V1.0

Training Program on Hibernate

4P Advisory Services

www.4pa.in

Hibernate Training Program

What is Hibernate?

Hibernate is an open-source object-relational mapping (ORM) framework that provides a way to map Java objects to relational database tables and vice versa. It is a popular choice for developers who want to simplify database access and eliminate repetitive SQL coding.

Hibernate works by creating a layer of abstraction between the application and the database, allowing developers to work with objects and classes rather than SQL queries. With Hibernate, developers can create persistent objects that are automatically synchronized with the database, eliminating the need to write boilerplate code for database operations.

Hibernate supports various databases and provides a wide range of features, such as caching, lazy loading, transaction management, and automatic key generation. Hibernate also supports annotations and XML configuration files to define the mapping between the Java objects and database tables.

Hibernate is a powerful ORM framework that simplifies database access, increases developer productivity, and improves application performance.

Advantages of Hibernate for Java Programming:

- Simplifies Database Access: Hibernate provides a simple and intuitive API for working with relational databases. This allows developers to focus on the business logic of their application, rather than the details of interacting with the database.
- Reduces Development Time: Hibernate automates much of the mapping between object-oriented code and relational data, which can significantly reduce the amount of boilerplate code that developers need to write. This, in turn, can lead to faster development times and increased productivity.
- Improves Application Performance: Hibernate provides built-in caching mechanisms that can improve application performance by reducing the number of database queries and minimizing network traffic. Additionally, Hibernate supports lazy loading, which allows data to be loaded on-demand, further improving performance.
- Supports Object-Oriented Programming: Hibernate supports object-oriented programming by allowing developers to work with entities, which are Java classes that represent database tables. This makes it easy to work with data in an object-oriented way, which can lead to more maintainable and extensible code.
- Facilitates Database Portability: Hibernate provides a high degree of database portability, which means that applications developed with Hibernate can easily be switched from one database vendor to another without requiring significant code changes.
- Provides Powerful Querying Capabilities: Hibernate provides a powerful query language called Hibernate Query Language (HQL) that allows developers to express complex queries in a natural and intuitive way. Additionally, Hibernate supports Criteria API, which provides a type-safe alternative to HQL.

Hibernate can be a valuable tool for Java developers who are looking to simplify database access, reduce development time, and improve application performance..

.Audience

The hibernate training is a 4-days course designed for:

- Software Professionals
- Java Developers: Hibernate is designed to work seamlessly with Java, so Java developers who want to learn how to use Hibernate to simplify database access and improve application performance can benefit from this training.
- Database Developers: Database developers who want to learn how to use Hibernate to map relational data to object-oriented code can benefit from this training.
- Software Architects: Software architects who want to evaluate Hibernate as a potential ORM solution for their organization can benefit from this training.
- Technical Leads: Technical leads who want to ensure that their development teams are using Hibernate effectively and efficiently can benefit from this training.
- Development Managers: Development managers who want to understand how Hibernate can be used to improve development productivity and application performance can benefit from this training.
- Anyone who is involved in Java development and wants to learn how to use Hibernate to simplify database access and improve application performance can benefit from this training.Learning

Objectives:

- Understanding Object-Relational Mapping (ORM) concepts and how Hibernate simplifies the mapping of object-oriented code to relational databases.
- Learning the Hibernate architecture and how to configure it to work with different types of databases.
- Understanding Hibernate Session and Transaction management and how to use it to interact with databases.
- Learning how to map entities to database tables using Hibernate annotations or XML configuration.
- Understanding Hibernate Query Language (HQL) and Criteria API for querying databases using Hibernate.
- Learning how to manage associations and relationships between entities using Hibernate.
- Understanding Hibernate caching mechanisms and how to configure and use them to improve application performance.
- Learning how to implement inheritance mapping and table relationships in Hibernate.
- Understanding Hibernate transaction and concurrency management.
- Learning how to integrate Hibernate with other Java technologies such as Spring and Java Persistence API (JPA).

By the end of the training program, participants should be able to develop Java applications that interact with relational databases using the Hibernate framework.

