Hibernate Tool or

Hibernate Reverse Engineering Tool

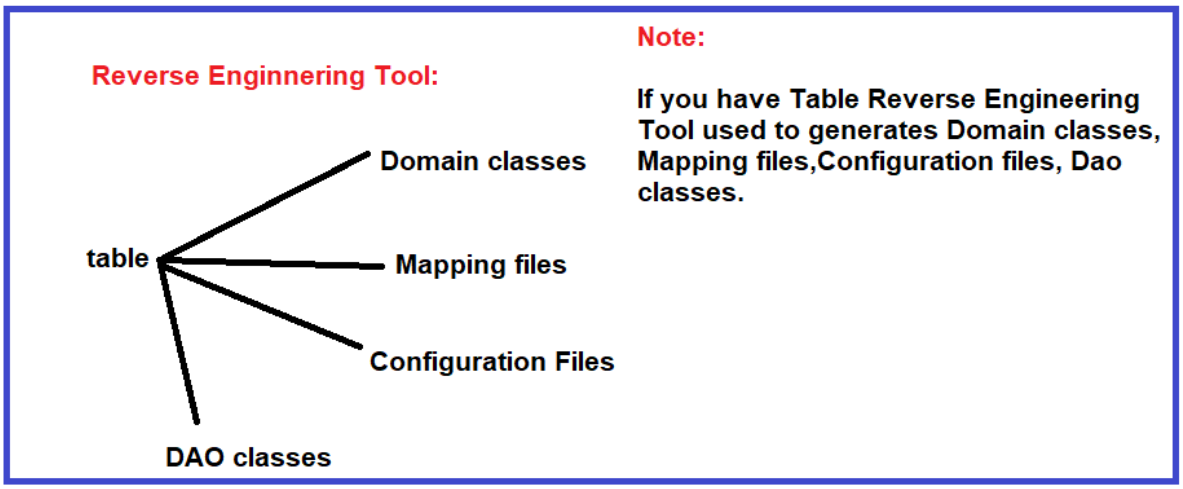
➔ **Reverse Engineering:**

Any thing opposite of Regular activity is called as Reverse Engineering Tool.

➔ **Hibernate Reverse Engineering Tool:**

If you have a table this reverse Engineering tool will generate Domain classes, Mapping files, Configuration files if needed DAO classes automatically and dynamically this is called as Hibernate Reverse Engineering Tool.

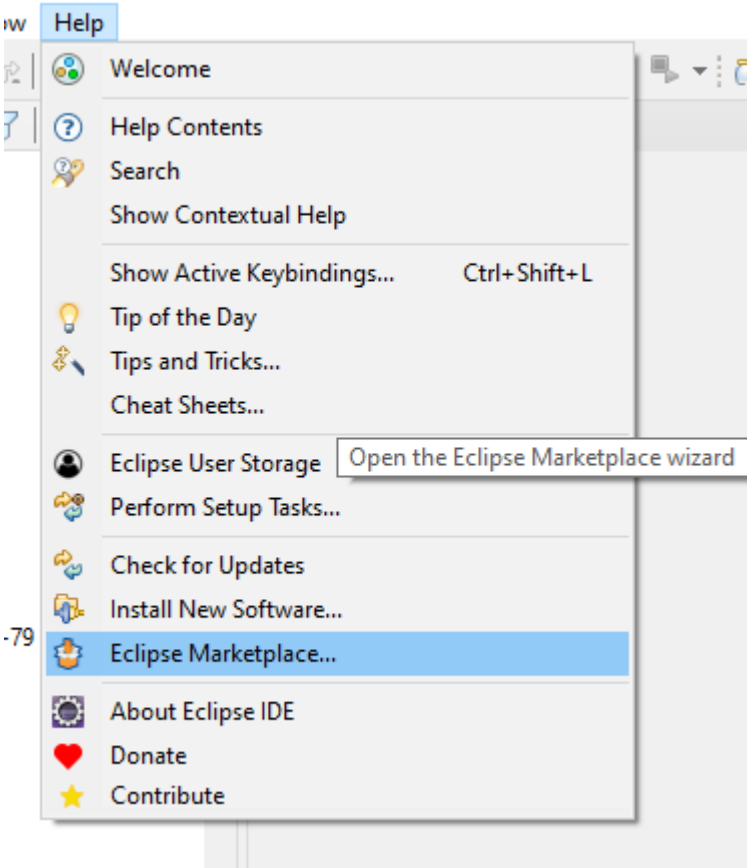
Note: If you are using annotations-based Domain classes then no need of mapping files.



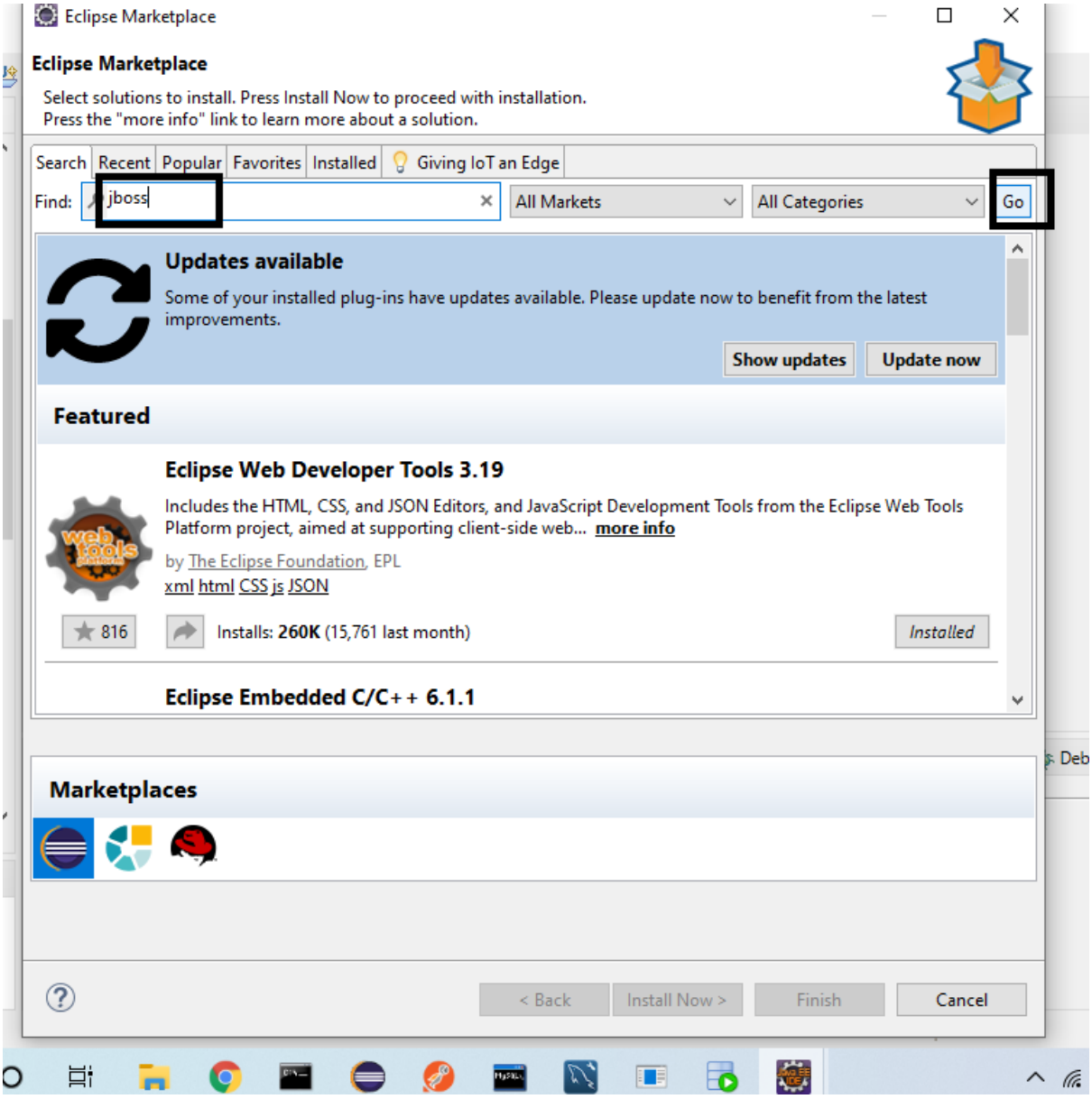
Adding JBOSS(Hibernate tool) plugin in Eclipse:

Step1:

