34
Shares

26

7

1

| TUTORIAL | #INDEX POSTS | #INTERVIEW QUESTIONS | RESOURCES | STORE |

E » HIBERNATE » HQL – HIBERNATE QUERY LANGUAGE – EXAMPLE TUTORIAL

HQL – Hibernate Query Language – Example Tutorial

JNE 13, 2016 BY PANKAJ — 7 COMMENTS

HQL or Hibernate Query Language is the object-oriented query language of Hibernate Framework. HQL is very similar to SQL except that we use Objects instead of table names, that makes it more close to object oriented programming.

Hibernate Query Language – HQL

HQL and Case Sensitivity: HQL is case-insensitive except for java class and variable names. So `SeLeCT` is the same as `sELEct` is the same as `SELECT`, but `com.journaldev.model.Employee` is not same as `com.journaldev.model.EMPLOYEE`.

Some of the commonly supported clauses in HQL are:

1. **HQL From**: HQL From is same as select clause in SQL, `from Employee` is same as `select * from Employee`. We can also create alias such as `from Employee emp` or `from Employee as emp`.
2. **HQL Join** : HQL supports inner join, left outer join, right outer join and full join. For example, `select e.name, a.city from Employee e INNER JOIN e.address a`. In this query, Employee class should have a variable named address. We will look into it in the example code.
3. **Aggregate Functions**: HQL supports commonly used aggregate functions such as count(*), count(distinct x), min(), max(), avg() and sum().
4. **Expressions**: HQL supports arithmetic expressions (+, -, *, /), binary comparison operators (=, >=, <=, <>, !=, like), logical operations (and, or, not) etc.
5. HQL also supports ordre by and group by clauses.
6. HQL also supports sub-queries just like SQL queries.
7. HQL supports DDL, DML and executing store procedures too.

34
Shares

am using MySQL database for my example, below script will create two tables Employee and Address. They have one-to-one mapping and I am inserting some demo data for my example.

26

7

1

```sql
CREATE TABLE `Employee` (
  `emp_id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `emp_name` varchar(20) NOT NULL,
  `emp_salary` double(10,0) NOT NULL DEFAULT '0',
  PRIMARY KEY (`emp_id`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8;

CREATE TABLE `Address` (
  `emp_id` int(11) unsigned NOT NULL,
  `address_line1` varchar(50) NOT NULL DEFAULT '',
  `zipcode` varchar(10) DEFAULT NULL,
  `city` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`emp_id`),
  CONSTRAINT `emp_fk_1` FOREIGN KEY (`emp_id`) REFERENCES `Employee` (`emp_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

INSERT INTO `Employee` (`emp_id`, `emp_name`, `emp_salary`)
VALUES
        (1, 'Pankaj', 100),
        (2, 'David', 200),
        (3, 'Lisa', 300),
        (4, 'Jack', 400);


INSERT INTO `Address` (`emp_id`, `address_line1`, `zipcode`, `city`)
VALUES
        (1, 'Albany Dr', '95129', 'San Jose'),
        (2, 'Arques Ave', '95051', 'Santa Clara'),
        (3, 'BTM 1st Stage', '560100', 'Bangalore'),
        (4, 'City Centre', '100100', 'New Delhi');

commit;
```

Create a maven project in Eclipse or the IDE you are using, our final project will look like below image.

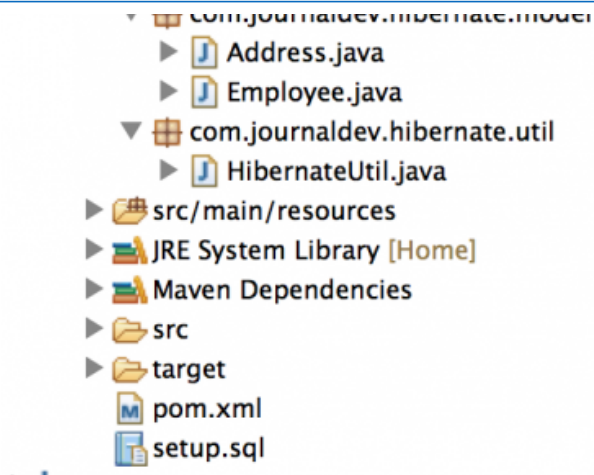   com.journaldev.hibernate.model
                    ▶ J Address.java
                    ▶ J Employee.java
                ▼ 🔲 com.journaldev.hibernate.util
                    ▶ J HibernateUtil.java
            ▶ 📁 src/main/resources
            ▶ 📚 JRE System Library [Home]
            ▶ 📚 Maven Dependencies
        ▶ 📂 src
        ▶ 📂 target
            M pom.xml
            📄 setup.sql

Hibernate Maven Dependencies

Our final pom.xml contains dependencies for Hibernate and MySQL driver.