Candidate Prerequisites

- Basic computer skills: Participants should be comfortable using a computer and have a basic understanding of computer programming concepts.
- Java Programming: A good understanding of Java programming concepts is essential, as Hibernate is a Java-based framework.
- Object-Oriented Programming (OOP) concepts: A good understanding of OOP concepts such as inheritance, encapsulation, abstraction, and polymorphism is required.
- SQL and Relational Database Concepts: A good understanding of SQL and relational database concepts such as tables, columns, relationships, and indexes is essential.
- JDBC: A basic understanding of JDBC (Java Database Connectivity) is required, as Hibernate builds on top of JDBC.
- XML and Annotation Configuration: A basic understanding of XML and Annotation configuration is required as Hibernate can be configured using either XML or Annotations.
- Basic Knowledge of Web Technologies: A basic understanding of web technologies such as HTML, CSS, and JavaScript is preferred, as Hibernate is often used in web applications.

That is, the participants should have a solid foundation in Java programming, object-oriented programming concepts, SQL, and relational database concepts, as well as a basic understanding of web technologies to attend a Hibernate training program successfully.

Infrastructure Prerequisites

Software:

Java Development Kit (JDK): Participants should have the latest version of JDK installed on their machines.
Integrated Development Environment (IDE): Participants should have an IDE installed on their machines, such as Eclipse, IntelliJ IDEA, or NetBeans.
Hibernate Framework: Participants should download and install the Hibernate framework on their machines. The latest version of Hibernate can be downloaded from the official Hibernate website.
Database: Participants should have access to a relational database such as MySQL, Oracle, or PostgreSQL.

Hardware:

Computer or Laptop: Participants should have a computer or laptop with sufficient memory and processing power to run the IDE and database.

Cloud Infrastructure:

Cloud Services: Participants can use cloud services such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP) to set up virtual machines and databases to practice Hibernate. These cloud services provide the necessary infrastructure for running the training programs, and they can be accessed from anywhere with an internet connection.

Training Outline:

Note: The duration and the contents are approximate. The trainer may change the approach and the duration, based on the response of the participants.

Day 1

Session 1:

- Introduction to Hibernate
- Understanding Object-Relational Mapping (ORM)
- Hibernate Architecture
- Configuring Hibernate

Session 2:

- Hibernate Session and Transaction Management
- Entity Mapping
- Mapping with Annotations and XML
- Primary Keys and Generated Values

Hands-On Activities:

- Setting up the Hibernate environment
- Creating the first Hibernate application
- Mapping entity classes to database tables
- Performing CRUD (Create, Read, Update, Delete) operations using Hibernate

Day 2:

Session 3:

- Hibernate Query Language (HQL)
- Criteria API
- Query by Example

Session 4:

- Associations and Relationships
- One-to-One and One-to-Many Relationships
- Many-to-Many Relationships
- Bidirectional Relationships

Hands-On Activities:

- Querying data using HQL and Criteria API
- Implementing associations and relationships in Hibernate
- Working with cascading and orphan removal

Day 3:

Session 5:

- Inheritance Mapping
- Table per Class Hierarchy
- Table per Subclass
- Table per Concrete Class

Session 6:

- Hibernate Caching
- First Level Cache
- Second Level Cache
- Query Cache

Hands-On Activities:

- Implementing inheritance mapping in Hibernate
- Configuring and using Hibernate caching

Day 4:

Session 7:

- Transactions and Concurrency
- Isolation Levels
- Optimistic and Pessimistic Locking

Session 8:

- Integration with Spring Framework
- Integration with Java Persistence API (JPA)

Hands-On Activities:

- Managing transactions and concurrency in Hibernate
- Integrating Hibernate with Spring and JPA

OPTIONAL PROJECT

Note: Additional Days Will Be Needed
Depending on the organization, the project may change

Project: Not be a separate project

The project can be designed to cover different aspects of Hibernate, such as entity mapping, associations, querying, transaction management, and caching. It can be a web application or a standalone application, depending on the focus of the training program.

The project can be developed as a group project, with participants working in teams to apply the concepts learned in the training program. Alternatively, it can be developed as an individual project, where each participant works on their own to implement the project requirements.