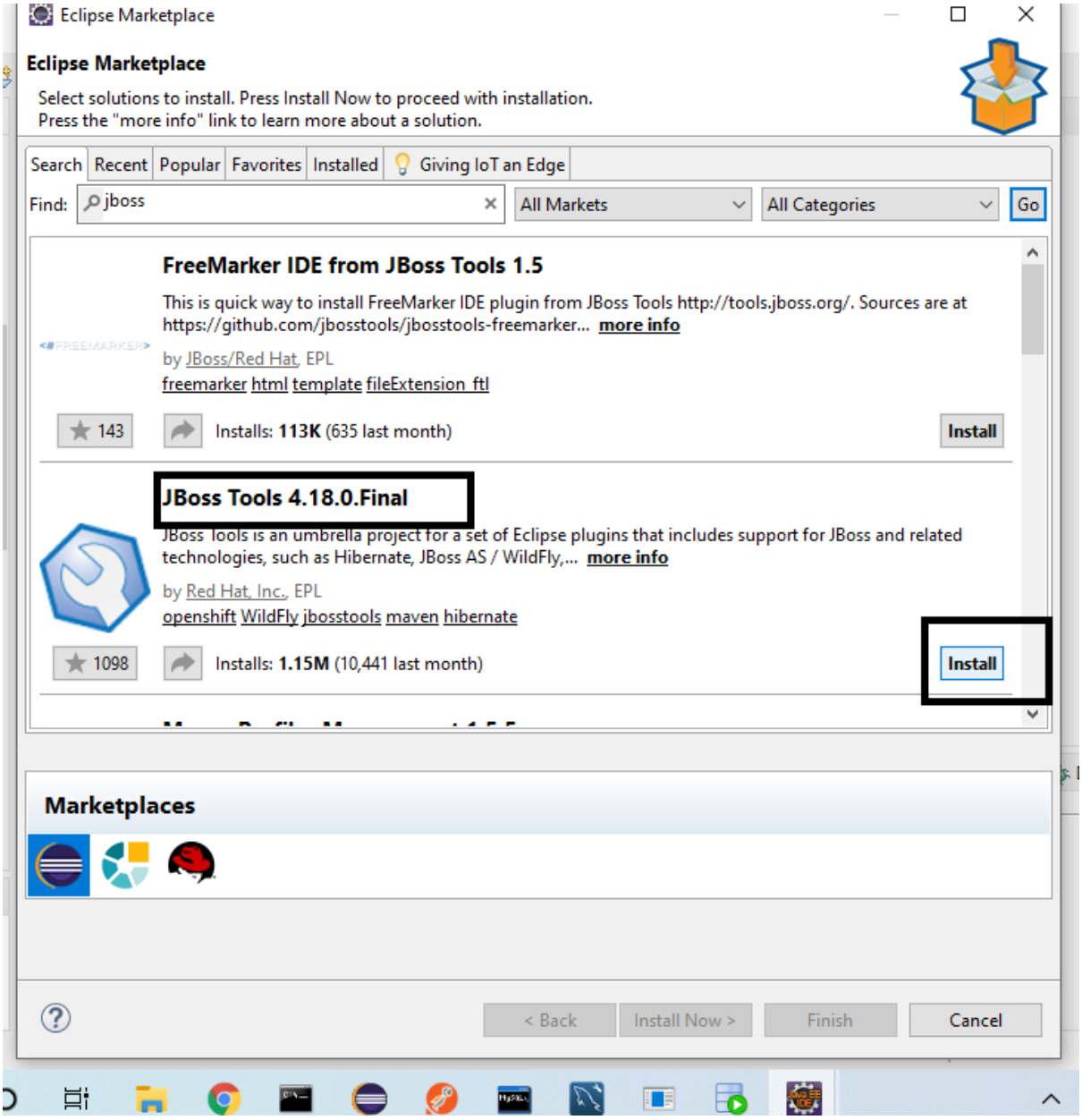
Help->Eclipse Marketplace



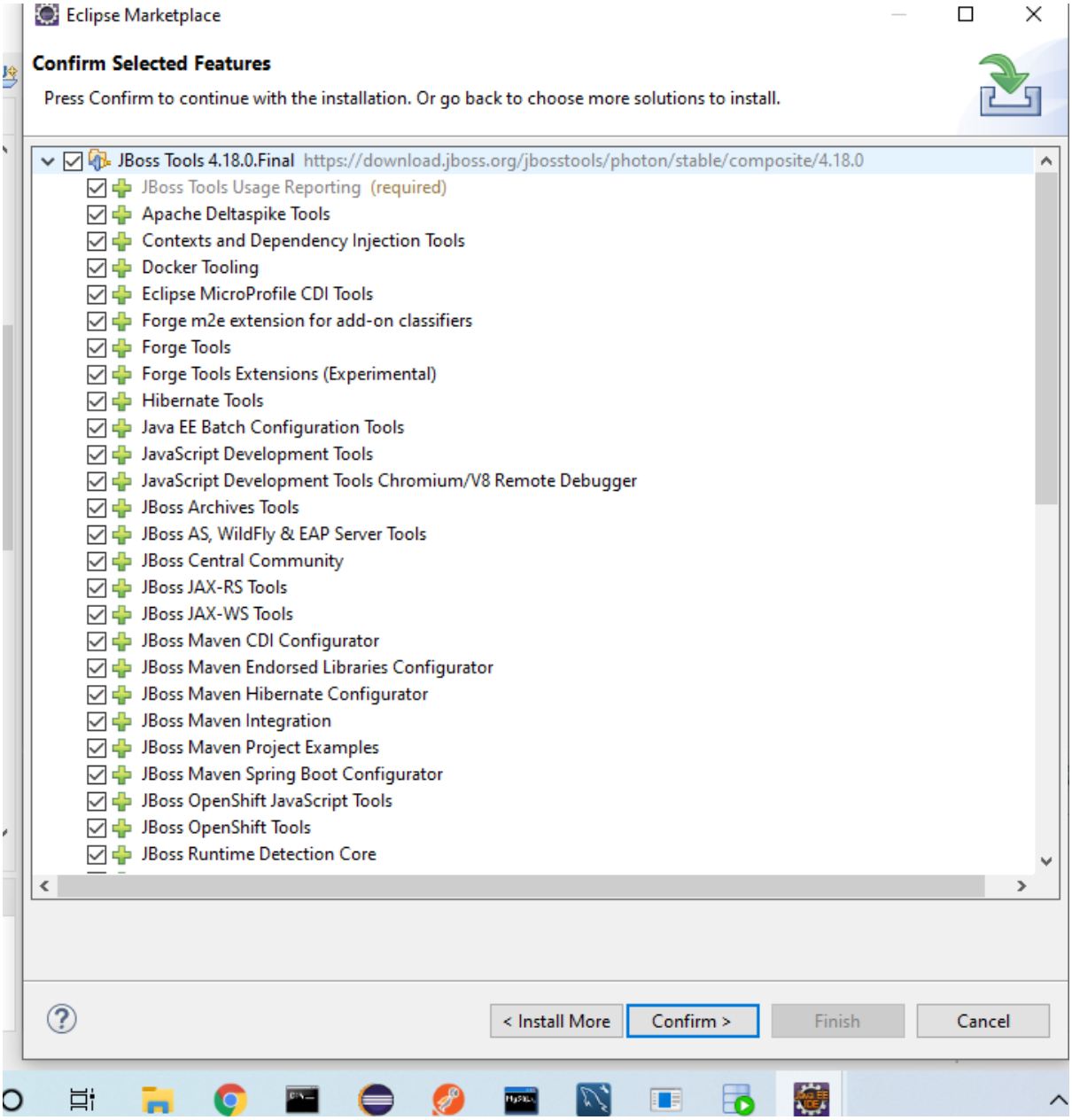
Step2:
Find text box there type-> jboss ->go



Step3:
Jboss tools->install

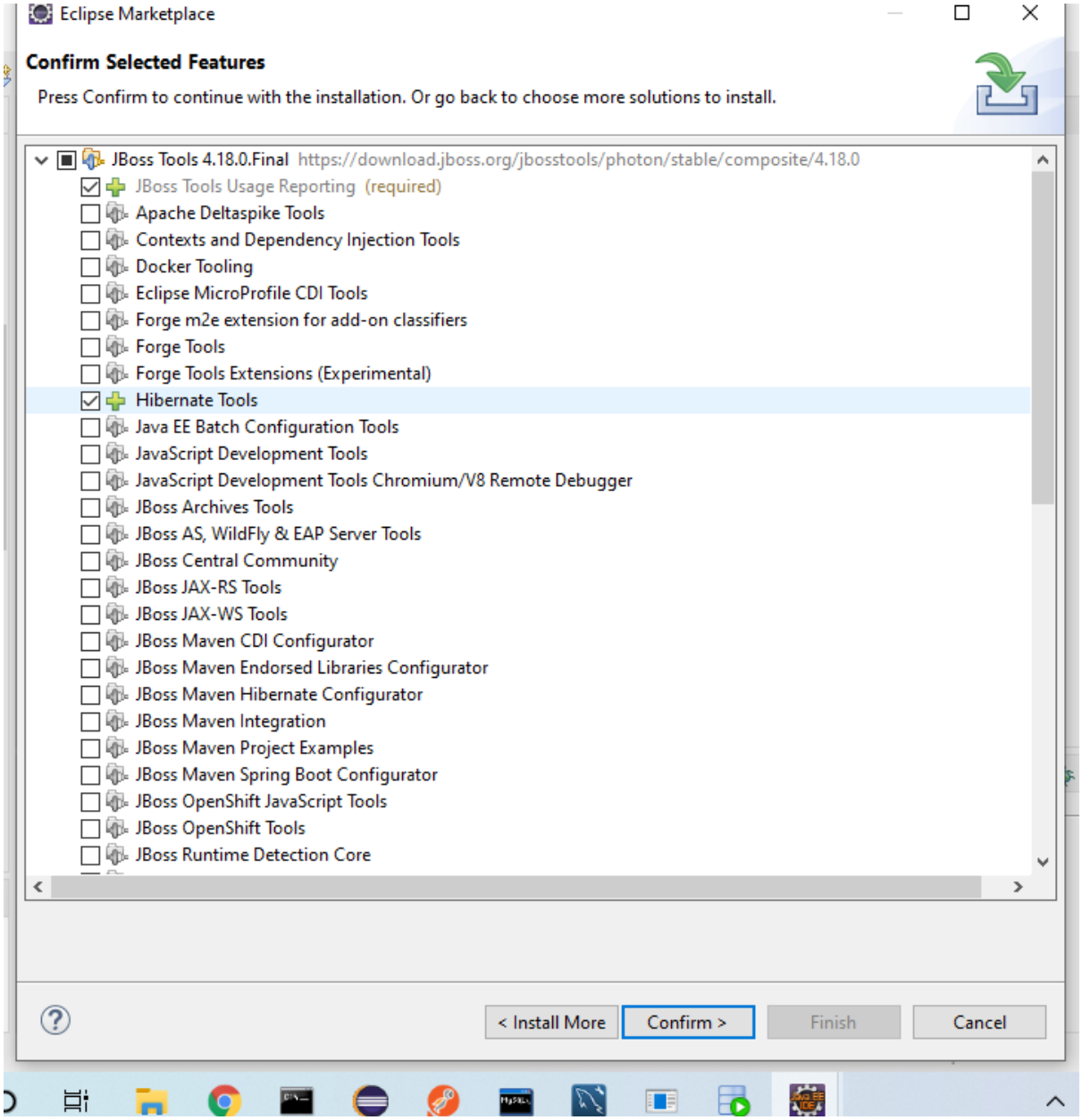


Step4:
De-select All



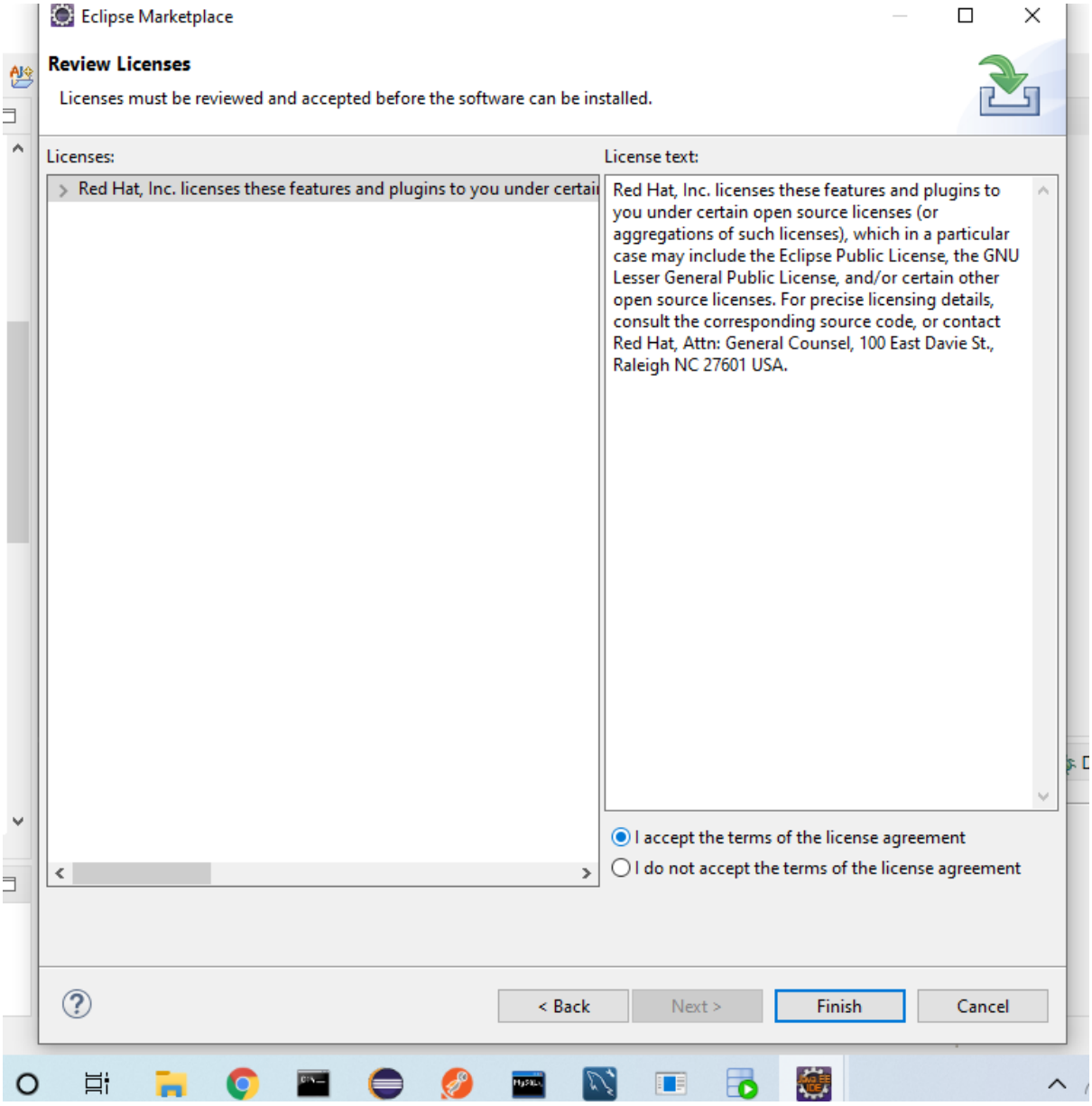
Step5:

