

# Difference Between Hibernate get() and load() Methods ?

Hibernate » on Sep 18, 2014 { 98 Comments } By Sivateja

Like 0

G+

What is the difference between hibernate `get()` and `load()` methods ? this is one of the famous hibernate interview questions, most of us will use hibernate get and load methods irrespective of knowing their exact behavior 😊 will you accept that ? of course even myself too in the initial stages 😊 Today i will try to clear that confusion.

## Few Points About Hibernate get() & load()

Both are from `Session` interface, and we will call them as `session.get()` & `session.load()`

Both will be use for retrieving the object (a row) from the database

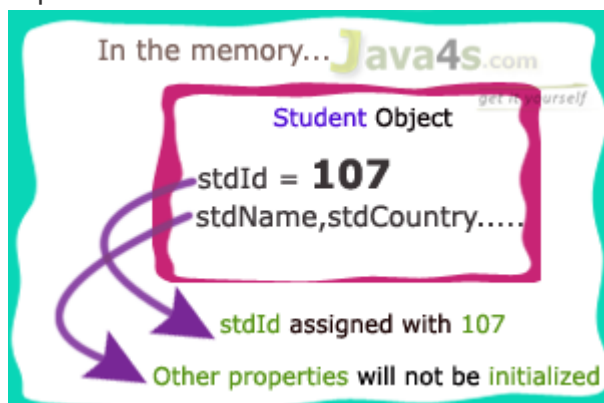
Then what's the difference them ? lets start with `load()` and then `get()` method.

Consider a Student class having 3 properties `stdId`, `stdName`, `stdCountry`

### 1. session.load()

When you call `session.load()` method, it will always return a "proxy" object, whats the meaning of proxy object ?

Proxy means, hibernate will prepare some fake object with given *identifier value* in the memory without hitting the database, for example if we call `session.load(Student.class, new Integer(107));` > hibernate will create one fake Student object [row] in the memory with id 107, but remaining properties of Student class will not even be initialized, observe this graphical representation...



It will hit the database only when we try to retrieve the other properties of Student object i mean stdName, stdCountry. If we call s2.getStdName() then hibernate will hit the database and search the row with student id 107 and retrieve the values, if object [row] not found in the database it will throws **ObjectNotFoundException**.,

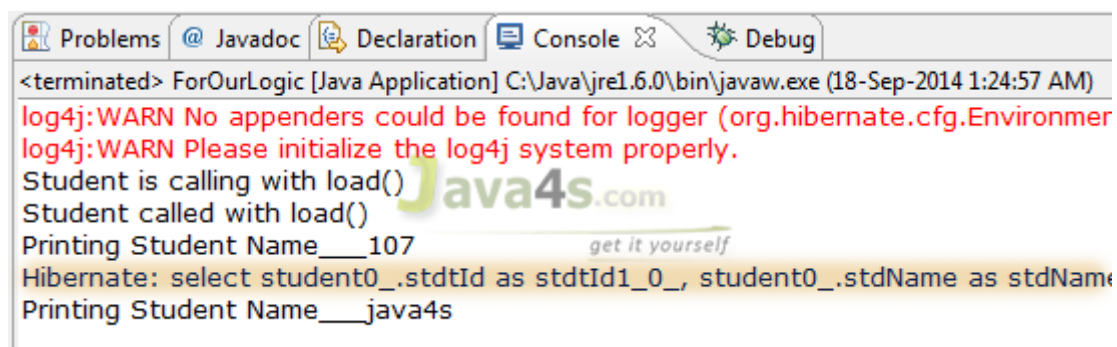
Let me explain each point by taking an example

## Example of session.load()

index.jsp:

```
1 System.out.println("Student is calling with load()");
2 s2 =(Student) session.load(Student.class,new Integer(107));
3 System.out.println("Student called with load()");
4 System.out.println("Printing Student Name____"+s2.getStdId());
5 System.out.println("Printing Student Name____"+s2.getStdName());
```

## Output



```
<terminated> ForOurLogic [Java Application] C:\Java\jre1.6.0\bin\javaw.exe (18-Sep-2014 1:24:57 AM)
log4j:WARN No appenders could be found for logger (org.hibernate.cfg.Environment)
log4j:WARN Please initialize the log4j system properly.
Student is calling with load()
Student called with load()
Printing Student Name____107
Hibernate: select student0_.stdId as stdId1_0_, student0_.stdName as stdName
Printing Student Name____java4s
```

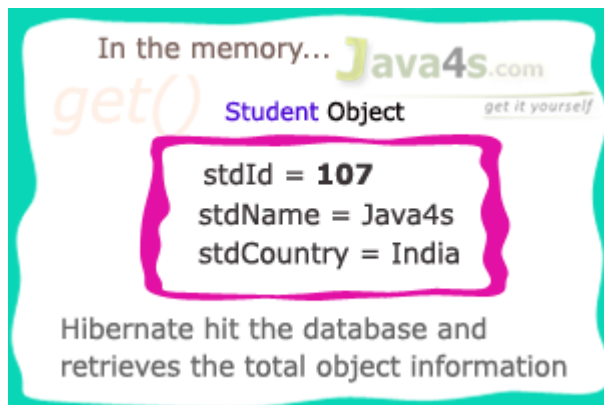
## Explanation:

In index.jsp > line number 4, i have called **s2.getStdId()** and hibernate simply printed 107 [at Output > line number 3] without creating any database query why ? because as i have explained hibernate will prepare some fake object with given *identifier value* in the memory without hitting the database. So when we call load() method (at index.jsp > line number 2 ) with 107 value, hibernate will create a fake object with 107 as identifier value, i mean temporarily it will consider the student id as 107. If you observe the output, hibernate was generated the database query (at Output > line number 4 ) when we called **s2.getStdName()**; ( index.jsp > line number 5 ).

That's it, so finally we came to know that session.load() will hit the database only when we start retrieving the object (row) values.

## 2. session.get()

When you call session.get() method, it will hit the **database** immediately and returns the original object



If the row is not available in the database, it returns **null**

## Example of session.get()

```
1 System.out.println("Student is calling with get()");
2 s1 = (Student) session.get(Student.class,new Integer(107));
3 System.out.println("Student called with get()");
4 System.out.println("Printing Student Name____"+s1.getStdId());
5 System.out.println("Printing Student Name____"+s1.getStdName());
```

## Output

```
<terminated> ForOurLogic [Java Application] C:\Java\jre1.6.0\bin\javaw.exe (18-Sep-2014 1:33:12 AM)
log4j:WARN No appenders could be found for logger (org.hibernate.cfg.Environment)
log4j:WARN Please initialize the log4j system properly.
Student is calling with get()
Hibernate: select student0_.stdId as stdId1_0_, student0_.stdName as stdName
Student called with get()
Printing Student Name____107
Printing Student Name____java4s
```

## Explanation:

Nothing to explain here 😊 as i told, when we call session.get() method hibernate will hit the database and returns the original object [ row ], that's the reason it was generated a query [ check line number 2 in the output screen]

Hmm..., so which is the best method to use, hibernate load() or get() ? its completely your choice 😊

## You Might Also Like

[Difference between ServletConfig and ServletContext in Java](#)  
[Example On Hibernate Select Query](#)