JdbcTemplate vs HibernateTemplate

Both **HibernateTemplate** and **JdbcTemplate** is a helper class which provides different methods for querying/retrieving data from the database. It also converts checked exceptions into unchecked DataAccessExceptions.

Spring provides support for both hibernate and JDBC template classes. It provides template classes which contains all common code. But JDBC as we all know is not an ORM tool it does not represent rows as objects whereas Hibernate does that.

If you are using hibernate in your project then recommend to use HibernateTemplate and if you are not using hibernate then recommend to use JdbcTemplate class.

**Why Hibernate Over JDBC**

1. In JDBC the table name and column names are tightly coupled with your application, so any change in the database you have to modify the code, so it needs again development cost, as well as testing cost, which is waste of time, money and energy.

But in Hibernate The class name and columns are not tightly coupled with classes, those arte configured with mapping file(one xml), if any change in DB you can change in Mapping file only.

2. Hibernate stores directly objects but jdbc can not,so Hibernate supports ORM but JDBC can not.

3. Hibernate support versioning, that means if you modified one record (Object ) multiple times then it can be identified , how many times the object or record modified.

4. Hibernate supports lazy loading

5. Hibernate supports cacheing mechanism.

6. Hibernate supports Database independent queries. (HQL)