Select Hibernate Tool and Confirm



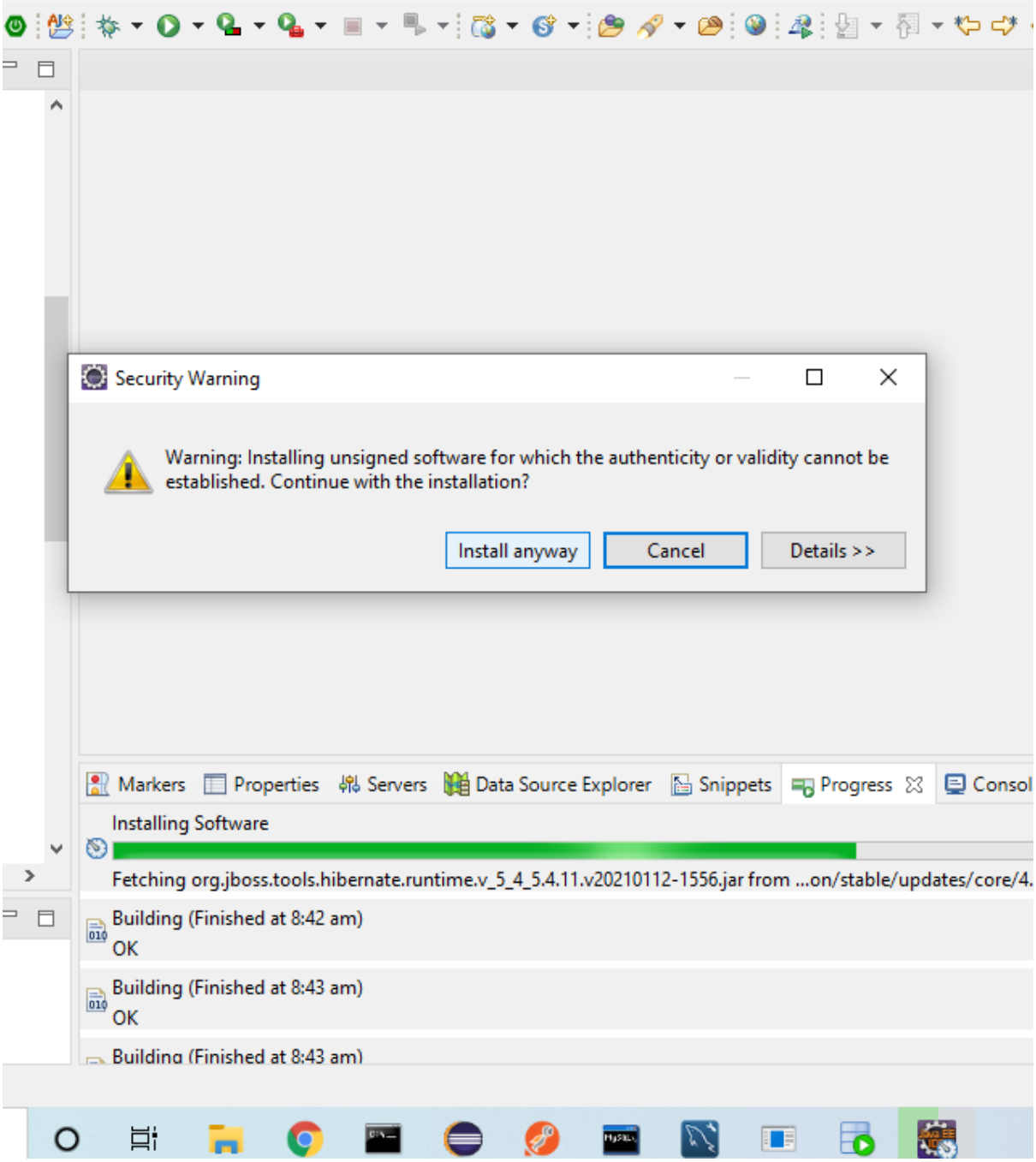
Step6:

Accept Terms and Conditions Finish



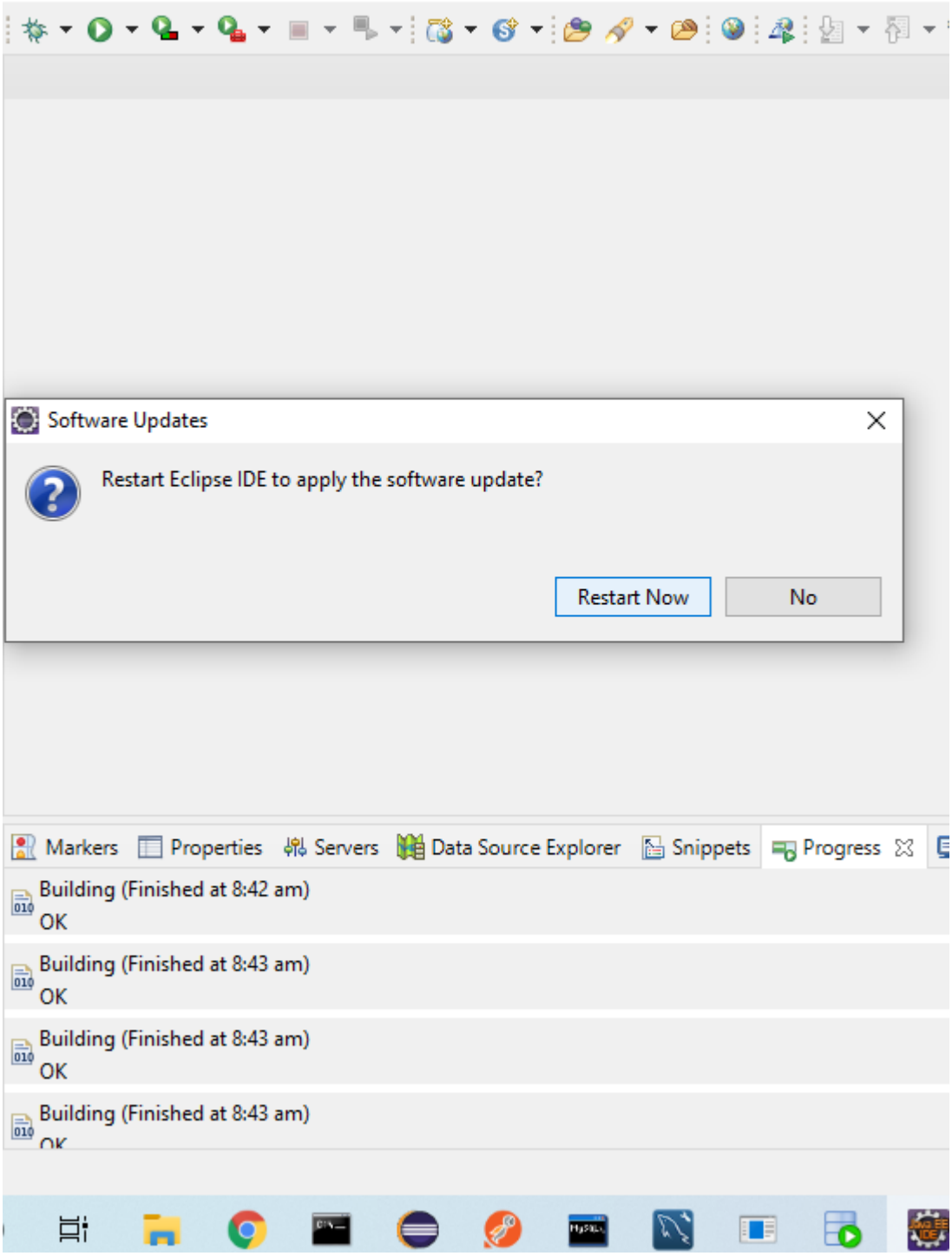
Step7:

