1. Create a pom project using following command:

C:>mvn archetype:generate -DgroupId=com.hibernate5.example -DartifactId=Hibernate5Example -Dversion=1.0 -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

2. pom.xml

|  |
| --- |
| <project xmlns=*"http://maven.apache.org/POM/4.0.0"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4\_0\_0.xsd"*>  <modelVersion>4.0.0</modelVersion>  <groupId>hibernate.example</groupId>  <artifactId>HibernateExample5</artifactId>  <packaging>jar</packaging>  <version>1.0</version>  <name>HibernateExample5</name>  <url>http://maven.apache.org</url>  <dependencies>  <dependency>  <groupId>junit</groupId>  <artifactId>junit</artifactId>  <version>3.8.1</version>  <scope>test</scope>  </dependency>    <dependency>  <groupId>org.hibernate</groupId>  <artifactId>hibernate-core</artifactId>  <version>5.1.0.Final</version>  </dependency>  </dependencies>  <build>  <plugins>  <plugin>  <groupId>org.maven.plugins</groupId>  <artifactId>maven-compiler-plugin</artifactId>  <version>2.3.1</version>  <configuration>  <source>1.8</source>  <target>1.8</target>  </configuration>  </plugin>  </plugins>  </build>  </project> |

3. hibernate.cfg.xml

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"utf-8"*?>  <!DOCTYPE hibernate-configuration SYSTEM "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">  <hibernate-configuration>  <session-factory>  <property name=*"hibernate.dialect"*> org.hibernate.dialect.H2Dialect  </property>  <property name=*"hibernate.connection.driver\_class"*> org.h2.Driver  </property>  <!-- url to create db in user home directory -->  <property name=*"hibernate.connection.url"*> jdbc:h2:~/test  ;MV\_STORE=FALSE</property>  <property name=*"hibernate.connection.username"*> test </property>  <property name=*"hibernate.connection.password"*> test </property>  <property name=*"hibernate.show\_sql"*>true</property>  <property name=*"hibernate.hbm2ddl.auto"*>update</property>    <mapping class=*"com.hibernate.example.Employee"*/>  </session-factory>  </hibernate-configuration> |

4. Employee.java

|  |
| --- |
| **package** com.hibernate.example;  **import** javax.persistence.Column;  **import** javax.persistence.Entity;  **import** javax.persistence.Id;  **import** javax.persistence.Table;  @Table  @Entity(name = "EMPLOYEE")  **public** **class** Employee {  **private** Long id;  **private** String name;  **private** String designation;  @Id  @Column(name = "ID")  **public** Long getId() {  **return** id;  }  **public** **void** setId(Long id) {  **this**.id = id;  }  @Column(name = "NAME")  **public** String getName() {  **return** name;  }  **public** **void** setName(String name) {  **this**.name = name;  }  @Column(name = "DESIGNATION")  **public** String getDesignation() {  **return** designation;  }  **public** **void** setDesignation(String designation) {  **this**.designation = designation;  }  } |

5. Test class

|  |
| --- |
| **package** com.hibernate.example;  **import** org.hibernate.Session;  **import** org.hibernate.SessionFactory;  **import** org.hibernate.cfg.Configuration;  **public** **class** HibernateExample {  **public** **static** **void** main(String[] args) {  **new** HibernateExample().getEmployees();  }  **public** **void** insert() {  Employee employee = **new** Employee();  employee.setId(1L);  employee.setName("john");  employee.setDesignation("SSE");  Configuration configuration = **new** Configuration().configure("h2.hibernate.cfg.xml");  SessionFactory sessionFactory = configuration.buildSessionFactory();  Session session = sessionFactory.openSession();  session.beginTransaction();  session.save(employee);  session.getTransaction().commit();  session.close();  }  **public** **void** getEmployees() {  Configuration configuration = **new** Configuration().configure("h2.hibernate.cfg.xml");  SessionFactory sessionFactory = configuration.buildSessionFactory();  Session session = sessionFactory.openSession();  Employee employee = (Employee) session.get(Employee.**class**, 1L);  System.***out***.println(employee);  session.close();  }  } |

6. project structure

