* How to start with Hibernate ?
* How Hibernate is useful ? or **Why Hibernate** ?
* Hibernate is an ORM tool.
* ORM means Object Relational Mapping i.e. Objects of a class is mapped to the Table in Database.
* A class acts as a Table,
* Class member variables acts as Columns of the table.
* So each and every variable in the class has specific column in the table.
* But its not necessary to have equal numbers of variables and columns.
* We can omit some variables in the class.
* An object of that class acts as a row in the table.
* **So we don’t need to write SQL queries to manipulate database.**
* **Hibernate will provide methods to do the same.**
* Just we need to do the exact mapping or in other words, we need to tell the hibernate that this class should represent a table in database.
* Mapping is done using an xml file, known as hibernate.cfg.xml file.

**Creating our first Project of Hibernate**

1. Create a Maven project

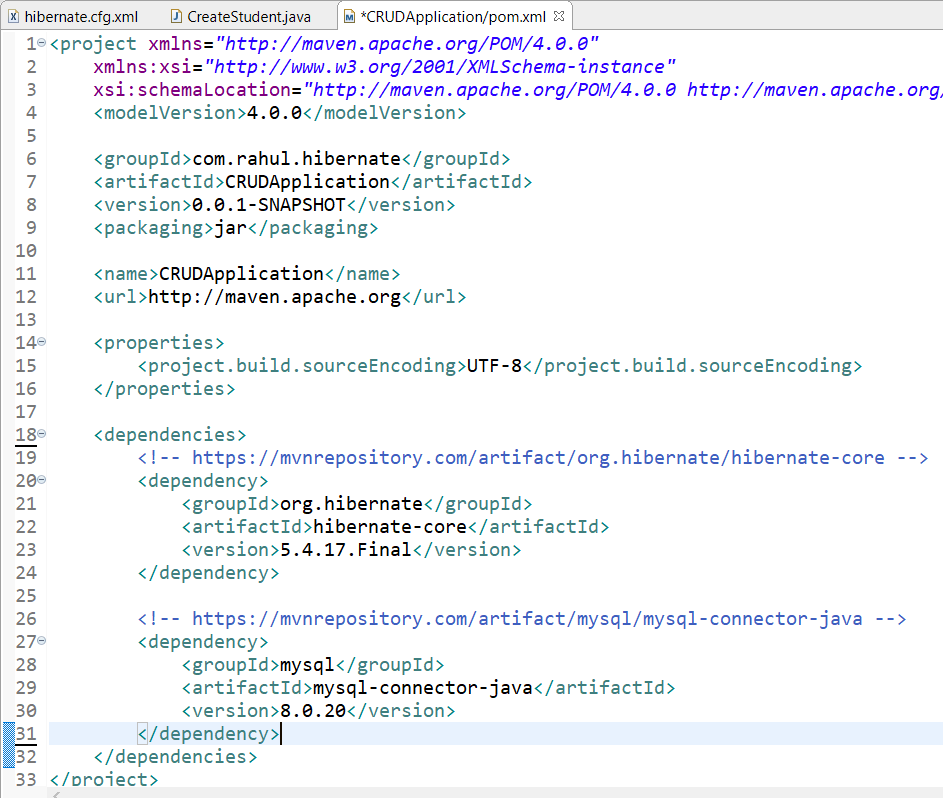