Install anyway



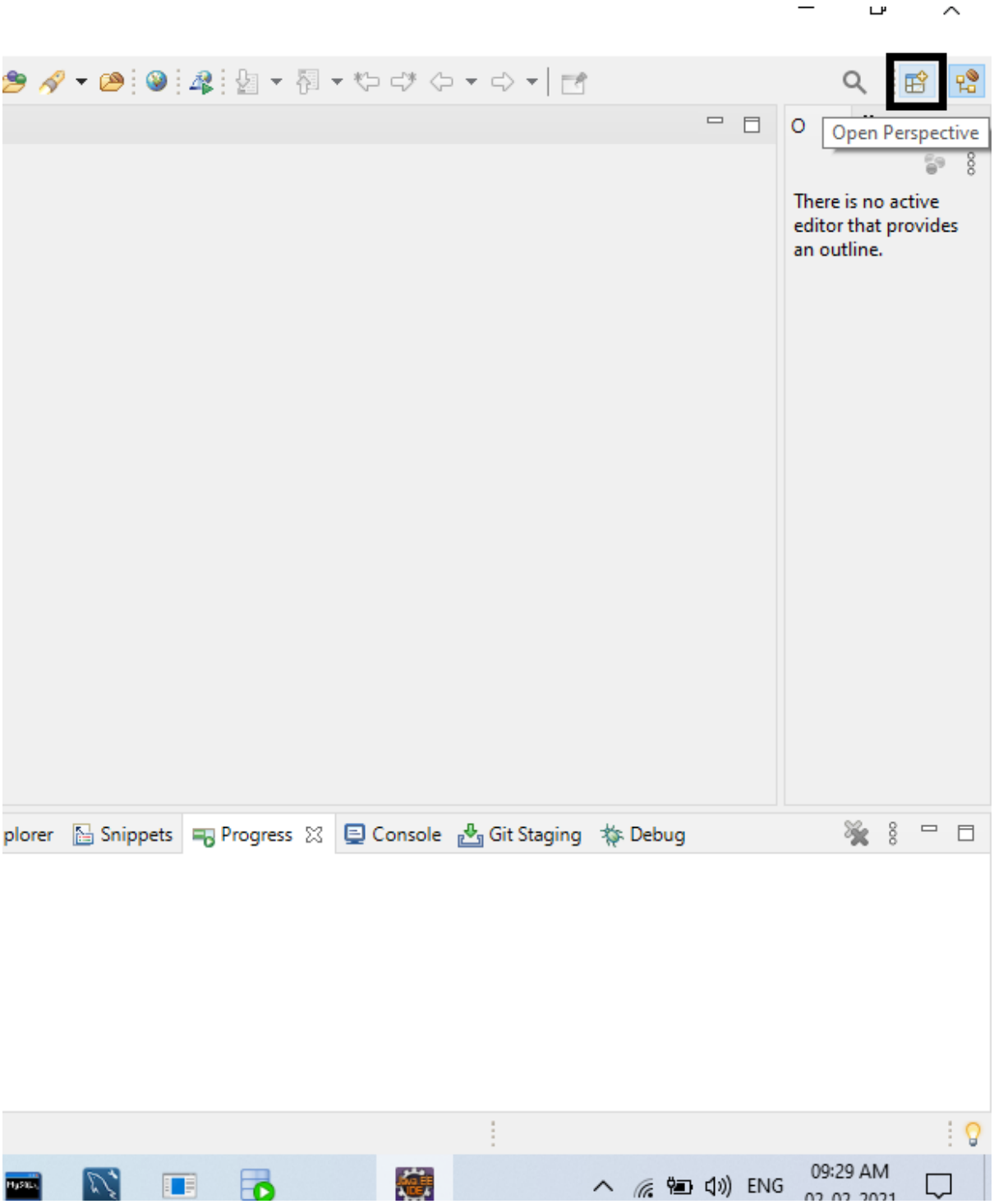
Step8:

Restart now



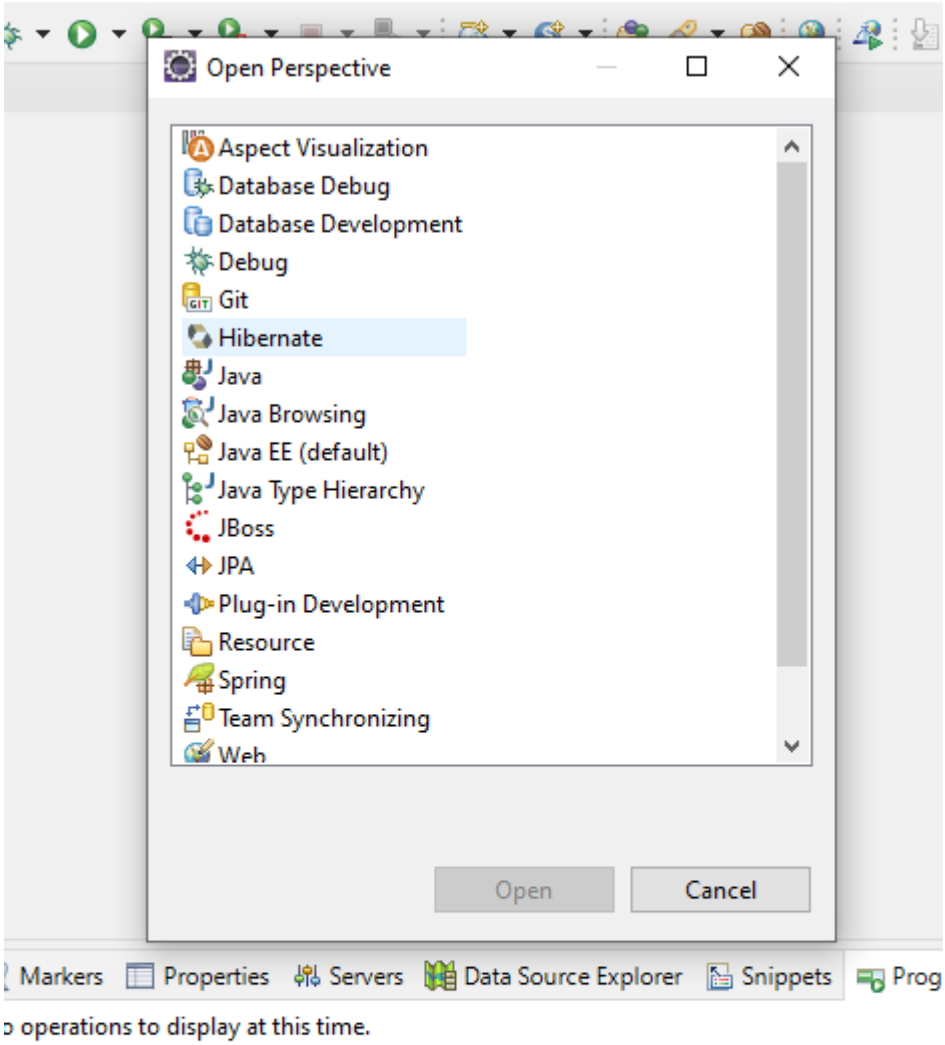
Step9:

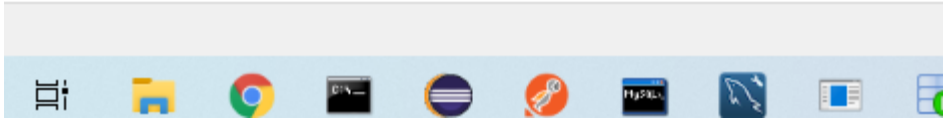
Open Perspective



Step10:

Confirm successfully installed or not hibernate tool
See hibernate option in perspective



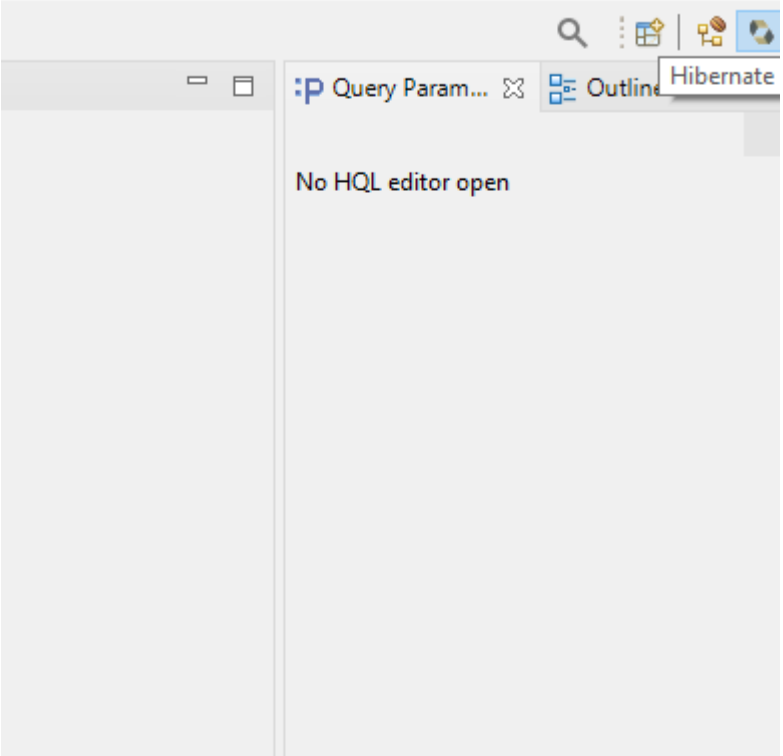


plugin adding completed

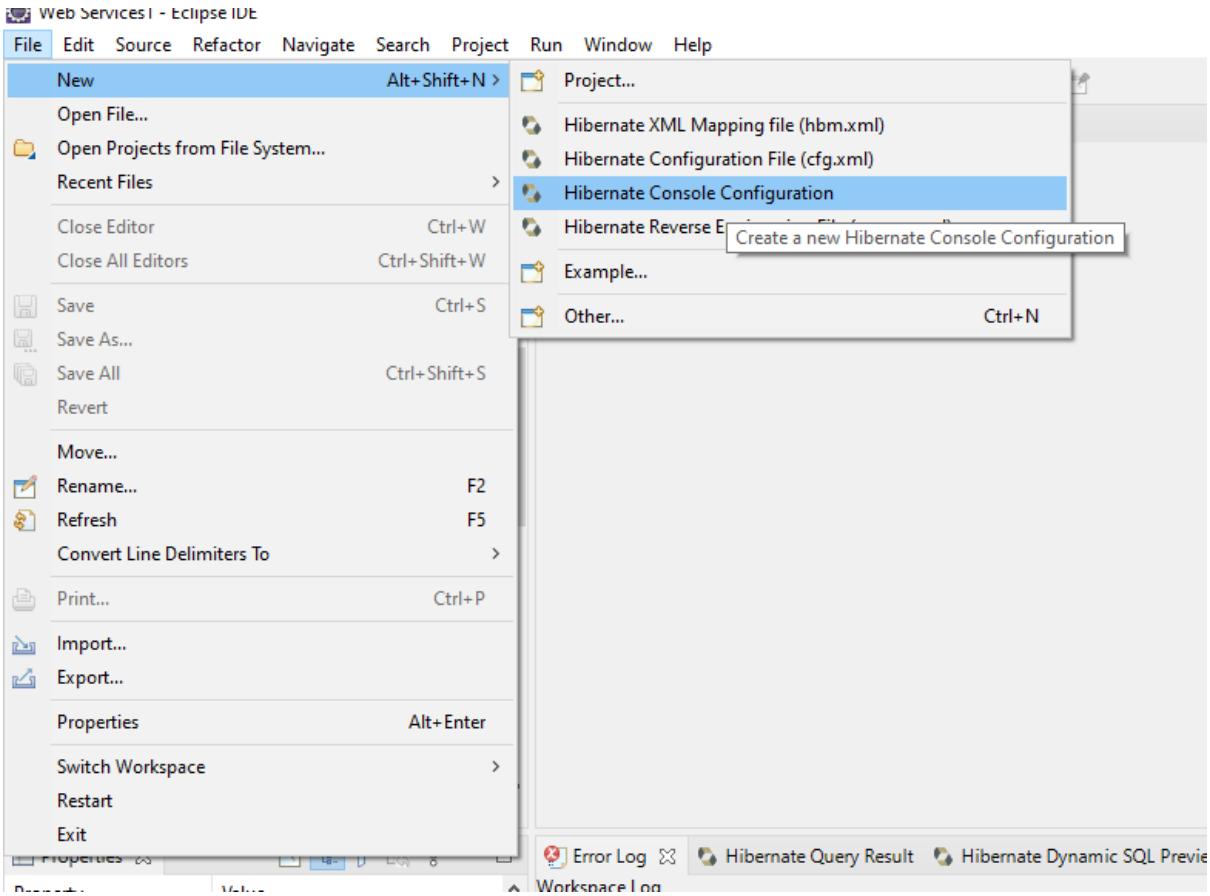
Creating Configuration File :

Note: if you are Created one time Configuration file you can use in multiple projects.