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.journaldev.hibernate</groupId>
    <artifactId>HQLExample</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <dependencies>
        <dependency>
            <groupId>org.hibernate</groupId>
            <artifactId>hibernate-core</artifactId>
            <version>4.3.5.Final</version>
        </dependency>
        <dependency>
            <groupId>mysql</groupId>
```

34
Shares

```
</project>
```

Hibernate Configuration XML

26

Our hibernate configuration xml file contains database connection related properties and mapping classes. I

7   will be using annotations for Hibernate mapping.

1   hibernate.cfg.xml code:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
                "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
                "http://hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
    <session-factory>
        <property
name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
        <property name="hibernate.connection.password">pankaj123</property>
        <property
name="hibernate.connection.url">jdbc:mysql://localhost/TestDB</property>
        <property name="hibernate.connection.username">pankaj</property>
        <property
name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

        <property
name="hibernate.current_session_context_class">thread</property>
        <property name="hibernate.show_sql">true</property>

        <mapping class="com.journaldev.hibernate.model.Employee"/>
        <mapping class="com.journaldev.hibernate.model.Address"/>
    </session-factory>
</hibernate-configuration>
```

Hibernate SessionFactory Utility class

We have a utility class to configure hibernate SessionFactory.

```java
    private static SessionFactory sessionFactory;

    private static SessionFactory buildSessionFactory() {
    try {
        // Create the SessionFactory from hibernate.cfg.xml
            Configuration configuration = new Configuration();
            configuration.configure("hibernate.cfg.xml");
            System.out.println("Hibernate Configuration loaded");
```

34
Shares

26

7

1

```java
            System.out.println("Hibernate serviceRegistry created");

            SessionFactory sessionFactory =
configuration.buildSessionFactory(serviceRegistry);

            return sessionFactory;
        }
        catch (Throwable ex) {
            System.err.println("Initial SessionFactory creation failed." +
ex);
            ex.printStackTrace();
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory() {
            if(sessionFactory == null) sessionFactory =
buildSessionFactory();
        return sessionFactory;
    }
}
```

Model Classes with Annotation based mapping

Our model classes with JPA annotations looks like below.

```java
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.OneToOne;
import javax.persistence.Table;

import org.hibernate.annotations.Cascade;

@Entity
@Table(name = "EMPLOYEE")
public class Employee {

        @Id
        @GeneratedValue(strategy = GenerationType.IDENTITY)

        @Column(name = "emp_id")
        private long id;

        @Column(name = "emp_name")
        private String name;
```

```java
@Cascade(value = org.hibernate.annotations.CascadeType.ALL)
private Address address;

public long getId() {
        return id;
}

public void setId(long id) {
        this.id = id;
}

public Address getAddress() {
```

```java
package com.journaldev.hibernate.model;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;
import javax.persistence.OneToOne;
import javax.persistence.PrimaryKeyJoinColumn;
import javax.persistence.Table;

import org.hibernate.annotations.GenericGenerator;
import org.hibernate.annotations.Parameter;

@Entity
@Table(name = "ADDRESS")
public class Address {

        @Id
        @Column(name = "emp_id", unique = true, nullable = false)
        @GeneratedValue(generator = "gen")
        @GenericGenerator(name = "gen", strategy = "foreign",
                                parameters = { @Parameter(name = "property",
value = "employee") })
        private long id;

        @Column(name = "address_line1")
        private String addressLine1;

        @Column(name = "zipcode")
        private String zipcode;

        @Column(name = "city")
```

34
Shares

et's see how to use HQL in java programs.

26

7

1

```java
        query = session.createQuery("select e.name, a.city from Employee e "
                        + "INNER JOIN e.address a");
        List<Object[]> list = query.list();
        for(Object[] arr : list){
                System.out.println(Arrays.toString(arr));
        }

        //HQL group by and like example
        query = session.createQuery("select e.name, sum(e.salary), count(e)"
                        + " from Employee e where e.name like '%i%' group by e.name");
        List<Object[]> groupList = query.list();
        for(Object[] arr : groupList){
                System.out.println(Arrays.toString(arr));
        }

        //HQL order by example
        query = session.createQuery("from Employee e order by e.id desc");
        empList = query.list();
        for(Employee emp3 : empList){
                System.out.println("ID Desc Order Employee::"+emp3.getId()+","+emp3.getAddress().getCity());
        }

        //rolling back to save the test data
        tx.rollback();

        //closing hibernate resources
```

34
Shares

Notice that I am using HQL for Select, Update and Delete operations. It also shows how to use HQL Join and
HQL Aggregate functions.

26   When I run above hql example program, we get following output.

```
org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderI
 configure
WARN: HHH000402: Using Hibernate built-in connection pool (not for production
use!)
May 22, 2014 1:55:37 PM
org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderI
 buildCreator
INFO: HHH000401: using driver [com.mysql.jdbc.Driver] at URL
[jdbc:mysql://localhost/TestDB]
May 22, 2014 1:55:37 PM
org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderI
 buildCreator
INFO: HHH000046: Connection properties: {user=pankaj, password=****}
May 22, 2014 1:55:37 PM
org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderI
 buildCreator
INFO: HHH000006: Autocommit mode: false
May 22, 2014 1:55:37 PM
org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderI
 configure
INFO: HHH000115: Hibernate connection pool size: 20 (min=1)
May 22, 2014 1:55:37 PM org.hibernate.dialect.Dialect <init>
INFO: HHH000400: Using dialect: org.hibernate.dialect.MySQLDialect
May 22, 2014 1:55:37 PM org.hibernate.engine.jdbc.internal.LobCreatorBuilder
useContextualLobCreation
INFO: HHH000423: Disabling contextual LOB creation as JDBC driver reported
JDBC version [3] less than 4
May 22, 2014 1:55:38 PM
org.hibernate.engine.transaction.internal.TransactionFactoryInitiator
initiateService
INFO: HHH000399: Using default transaction strategy (direct JDBC
transactions)
May 22, 2014 1:55:38 PM
org.hibernate.hql.internal.ast.ASTQueryTranslatorFactory <init>
```