Internal, quickstart, version 1.1

1. Adding Maven Dependency in pom.xml
2. Hibernate ORM dependency

🡪 Hibernate Core Relation Version 5.4.17

2) MySql Connector

🡪 mysql connector/J Version 8.0.20

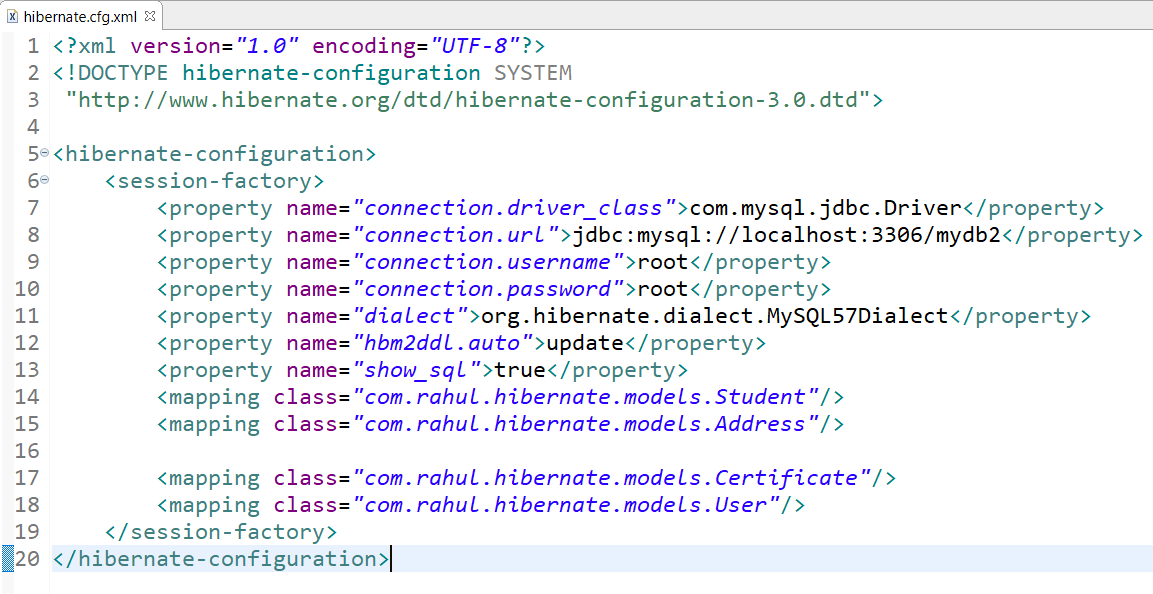


1. Create hibernate.cfg.xml file.

We can create it anywhere in the project, just need to give the path while using it.

For now or simplicity create it in the src/main/java folder

1. Configure the hibernate.cfg.xml file for mapping.



1. To get suggestion while typing, add these lines

<!DOCTYPE hibernate-configuration SYSTEM

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

1. CRUD
2. Create

Need Model or Entity or Object and some mapping like

telling this Class will be the table, specifying the name of the Table, Columns, Making Primary Key etc.

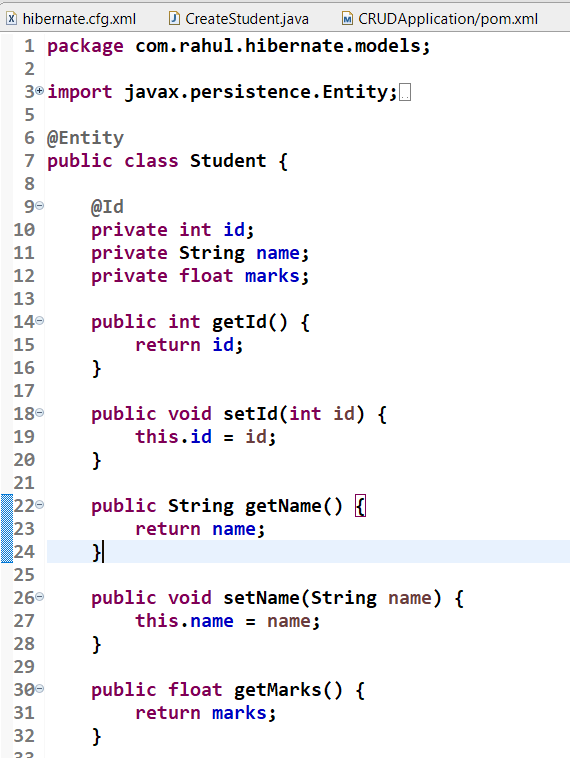
These all can be done using Annotations

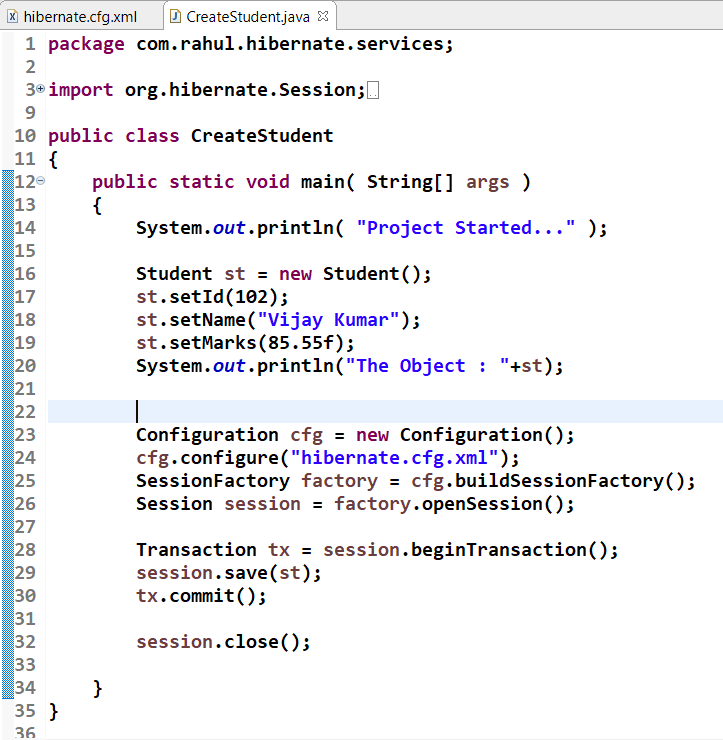
Eg,

@Entity

@Id

Etc.





1. Read

