



Hibernate - get() and load() Method

Last Updated : 28 Apr, 2025

Hibernate is a Java framework that provides a powerful set of tools for persisting and accessing data in a Java environment. It is often used in conjunction with Spring. Spring and Hibernate are both widely used in the Java community, and they can be used together to build powerful and efficient Java-based applications.

Let's deal with Fetching objects in Spring Hibernate:

Hibernate provides different methods to fetch data from the database.

1. **get()**
2. **load()**

1. get() method

- **get()** method is used to retrieve a persistent object from the database. It is a member of the Session interface, and it takes the class of the object to be retrieved and the primary key of the object as arguments.
- **get()** method only hits the database if the object is not present in the session cache. If the same object is already present in the cache then it returns previously stored data from the cache.
- **get()** method returns null if there is no object present in the database.

Example:

```
// Open a session
Session session = sessionFactory.openSession();

// Begin a transaction
```



```
Transaction transaction = session.beginTransaction();

// Retrieve the object using the primary key
Customer customer = (Customer) session.get(Customer.class, 1L);

// Commit the transaction
transaction.commit();

// Close the session
session.close();
```

In this example, the `get()` method is used to retrieve a `Customer` object with a primary key of 1. The object is then cast to the `Customer` class and stored in the custom variable.

2. load() method

- **load()** method is used to retrieve an object from the database by its identifier (primary key). It is used to initialize a proxy object instead of a fully-initialized object, so it can be used to lazily load the object when it is needed.
- **load()** method does not retrieve the object from the database when it is called. Instead, it returns a proxy object that represents the object. The actual object is only retrieved from the database when it is needed, such as when a method of the object is called or a property is accessed. This technique is known as "lazy loading" and it is used to improve the performance of Hibernate by avoiding unnecessary database queries.
- **load()** method throws `ObjectNotFoundException` if there is no object found in the database.

Example:

```
Session session = sessionFactory.openSession();
Transaction transaction = session.beginTransaction();

// Load the entity with the identifier 1
Employee employee = (Employee) session.load(Employee.class, 1L);

// Print out the employee's name
System.out.println(employee.getName());
```



```
transaction.commit();  
session.close();
```

In this example, the load() method is used to retrieve an Employee object with a primary key of 1. The object is then cast to the Employee class and stored in the employee variable and printed the employee object.

[Comment](#)[More info](#)[Campus Training Program](#)**Corporate & Communications Address:**

A-143, 7th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305)

Registered Address:

K 061, Tower K, Gulshan Vivante
Apartment, Sector 137, Noida, Gautam
Buddh Nagar, Uttar Pradesh, 201305



Advertise with us

Company

[About Us](#)
[Legal](#)
[Privacy Policy](#)
[Careers](#)
[Contact Us](#)

Explore

[POTD](#)
[Job-A-Thon](#)
[Connect](#)
[Community](#)
[Videos](#)