

[JAVA TUTORIAL](#)[#INDEX POSTS](#)[#INTERVIEW QUESTIONS](#)[RESOURCES](#)[HIRE ME](#)[DOWNLOAD ANDROID APP](#)[CONTRIBUTE](#)**Subscribe to Download Java Design Patterns eBook****DOWNLOAD NOW**[HOME](#) » [HIBERNATE](#) » [HIBERNATE CACHING - FIRST LEVEL CACHE](#)

Hibernate Caching – First Level Cache

APRIL 2, 2018 BY [PANKAJ](#) — [12 COMMENTS](#)

Welcome to Hibernate Caching – First Level Cache Example Tutorial. Recently we looked into [Hibernate architecture](#), [hibernate mapping](#) and how to use [HQL](#) for firing SQL queries in object oriented way. Today we will look into one of the important aspects of Hibernate – **Hibernate Cache**.

Hibernate Caching



Hibernate Cache can be very useful in gaining fast application performance if used correctly. The idea behind cache is to reduce the number of database queries, hence reducing the throughput time of the application.

Hibernate comes with different types of Cache:

1. **First Level Cache:** Hibernate first level cache is associated with the Session object. Hibernate first level cache is enabled by default and there is no way to disable it. However hibernate provides methods through which we can delete selected objects from the cache or clear the cache completely. Any object cached in a session will not be visible to other sessions and when the session is closed, all the cached objects will also be lost.
2. **Second Level Cache:** Hibernate Second Level cache is disabled by default but we can enable it through configuration. Currently EHCACHE and Infinispan provides implementation for Hibernate Second level cache and we can use them. We will look into this in the next tutorial for hibernate caching.
3. **Query Cache:** Hibernate can also cache result set of a query. Hibernate Query Cache doesn't cache the state of the actual entities in the cache; it caches only identifier values and results of value type. So it should always be used in conjunction with the second-level cache.

Hibernate Caching – First Level Cache Example

For my hibernate first level cache example program, I am using same configuration as in [HQL Example](#), you can check that out and configure the tables and populate it with dummy data.

Let's first look into the program, it's output and then we will go through some of the important points related to Hibernate First Level Cache.

HibernateCacheExample.java

```
package com.journaldev.hibernate.main;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import com.journaldev.hibernate.model.Employee;
import com.journaldev.hibernate.util.HibernateUtil;

public class HibernateCacheExample {

    public static void main(String[] args) throws InterruptedException {

        SessionFactory sessionFactory = HibernateUtil.getSessionFactory();
```

```

Session session = sessionFactory.getCurrentSession();
Transaction tx = session.beginTransaction();

//Get employee with id=1
Employee emp = (Employee) session.load(Employee.class, new Long(1));
printData(emp,1);

//waiting for sometime to change the data in backend

```

When we run above example, the output contains a lot of hibernate related information. But we are mostly interested in our code specific output and the queries fired by hibernate to load the data. The output snippet looks like below.

```

Hibernate Configuration loaded
Hibernate serviceRegistry created
Hibernate: select employee0_.emp_id as emp_id1_1_0_, employee0_.emp_name as
emp_name2_1_0_, employee0_.emp_salary as emp_sala3_1_0_, address1_.emp_id as
emp_id1_0_1_, address1_.address_line1 as address_2_0_1_, address1_.city as
city3_0_1_, address1_.zipcode as zipcode4_0_1_ from EMPLOYEE employee0_ left outer
join ADDRESS address1_ on employee0_.emp_id=address1_.emp_id where
employee0_.emp_id=?
1:: Name=Pankaj, Zipcode=95129
2:: Name=Pankaj, Zipcode=95129
Hibernate: select employee0_.emp_id as emp_id1_1_0_, employee0_.emp_name as
emp_name2_1_0_, employee0_.emp_salary as emp_sala3_1_0_, address1_.emp_id as
emp_id1_0_1_, address1_.address_line1 as address_2_0_1_, address1_.city as
city3_0_1_, address1_.zipcode as zipcode4_0_1_ from EMPLOYEE employee0_ left outer
join ADDRESS address1_ on employee0_.emp_id=address1_.emp_id where
employee0_.emp_id=?
3:: Name=PankajK, Zipcode=95129
Hibernate: select employee0_.emp_id as emp_id1_1_0_, employee0_.emp_name as
emp_name2_1_0_, employee0_.emp_salary as emp_sala3_1_0_, address1_.emp_id as
emp_id1_0_1_, address1_.address_line1 as address_2_0_1_, address1_.city as
city3_0_1_, address1_.zipcode as zipcode4_0_1_ from EMPLOYEE employee0_ left outer
join ADDRESS address1_ on employee0_.emp_id=address1_.emp_id where

```

First Level Cache in Hibernate Important Points

Important Points about First level cache in Hibernate that can be derived from above program are:

1. Hibernate First Level cache is enabled by default, there are no configurations needed for this.
2. Hibernate first level cache is session specific, that's why when we are getting the same data in same session there is no query fired whereas in other session query is fired to load the data.

