Hibernate

Wt is hibernate

Benefits

Code snippet

Hibernate

A frame work for persisting/ saving java obj in DB

Saving and retriivng from DB

Benfits

1 handle all low level sql

2 minize JDBC code

3 provide ORM ( store and retrive obj from DB)

ORM

Developer defines mapping betwwen java class and DB table

Mapping need to be done through configuration either thorugh XML or java annotation

Realtion Between Hibernate and JDBC

- hibernate user JDBC for all DB communications

- hibernate is an another extration layer on top of jdbc

- using hibernate API

- hibernate does all low level JDBC work in background but in background all goes through standard JDBC API

Environment Setup

Hibernate 5.2 requires Java 8

Steps of Envionement

1 Java JDK

2 IDE

3 database server

4 hibernate jar and JDBC driver

1 Install MYsql / postgress /oracle

2 setup up DB tables -> create user

CREATE USER 'hbstudent'@'localhost' IDENTIFIED BY 'hbstudent';

GRANT ALL PRIVILEGES ON \* . \* TO 'hbstudent'@'localhost';

CREATE DATABASE IF NOT EXISTS `hb\_student\_tracker`;

USE `hb\_student\_tracker`;

CREATE TABLE student (

id serial PRIMARY KEY,

first\_name varchar(45) DEFAULT NULL,

last\_name varchar(45) DEFAULT NULL,

email varchar(45) DEFAULT NULL

)

Serial for auto increment

Setup hibernate in Eclipse

1 create eclipse project

-new Java Project -HibernateLearning -- new folder lib add all hibernate n jdbc jars

2 download hibernate files

-https://hibernate.org/ download Hibernate ORM 5.4.9

- extract zip under lib copy all jars in reuired folder to project lib basci requird jars

3 download driver as per DB

4 add jar files build path

Write JDBC code to chcek wokring or not

**public** **class** TestJdbc {

**public** **static** **void** main(String[] args) {

String jdbcUrl="jdbc:postgresql://localhost:5432/personal";

//String jdbcUrl="jdbc:postgresql://localhost:5432/personal?useSSL=false"; // to bypass SSL warings

String userName="postgres";

String passWord="sachin";

**try**

{

System.***err***.println("Connecting Data base: "+ jdbcUrl);

Connection con=DriverManager.*getConnection*(jdbcUrl, userName, passWord);

System.***err***.println("Connection Successful");

}

**catch** (Exception e) {

System.***err***.println(e.getMessage());

}

}

}

Hibernate Dev process

1 add Hibernate Config file

- basically tell hibernate how to connect to the data base

-hibernate jdbc in back end to communicate

JDBCurl,userid,pwd etc

- create / paste hibernate.cfg.xml file in src bcz its actual class path off application

2 Annotate java Class

3 develop java code perform DB operations