Notice that once delete operation is performed, further operations are not showing that record data (sum of
salary is 600). However I am rolling back the transaction, so the data in table will remain unchanged. Change
the code to commit the transaction and it will be reflected in the database tables.

Your email address please..

SEND ME NOW!

LED UNDER: HIBERNATE

7

1

ut Pankaj

u have come this far, it means that you liked what you are reading. Why not reach little more and connect
me directly on **Google Plus**, **Facebook** or **Twitter**. I would love to hear your thoughts and opinions on my
:les directly.

ently I started creating video tutorials too, so do check out my videos on **Youtube**.

bernate Tools Eclipse Plugin

org.hibernate.HibernateException: Could not parse configuration: hibernate.cfg.xml »

Comments

pavan kumar says
MARCH 6, 2017 AT 5:28 AM

can i know how to use mappings to store data by using form

Reply

Vijay Raj R says
FEBRUARY 17, 2017 AT 9:30 PM

For Pagination we must use setMaxResults(2) instead of setFetchSize(2).

Reply

Issak Madeena RahamaTullah shaik says
JANUARY 27, 2015 AT 11:01 PM

Hi
I have requirement i.e I need to join morethan two tables using criteria api, so please guide me in this
regard

Your email address please..

SEND ME NOW!

34
Shares

Shailesh says
JUNE 30, 2014 AT 4:13 AM

26

Can you please add some tutorials about inheritance in hibernate.

Also what is 'discriminator' in hibernate??

7

It was asked in an interview.

1

Reply

Farhan says
MAY 22, 2014 AT 3:43 AM

Hi Pankaj,

Your HQL example is good but I need more and more for example. all joins examples with one-to-many and many-to-many association for HQL. some example of HQL method that is used with collection properities.
HQL using Map, List, wrapper object in it select clause for example select new List(empId, empName, empSex) from employees etc other stuff like pagination and etc

Reply

Pankaj says
MAY 22, 2014 AT 8:30 AM

Okay, I will add few more in the tutorial.

Reply

Pankaj says
MAY 22, 2014 AT 1:57 PM

I have added some more examples of order by, group by, pagination, like etc.

Reply

Your email address please..

SEND ME NOW!

34
Shares

26

7

1

Comment

Name *

Email *

Website

POST COMMENT

Custom Search

Hibernate Tutorial

34
Shares

26

7

1

› Hibernate Session get load

› Hibernate Session save

› HQL Example

› Hibernate Criteria

› Hibernate SQL

› Hibernate Named Query

› Hibernate Log4J

› Hibernate Validator

› Hibernate Tomcat DataSource

Hibernate Mapping

› Hibernate One to One Mapping

› Hibernate One to Many Mapping

› Hibernate Many to Many Join Tables

Hibernate Caching

› Hibernate Cache

› Hibernate EHCache

Hibernate Integrations

› Hibernate Spring

34
Shares

26

7

1

› Hibernate Primefaces

› Hibernate Primefaces Spring

› Hibernate SpringRoo Primefaces

› Hibernate JSF Spring

Miscellaneous

› Hibernate Tools Eclipse Plugin

› Hibernate Configuration Offline

› [Solved] No identifier specified

› Hibernate Program Not Terminating

› Access to DialectResolutionInfo

› get is not valid

› No CurrentSessionContext configured

› Hibernate Interview Questions

STAY UPDATED!

Name

E-Mail Address

I AM IN!

RECOMMENDED TUTORIALS

Java Tutorials: Java IO Tutorial, Java
Regular Expressions Tutorial,
Multithreading in Java, Java Logging API
Tutorial, Java Annotations, Java XML

Free eBook: Java Design Patterns (130 Pages) Download Now!   ▲

Your email address please..

SEND ME NOW!

34
Shares

26

7

1

Java EE: Servlet JSP Tutorial, Struts2 Tutorial, Spring Tutorial, Hibernate Tutorial, Primefaces Tutorial

Web Services: Apache Axis 2 Tutorial, JAX-RS Web Services Tutorial

Misc: Memcached Tutorial

Resources: Free eBooks, My Favorite Web Hosting

IMPORTANT INTERVIEW QUESTIONS

Java String Interview Questions, Java Multithreading Interview Questions, Java Programming Interview Questions, Java Interview Questions, Java Collections Interview Questions, Java Exception Interview Questions

Servlet Interview Questions, JSP Interview Questions, Struts2 Interview Questions, JDBC Interview Questions, Spring Interview Questions, Hibernate Interview Questions