3. Hibernate first level cache can have old values, as you can see above that I have put my program to sleep for 10 seconds and in that time I updated the value (name from Pankaj to PankajK) in database but it didn't get reflected in the same session. But in other session, we got the updated value.
4. We can use session `evict()` method to remove a single object from the hibernate first level cache.
5. We can use session `clear()` method to clear the cache i.e delete all the objects from the cache.
6. We can use session `contains()` method to check if an object is present in the hibernate cache or not, if the object is found in cache, it returns true or else it returns false.
7. Since hibernate cache all the objects into session first level cache, while running bulk queries or batch updates it's necessary to clear the cache at certain intervals to avoid memory issues.

That's all for hibernate caching and first level cache example, in future posts we will look into [Hibernate Second Level Cache - EHCACHE](#) implementation.



« PREVIOUS

[Hibernate Criteria Example Tutorial](#)

NEXT »

[Hibernate EHCACHE – Hibernate Second Level Cache](#)

About Pankaj

If you have come this far, it means that you liked what you are reading. Why not reach little more and connect with me directly on [Google Plus](#), [Facebook](#) or [Twitter](#). I would love to hear your thoughts and opinions on my articles directly.

Recently I started creating video tutorials too, so do check out my videos on [Youtube](#).

FILED UNDER: [HIBERNATE](#)

Comments

Jaspreet Singh says

OCTOBER 12, 2017 AT 11:42 PM

You should use get method or add comment before print method that query is fired or not because load will never fire a query. And it can confuse new learners.

[Reply](#)

Rohit Roy says

APRIL 14, 2017 AT 8:41 AM

Isn't printData(emp4,5); supposed to show name as PankajK as it is hitting database where data was updated earlier during Thread.sleep(10000).

Correct me if I am wrong.

[Reply](#)

Deep says

JULY 8, 2017 AT 11:49 PM

I also have same doubt

[Reply](#)

Vijay Raj R says

FEBRUARY 11, 2017 AT 11:31 PM

Very clearly explained, on the first read it was a little bit confusing but eventually ended with a very good explanation.

[Reply](#)

Aj says

JULY 13, 2016 AT 11:09 PM

Is criteria uses the cache technique or only get() and load() with will use it.

[Reply](#)

Ram says

JUNE 17, 2016 AT 2:38 AM

Nice explanation.....

[Reply](#)

Chintan says

MARCH 17, 2015 AT 10:18 PM

I am getting following error when I try to runt the program. Can you please provide the solution?

org.hibernate.AnnotationException: Unknown mappedBy in: com.test.model.Employee.address,

referenced property unknown: com.test.model.Address.employee

at org.hibernate.cfg.OneToOneSecondPass.doSecondPass(OneToOneSecondPass.java:153)

at org.hibernate.cfg.Configuration.originalSecondPassCompile(Configuration.java:1697)

at org.hibernate.cfg.Configuration.secondPassCompile(Configuration.java:1426)

at org.hibernate.cfg.Configuration.buildSessionFactory(Configuration.java:1846)

at com.test.util.HibernateUtilities.buildAnnotationSessionFactory(HibernateUtilities.java:52)

at com.test.util.HibernateUtilities.getSessionAnnotationFactory(HibernateUtilities.java:99)

at com.test.HibernateCacheExample.main(HibernateCacheExample.java:14)

[Reply](#)

Beena says

MARCH 14, 2015 AT 11:15 AM

awesome explanation

[Reply](#)

shoeb JD says

FEBRUARY 18, 2015 AT 8:42 AM

good example .

[Reply](#)

Ravi says

OCTOBER 24, 2014 AT 8:35 PM

Its pretty clear about each comments on every line of code. No diversion/confusion about understanding about the concepts how hibernate internally works!!!

Me too waiting for your upcoming blogs. I am interested to know Spring Channels, Dozer Mapper, Spring Cocoon, Spring Transaction, Spring Cache framework.

Thanks for KIS (Keep It Simple) comments ☐

-Ravi

[Reply](#)

Shashi says

JUNE 29, 2014 AT 9:49 PM

Very good Concept

[Reply](#)

shareef says

JUNE 2, 2014 AT 10:05 PM

i will wait for your next tutorial my friend , nice and short cool

[Reply](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *



Save my name, email, and website in this browser for the next time I comment.

POST COMMENT

Search for tutorials...

DOWNLOAD ANDROID APP



HIBERNATE FRAMEWORK

Hibernate Tutorial

- > [Hibernate Example](#)
- > [Hibernate SessionFactory](#)
- > [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](#)
- > [Hibernate Spring MVC](#)
- > [Hibernate Struts 2](#)
- > [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](#)

RECOMMENDED TUTORIALS

Java Tutorials

- > [Java IO](#)
- > [Java Regular Expressions](#)
- > [Multithreading in Java](#)
- > [Java Logging](#)
- > [Java Annotations](#)
- > [Java XML](#)
- > [Collections in Java](#)
- > [Java Generics](#)
- > [Exception Handling in Java](#)
- > [Java Reflection](#)
- > [Java Design Patterns](#)
- > [JDBC Tutorial](#)

Java EE Tutorials

- > [Servlet JSP Tutorial](#)
- > [Struts2 Tutorial](#)
- > [Spring Tutorial](#)
- > [Hibernate Tutorial](#)

- > [Primefaces Tutorial](#)
- > [Apache Axis 2](#)
- > [JAX-RS](#)
- > [Memcached Tutorial](#)