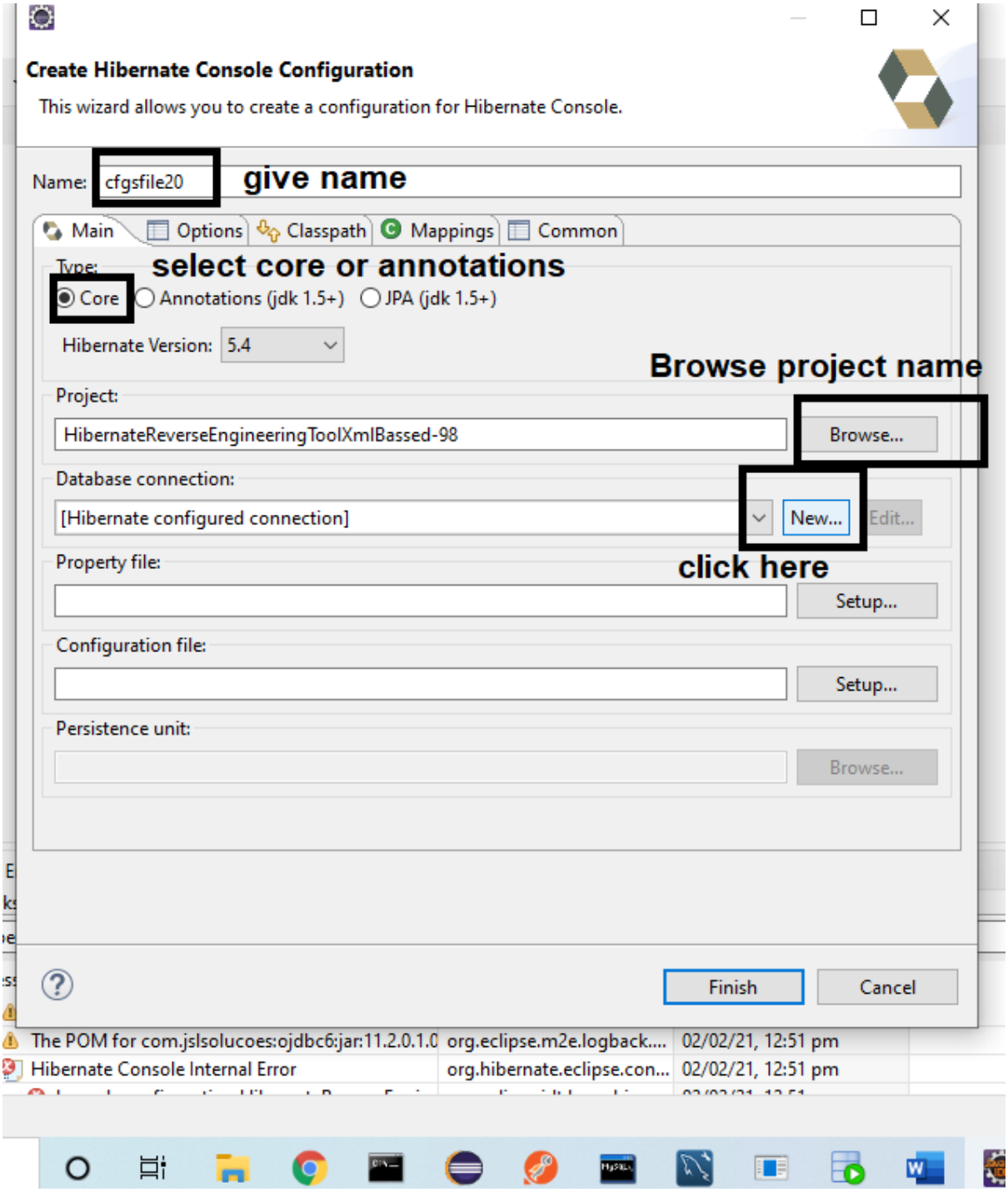
Step1: change perspective to Hibernate



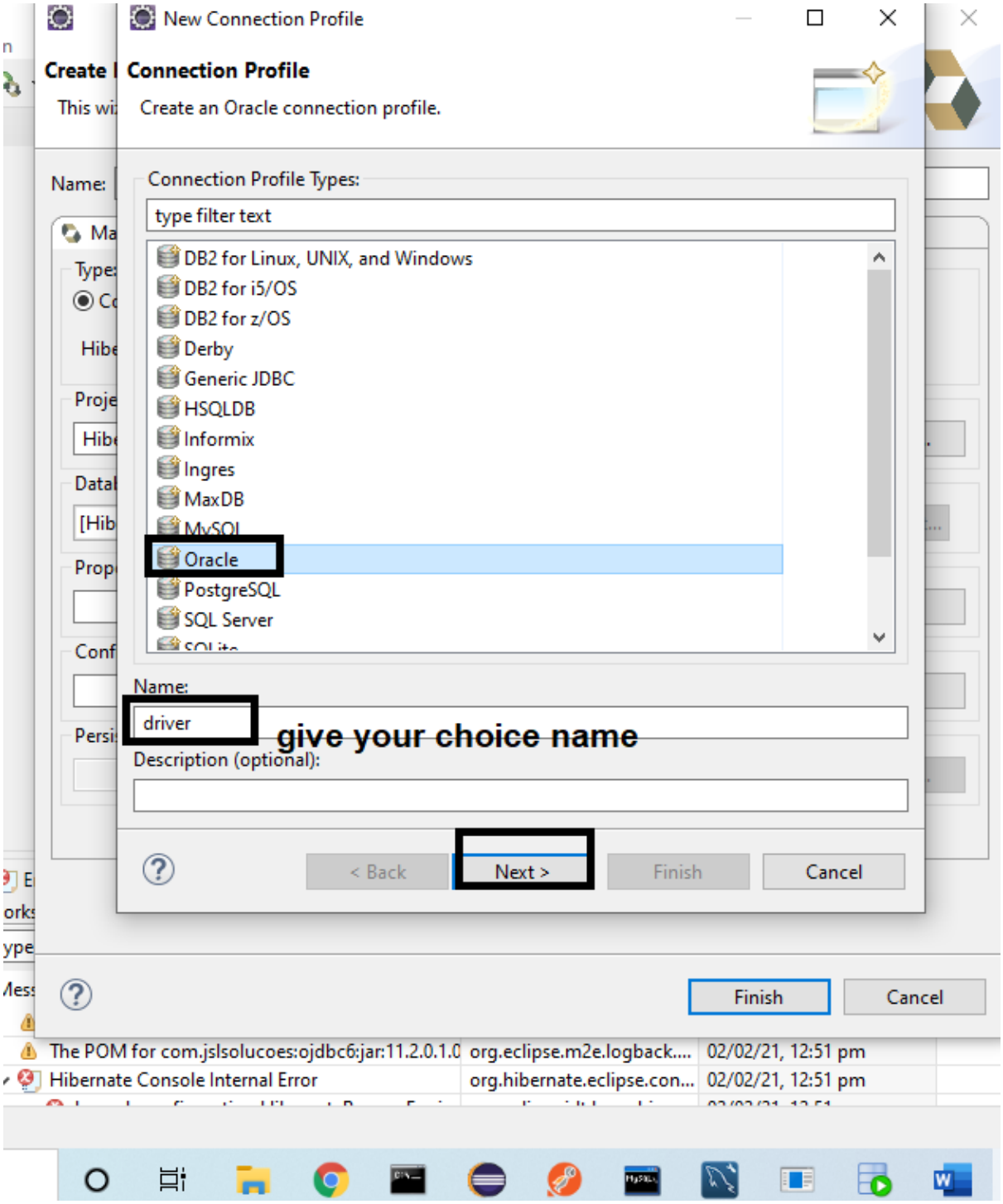
Step2:
Go to file……


Web Services1 - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
New Alt+Shift+N
Open File...
Open Projects from File System...
Recent Files
Close Editor Ctrl+W
Close All Editors Ctrl+Shift+W
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename... F2
Refresh F5
Convert Line Delimiters To
Print... Ctrl+P
Import...
Export...
Properties Alt+Enter
Switch Workspace
Restart
Exit
Project...
Hibernate XML Mapping file (hbm.xml)
Hibernate Configuration File (cfg.xml)
Hibernate Console Configuration
Hibernate Reverse E
Create a new Hibernate Console Configuration
Example...
Other... Ctrl+N
Error Log
Hibernate Query Result
Hibernate Dynamic SQL Previe
Workspace Log

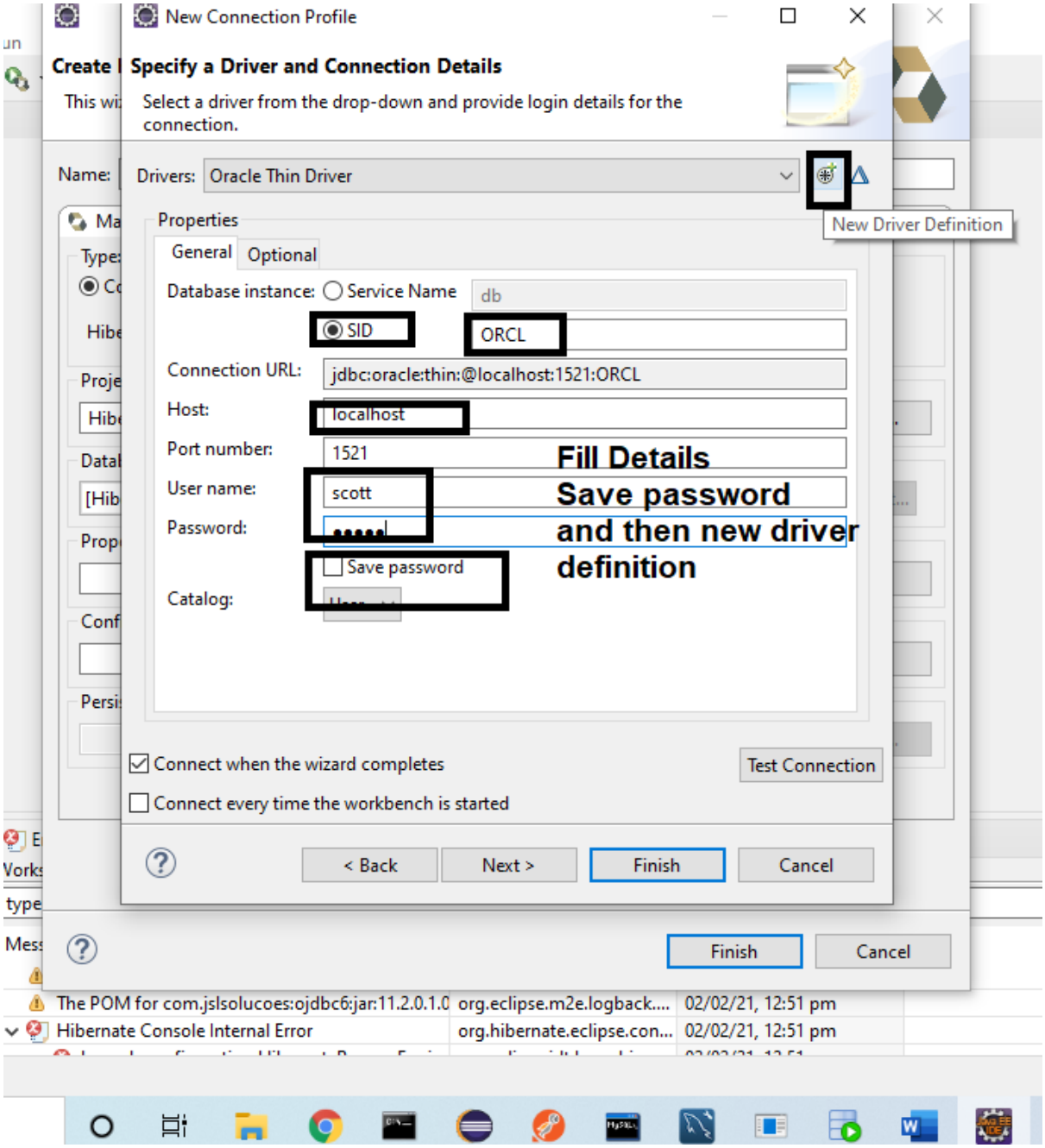
Step3:


Create Hibernate Console Configuration
This wizard allows you to create a configuration for Hibernate Console.
Name: cfgsfile20
give name
Main
Options
Classpath
Mappings
Common
Type:
select core or annotations
Core
Annotations (jdk 1.5+)
JPA (jdk 1.5+)
Hibernate Version: 5.4
Browse project name
Project:
HibernateReverseEngineeringToolXmlBassed-98
Browse...
Database connection:
[Hibernate configured connection]
New...
Edit...
click here
Property file:
Setup...
Configuration file:
Setup...
Persistence unit:
Browse...
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

Step4:

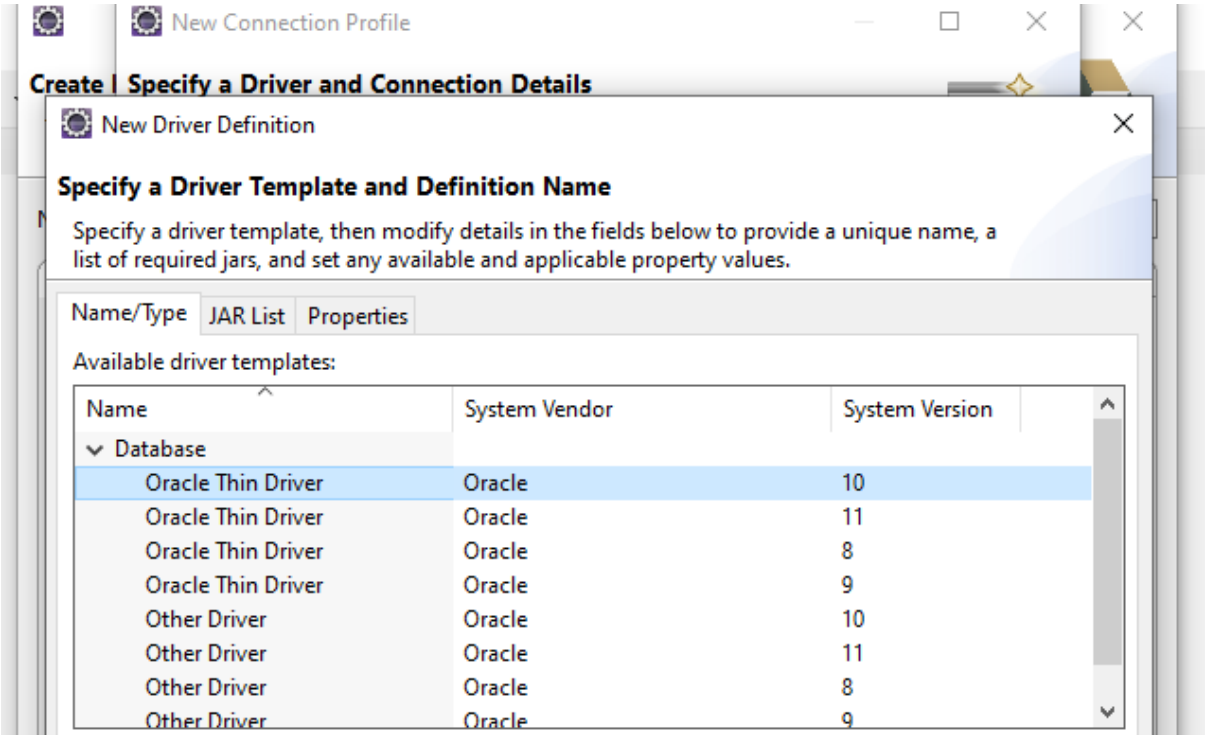

New Connection Profile
Connection Profile
Create an Oracle connection profile.
Connection Profile Types:
type filter text
DB2 for Linux, UNIX, and Windows
DB2 for i5/OS
DB2 for z/OS
Derby
Generic JDBC
HSQLDB
Informix
Ingres
MaxDB
MySQL
Oracle
PostgreSQL
SQL Server
Name:
driver
give your choice name
Description (optional):
< Back
Next >
Finish
Cancel
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

Step5:



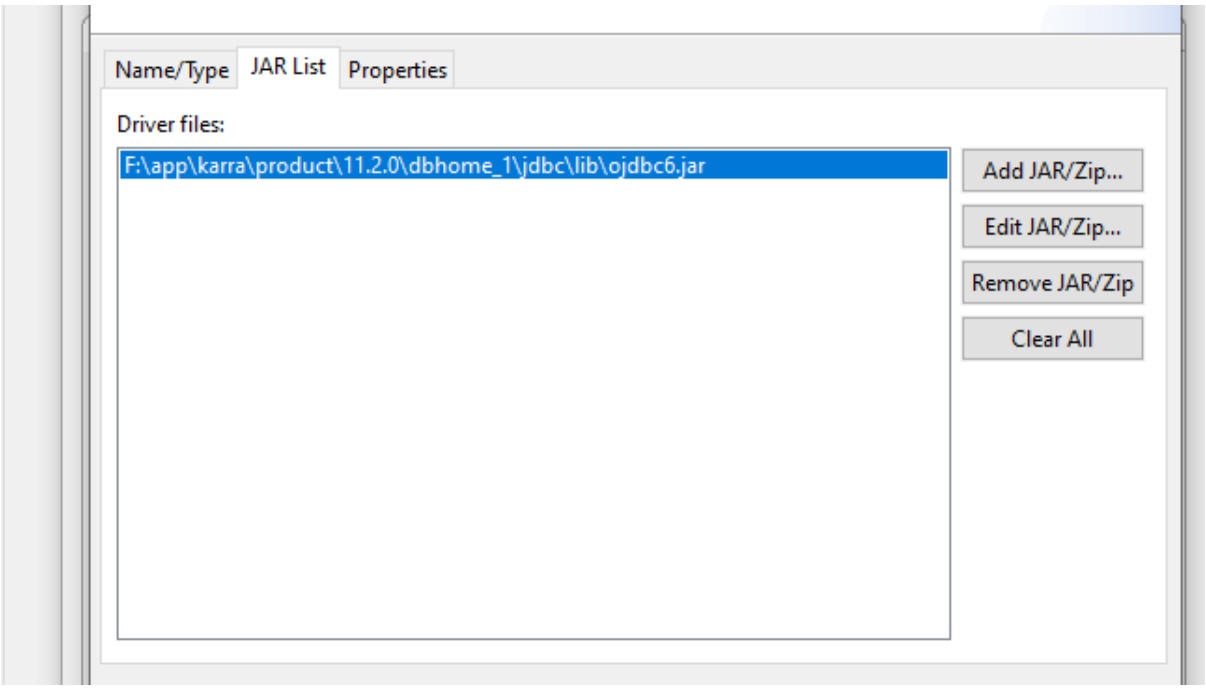
Step6:

Select type of Driver press ok

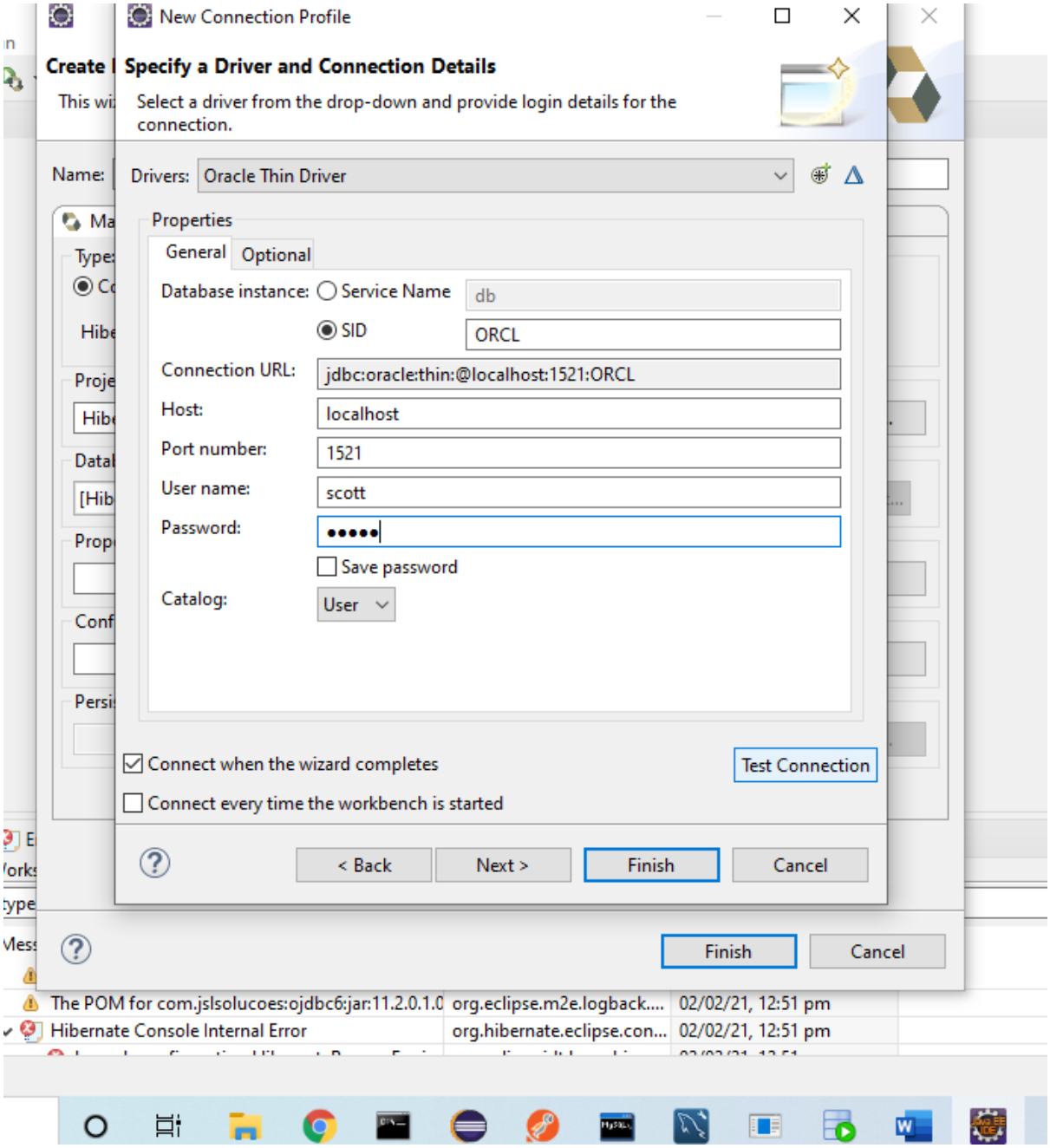


Step7:

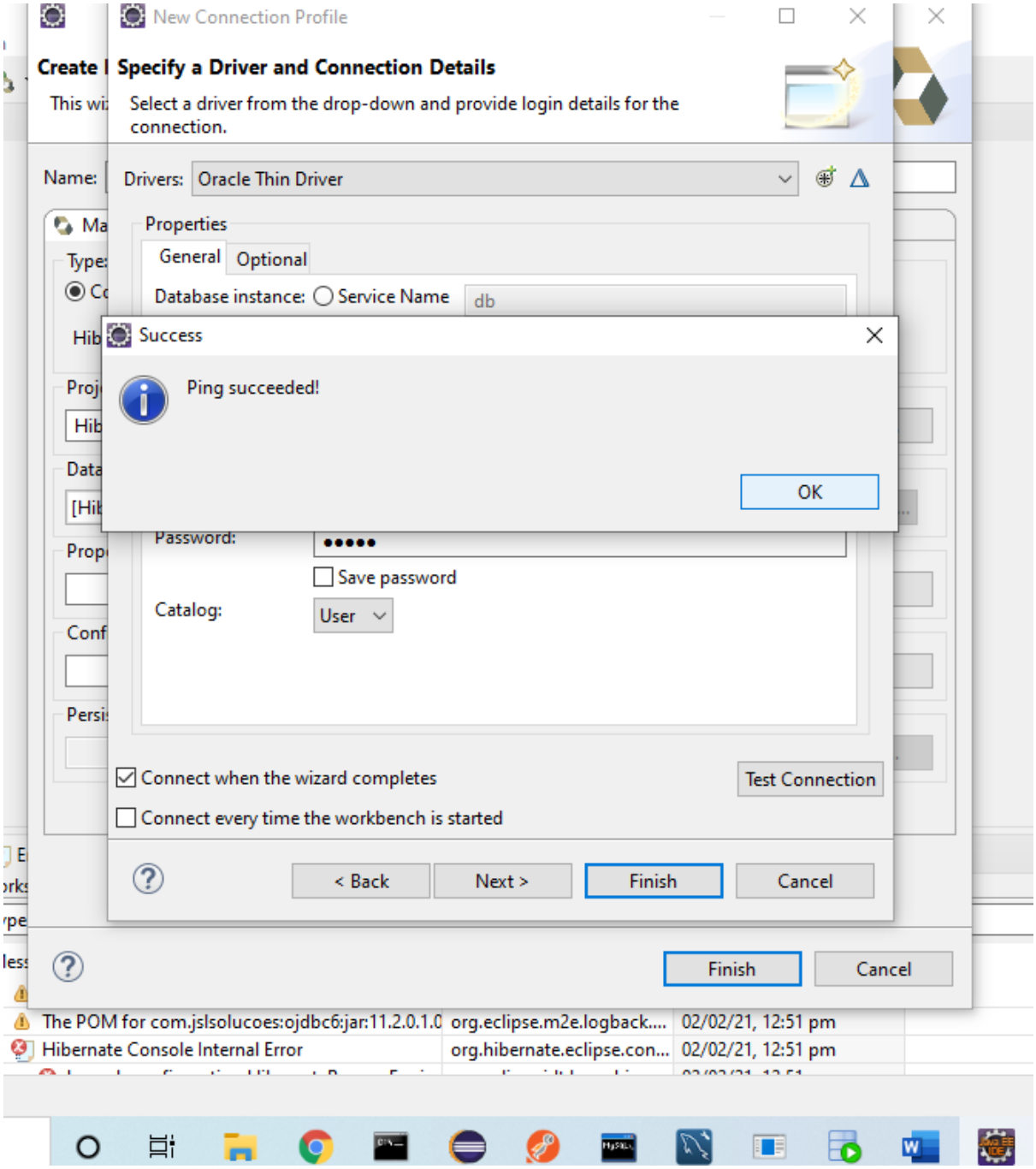
Next tab JAR List add ojdbc6.jar then ok



Step8:
Test connection

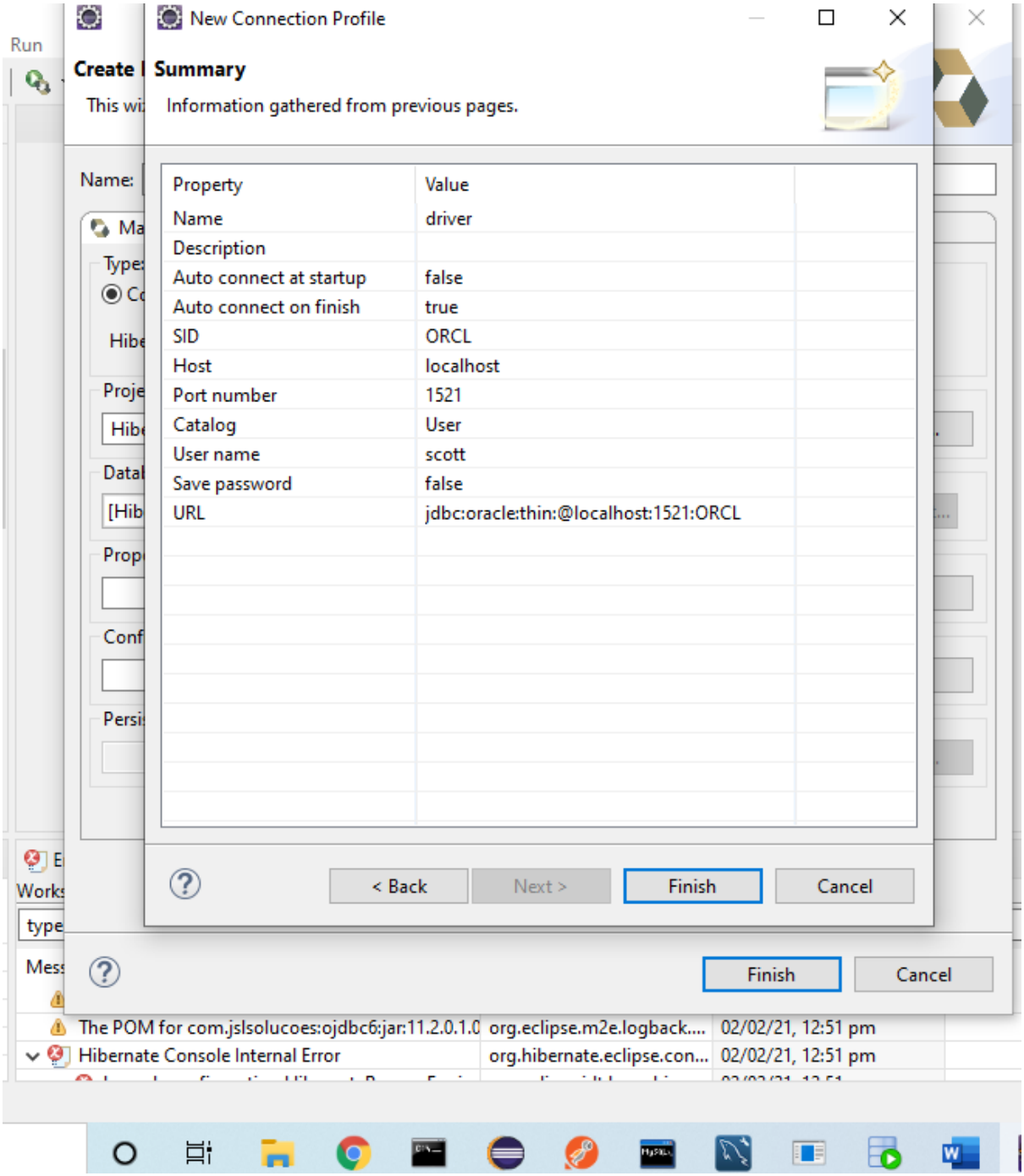

New Connection Profile
Specify a Driver and Connection Details
Select a driver from the drop-down and provide login details for the connection.
Drivers: Oracle Thin Driver
Properties
General
Optional
Database instance:
Service Name
db
SID
ORCL
Connection URL:
jdbc:oracle:thin:@localhost:1521:ORCL
Host:
localhost
Port number:
1521
User name:
scott
Password:
Save password
Catalog:
User
Connect when the wizard completes
Connect every time the workbench is started
Test Connection
< Back
Next >
Finish
Cancel
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

