

# Performing CRUD Operations with Hibernate

---

## INTRODUCTION



**Sekhar Srinivasan**

@sekharonline4u    [www.sekhartheguru.net](http://www.sekhartheguru.net)



# Overview

**Introduction to  
Hibernate**

**Set up the  
development  
environment**

**Configure  
Hibernate  
Properties**

**Creating ORM  
Entity**

**Performing CRUD  
Operations**



# Overview of Hibernate

---



# Overview of Hibernate

①

Why to learn Hibernate?

②

What is Hibernate and what are the benefits?

③

Relationship between Hibernate and JDBC



Why we need to learn  
Hibernate?





```
public class Account {  
    public int accountNo;  
    public String  
    accountHolderName;  
    public decimal balance;  
}
```

```
Account account = new Account();  
account.accountNo = 101;  
account.accountHolderName="Sek  
har";  
account.balance = 10000;
```

Accno	Name	Balance
101	Sekhar	10000



What is all  
these our  
question is  
why we need  
to learn  
Hibernate?





Object Relational Mapping  
(ORM)



SQL  
Server  
MySQL

```
public class Account {  
    public int accountNo;  
    public String  
    accountHolderName;  
    public decimal balance;  
}
```

```
Account account = new Account();  
account.accountNo = 101;  
account.accountHolderName="Sek  
har";  
account.balance = 10000;
```

Oracle		
Accno	Name	Balance
101	Sekhar	10000

Hibernate ORM  
or Hibernate is a  
framework  
which supports  
ORM



What is Hibernate and what  
are the benefits of using  
Hibernate?





# What is Hibernate?

Hibernate is an Object Relational Mapping tool for the Java Programming Language.

Provides Framework for mapping an Object Oriented Domain Model to a Relational Database



# Benefits of Using Hibernate

Handles Object-Relation impedance mismatch problems

Supports Persistent database accesses

Provides Data query and Retrieval facilities

Handles all the low level SQL Code

Reduces the JDBC code for interacting with database



Many new Components has been added  
Core Functionality is supported  
ORM

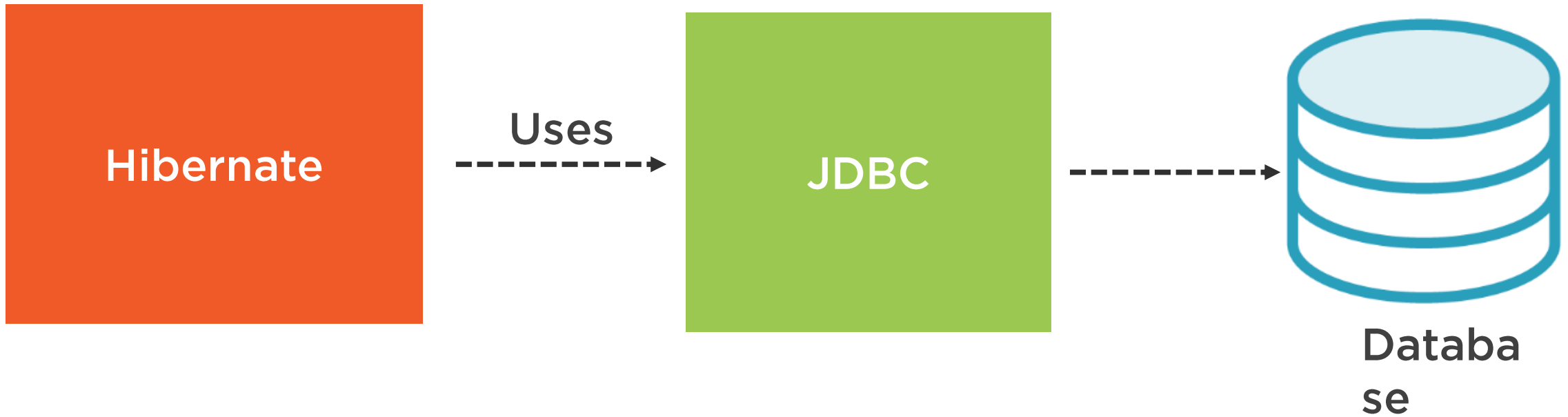
-Hibernate Annotations, Hibernate  
Entity Manager, Hibernate OGM,  
Hibernate Validator



What is the relationship  
between Hibernate and  
JDBC?



