- 받은메일 표시 -

웹에서 이 토론을 보려면 [https://groups.google.com/d/msgid/ksug/CAAq3W%2BxquE3eC\_Ve\_teQUG-2jKqLeuWdKPCDZ%2BG%2BbmAtpbipkg%40mail.gmail.com](https://groups.google.com/d/msgid/ksug/CAAq3W%2BxquE3eC_Ve_teQUG-2jKqLeuWdKPCDZ%2BG%2BbmAtpbipkg%40mail.gmail.com?utm_medium=email&utm_source=footer)을(를) 방문하세요.

더 많은 옵션을 보려면 <https://groups.google.com/d/optout>을(를) 방문하세요.

# [**[Hibernate] NonUniqueObjectException: a different object with the same identifier value was already associated with the session**](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html)

### Problem

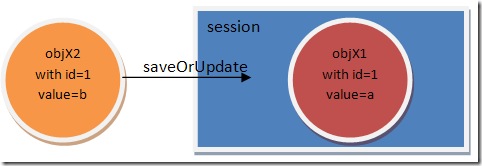
When saveOrUpdate() is called on hibernate template, it gives the above exception.

### Solution

1. use merge() instead of saveOrUpdate()or
2. load and set the value before calling saveOrUpdate()

### Reason

You have 2 objects: one in session and other you attempt to save both of which have the same identifier.

[](http://lh5.ggpht.com/_LFzrs9y4Am8/Sb9Tpi0bMvI/AAAAAAAAC4U/-TRotefhQmI/s1600-h/sessionObject%5b11%5d.jpg)

The object objX1 is already in the session, if you happened to load or find before calling saveOrUpdate().

When saveOrUpdate() is called on objX2, there is already an object, objX1, associated in the session with the same identifier (id=1) and it throws an exception.

### 1 – merge()

The simples way to sort this out is call merge() instead of saveOrUpdate().

The merge() method copy the state (value=b) of object objX2 to persistent instance, objX1. But it does not attach objX2 into session.

### 2 – load and set the value()

If you don’t want to or cannot use merge() , load and set the value to callsaveOrUpdate().

MyObject objX2 = new MyObject(1);

objX2.setValue(b);

hibernateTemplate.saveOrUpdate(objX2);

The above code may give the NonUniqueObjectException. So we load and set the value as following

MyObject objX2 = hibernateTemplate.get(objX2.class,new Integer(1));

if (objX2 == null){

   objX2 = new MyObject(1);

}

objX2.setValue(b);

hibernateTemplate.saveOrUpdate(objX2);

load() method will throw ObjectNotFoundException if no record found in database and thus get() method is used instead.

For the above code, objX2 is actually pointing to objX1 if there is already an object associated with the session.

### Automatic State Detection

According to “[Automatic state detection](http://www.hibernate.org/hib_docs/reference/en/html_single/#objectstate-saveorupdate)”

saveOrUpdate() does the following:

* if the object is already persistent in this session, do nothing
* if another object associated with the session has the same identifier, throw an exception
* if the object has no identifier property, save() it
* if the object's identifier has the value assigned to a newly instantiated object,save() it
* if the object is versioned (by a <version> or <timestamp>), and the version property value is the same value assigned to a newly instantiated object,save() it
* otherwise update() the object

and merge() is very different:

* if there is a persistent instance with the same identifier currently associated with the session, copy the state of the given object onto the persistent instance
* if there is no persistent instance currently associated with the session, try to load it from the database, or create a new persistent instance
* the persistent instance is returned
* the given instance does not become associated with the session, it remains detached

Posted 17th March 2009 by [Andy Myint](http://www.blogger.com/profile/04441850183513599266)

Labels: [Hibernate](http://pro-programmers.blogspot.kr/search/label/Hibernate)

18

### View comments

1. 

[**Sudhir**](http://www.blogger.com/profile/09080281818875815184)[March 25, 2009 at 8:05 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1238036700000#c6543628586551863147)

thanks for your post. It helped me understand the issue with non unique object in session and I was able to resolve my issue

[Reply](javascript:;)

1. http://img1.blogblog.com/img/b36-rounded.png

[**yuku**](http://www.blogger.com/profile/15226462280828485140)[September 3, 2009 at 7:57 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1252033038461#c339402403149881838)

Thanks :) So useful and solves my problem.

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**max4ever**[December 27, 2009 at 6:48 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1261925336914#c4084502692717307582)

thank you so much, it solved my problem, other pages didn't help

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**Nitro**[February 8, 2010 at 8:27 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1265646470895#c950104061314686369)

Thanks a lot!  
Very usefull post!

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**Anonymous**[March 13, 2010 at 11:17 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1268507853774#c6539493493196836438)

Awesome post! Really thanks, your explication helped me to solve a problem when I try to persist an object.

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**Anonymous**[May 6, 2010 at 11:04 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1273212285507#c3377276830092263216)

Good explanation with the Picture. I browsed many site to understand merge(), but your site gave a very good explanation.

[Reply](javascript:;)

1. http://img1.blogblog.com/img/b36-rounded.png

[**Jose Javier Ramirez Aguilar**](http://www.blogger.com/profile/02942639015699934335)[December 11, 2010 at 12:13 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1292098397185#c5781093692882283302)

Muchas Gracias Rajado me salvaste

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

[**Vitucho**](http://www.vitucho.com.ar/)[February 15, 2011 at 2:18 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1297808325610#c6269496995104323344)

you can use   
if (session.contains(referenceToObjX1))  
{  
session.evict(referenceToObjX1);  
}  
Then you can use saveOrUpdate(ObjX2) despite the fact that it has the same Id.  
  
Whatever, it's always better if you handle one entity using only one object reference. So, i do not recommed this practice except you must to.

[Reply](javascript:;)

1. http://img1.blogblog.com/img/b36-rounded.png

[**harsha**](http://www.blogger.com/profile/15983802731404153771)[June 21, 2011 at 4:53 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1308700387271#c5841641409145169212)

Awesome. i have searched a lot for finding the resolution but finally your post helped me resolving my problem. Thanks a lot

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**TheAnswer**[July 14, 2011 at 2:49 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1310636986075#c4136964316831787576)

Thanks for your post. It helped me.

[Reply](javascript:;)

1. http://img1.blogblog.com/img/b36-rounded.png

[**PJUK**](http://www.blogger.com/profile/04189030006340848747)[January 23, 2012 at 12:44 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1327308294647#c3189050645094809096)

Nice one! Got me going in the right direction.

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**Anonymous**[June 5, 2012 at 4:25 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1338895556895#c8415875077791255218)

Very clean explanation... Thanks

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**Anonymous**[November 23, 2012 at 4:48 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1353674903055#c3850197536182554520)

Crystal clear. Thank you!

[Reply](javascript:;)

[**Buyaroff Yuriy**](http://www.blogger.com/profile/01047799819601428545)[June 5, 2013 at 7:59 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1370444342019#c8837606966716624225)

Great help for me. Thanks!!!

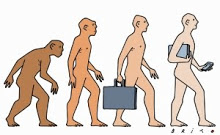
[Reply](javascript:;)

1. http://img1.blogblog.com/img/b36-rounded.png

[**Satyabrata Mohapatra**](http://www.blogger.com/profile/09848401445147295900)[July 30, 2013 at 5:12 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1375186332218#c7031140454606367732)

very nice one boss

[Reply](javascript:;)

1. 

[**SERGE ABOU-SALEH**](http://www.blogger.com/profile/01401634477182061022)[July 30, 2014 at 9:20 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1406737232174#c1838722683046692929)

Second Method worked for me, Thanks!!

[Reply](javascript:;)

1. http://img1.blogblog.com/img/anon36.png

**Anonymous**[March 17, 2015 at 1:40 PM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1426624808466#c3937941021213907763)

very good explanation, thanks

[Reply](javascript:;)

1. http://img1.blogblog.com/img/b36-rounded.png

[**Binh Thanh Nguyen**](http://www.blogger.com/profile/02069417158952884749)[November 20, 2015 at 1:52 AM](http://pro-programmers.blogspot.com/2009/03/hibernate-nonuniqueobjectexception.html?showComment=1448013174458#c6192951449855557115)

Thanks, nice explanation. It saved me.

[Reply](javascript:;)