# Relationship Between Hibernate and JDBC



# Setting up Development Environment for Supporting Hibernate Framework

---



# Setting up Development Environment for Supporting Hibernate Framework

**Download Hibernate Jar Files**

**Add Jar files to Project**



# Configuring Hibernate Properties using XML File

---



# Steps to Configure Hibernate with Spring MVC

①

Define Database Data Source

②

Set up Hibernate Session Factory

③

Set up Transaction Manager

④

Enable Configuration of Transactional Annotations



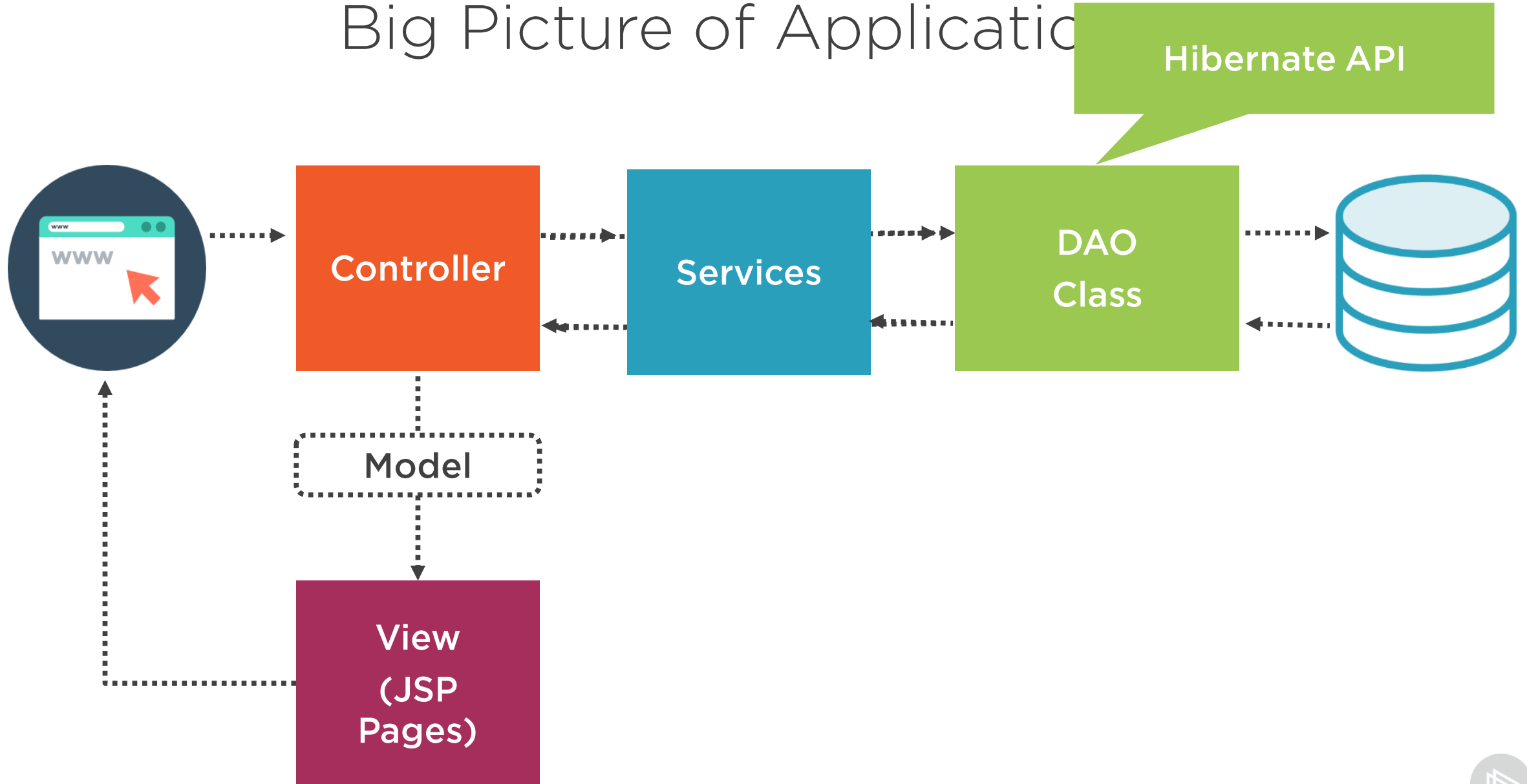


# Demo: Inserting a Record

---



# Big Picture of Application



# Demo: Using HQL for Retrieving the Records

---



# What is HQL?

Hibernate Query Language is an Object Oriented Query Language similar to SQL, but instead of operating on the Tables and Columns, HQL works with Persistent Objects and their Properties.



```
Query query =  
session.createQuery( "From  
Entity" );
```

```
Query query =  
session.createQuery ( "From  
Entity as E" );
```

```
Query query =  
session.createQuery ( "From  
Entity E" );
```

```
Query query =  
session.createQuery( "Select  
E.FieldName From Entity E" );
```

#### ◀ **From Clause**

◀ Used to load complete persistent objects into memory

#### ◀ **As Clause**

Used to provide an alias to the Entity class

#### ◀ **Select Clause**

Provides more control over the result set. Used to retrieve few properties of object instead of the complete object



```
Query query =  
session.createQuery( "From  
Element E where E.FieldName =  
value" );
```

```
Query query =  
session.createQuery( "From  
Element E Order by  
E.FieldName DESC" );
```

#### ◀ Where clause

Used to filter the result set based on the condition

#### ◀ Orderby clause

Used to sort the data based on the specified field name



Keywords like From, Select, Where and Order by are not case sensitive but the properties like table and column names are case sensitive in HQL



# Demo: Updating the Record

---