Step9:
next

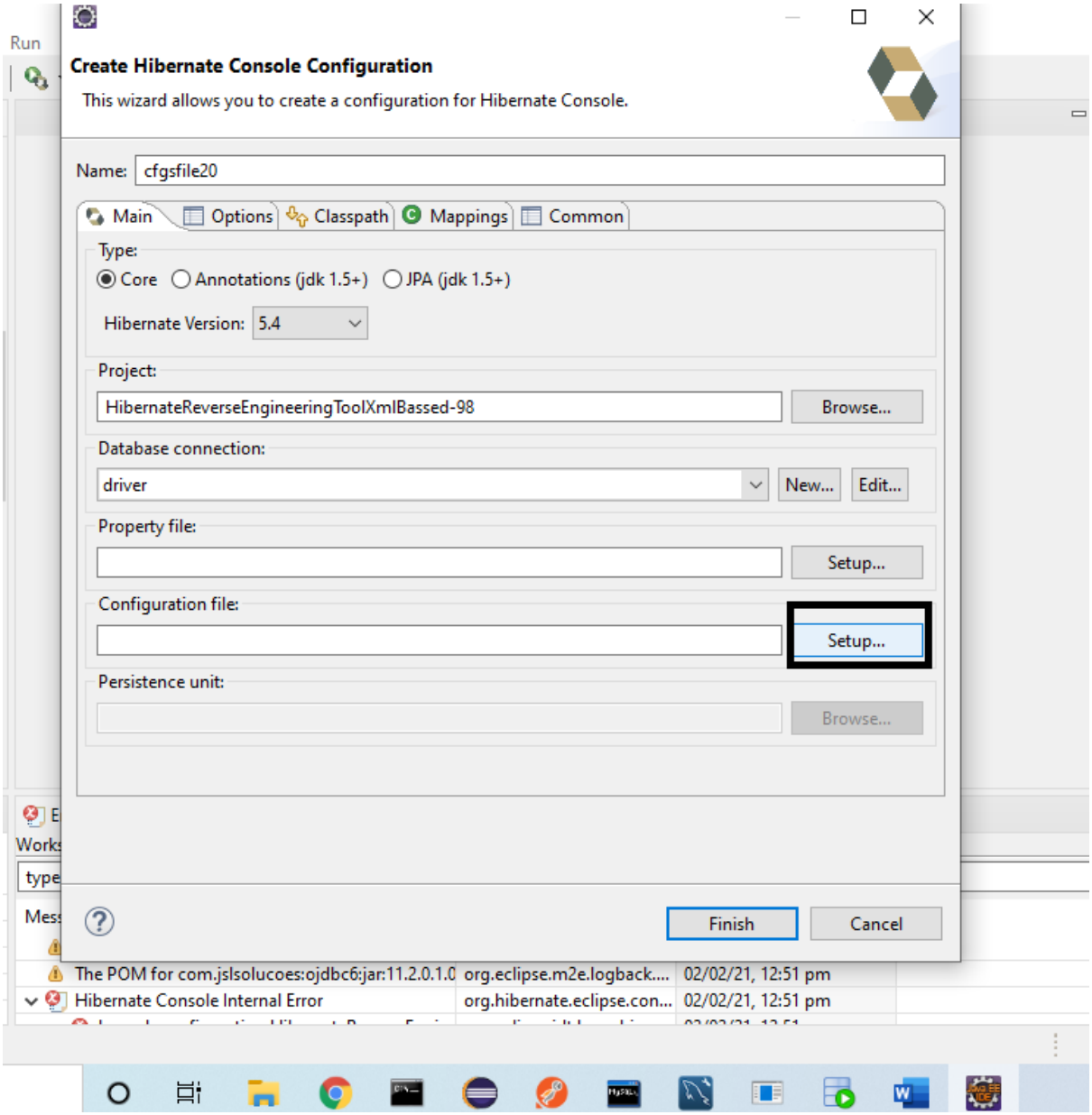


Step10:

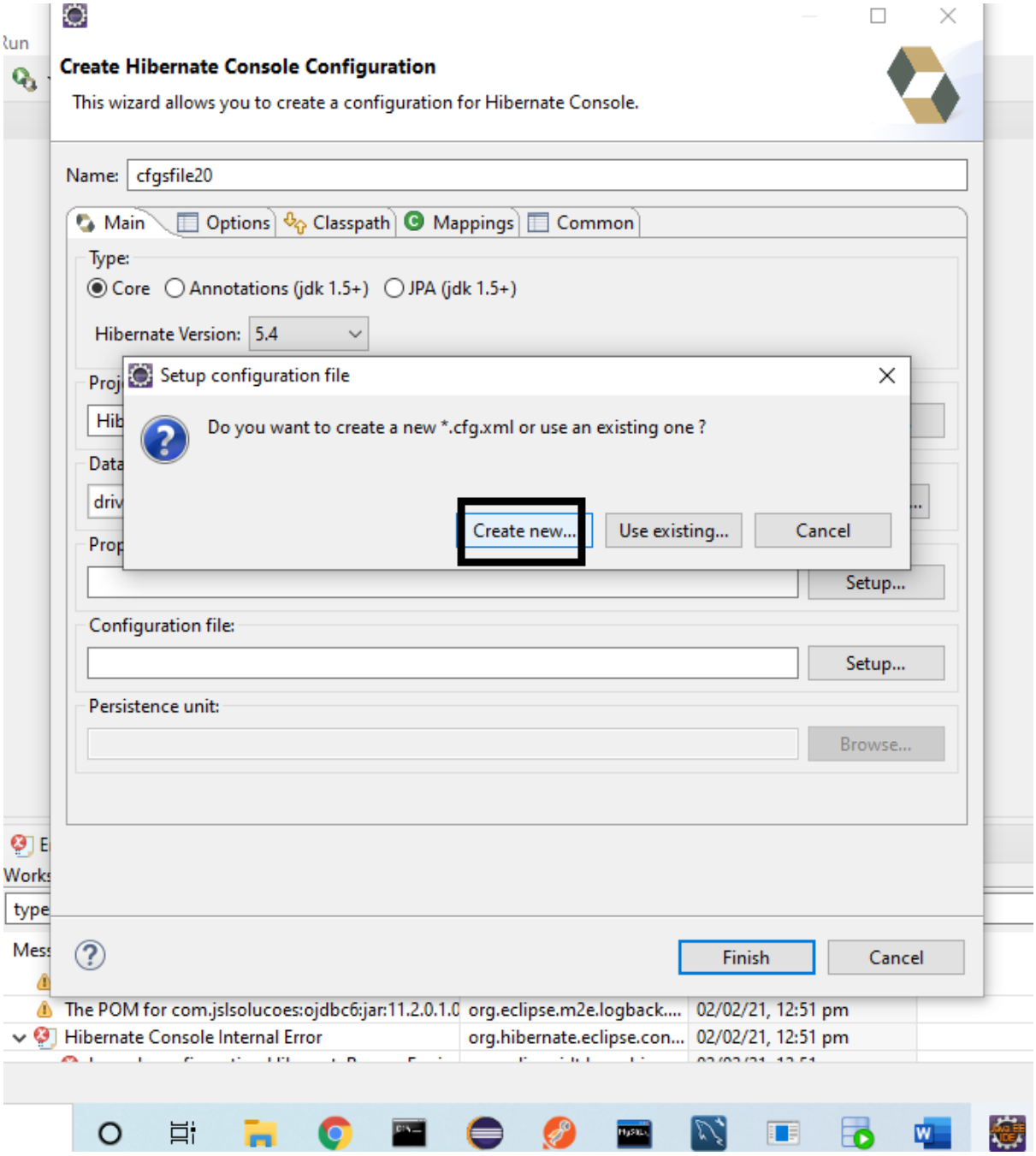
Finish


New Connection Profile
Summary
Information gathered from previous pages.
Property
Value
Name
driver
Description
Auto connect at startup
false
Auto connect on finish
true
SID
ORCL
Host
localhost
Port number
1521
Catalog
User
User name
scott
Save password
false
URL
jdbc:oracle:thin:@localhost:1521:ORCL
< Back
Next >
Finish
Cancel
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

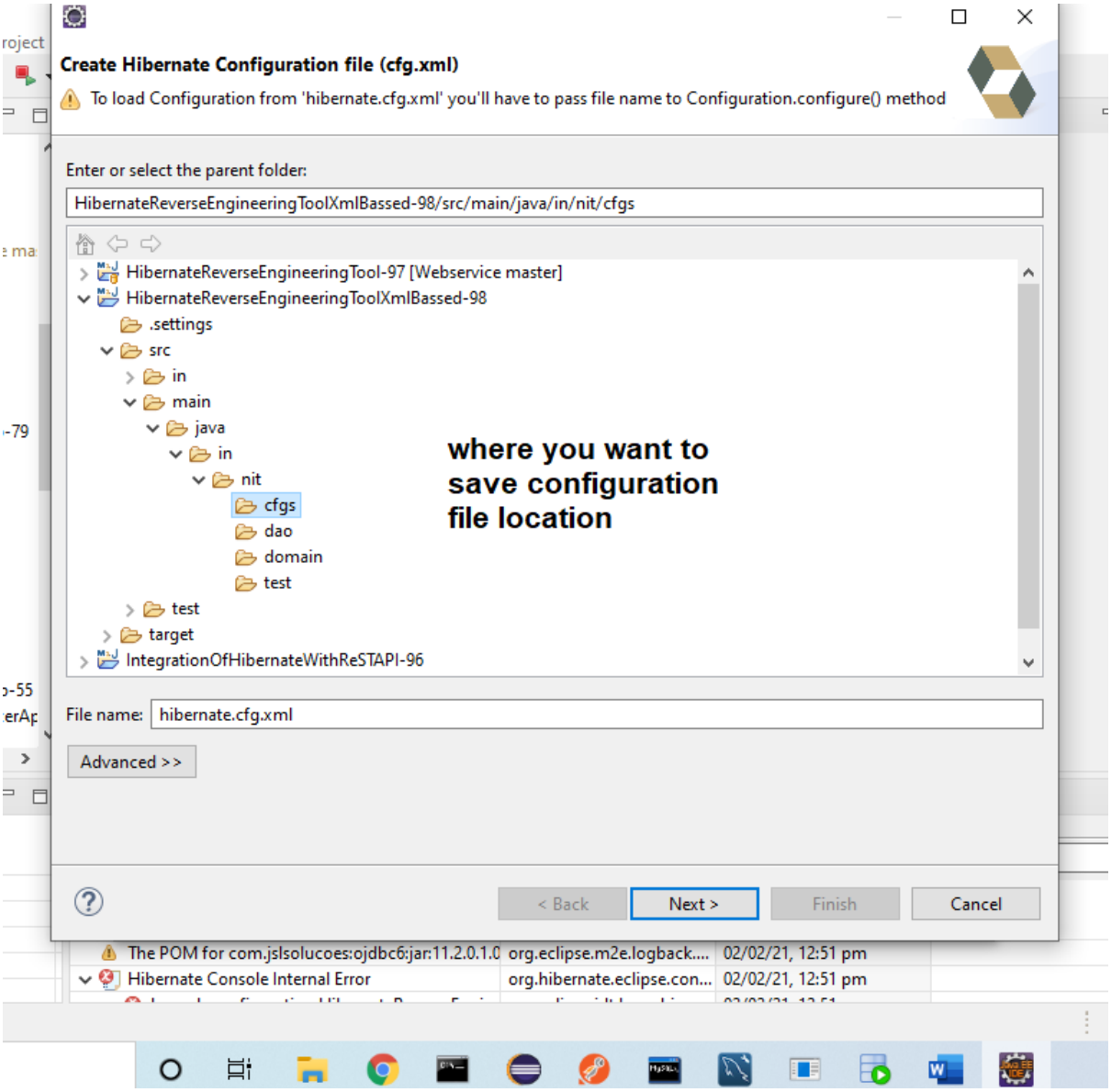
Step11:


Create Hibernate Console Configuration
This wizard allows you to create a configuration for Hibernate Console.
Name: cfgsfile20
Main
Options
Classpath
Mappings
Common
Type:
Core
Annotations (jdk 1.5+)
JPA (jdk 1.5+)
Hibernate Version: 5.4
Project:
HibernateReverseEngineeringToolXmlBassed-98
Browse...
Database connection:
driver
New...
Edit...
Property file:
Setup...
Configuration file:
Setup...
Persistence unit:
Browse...
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

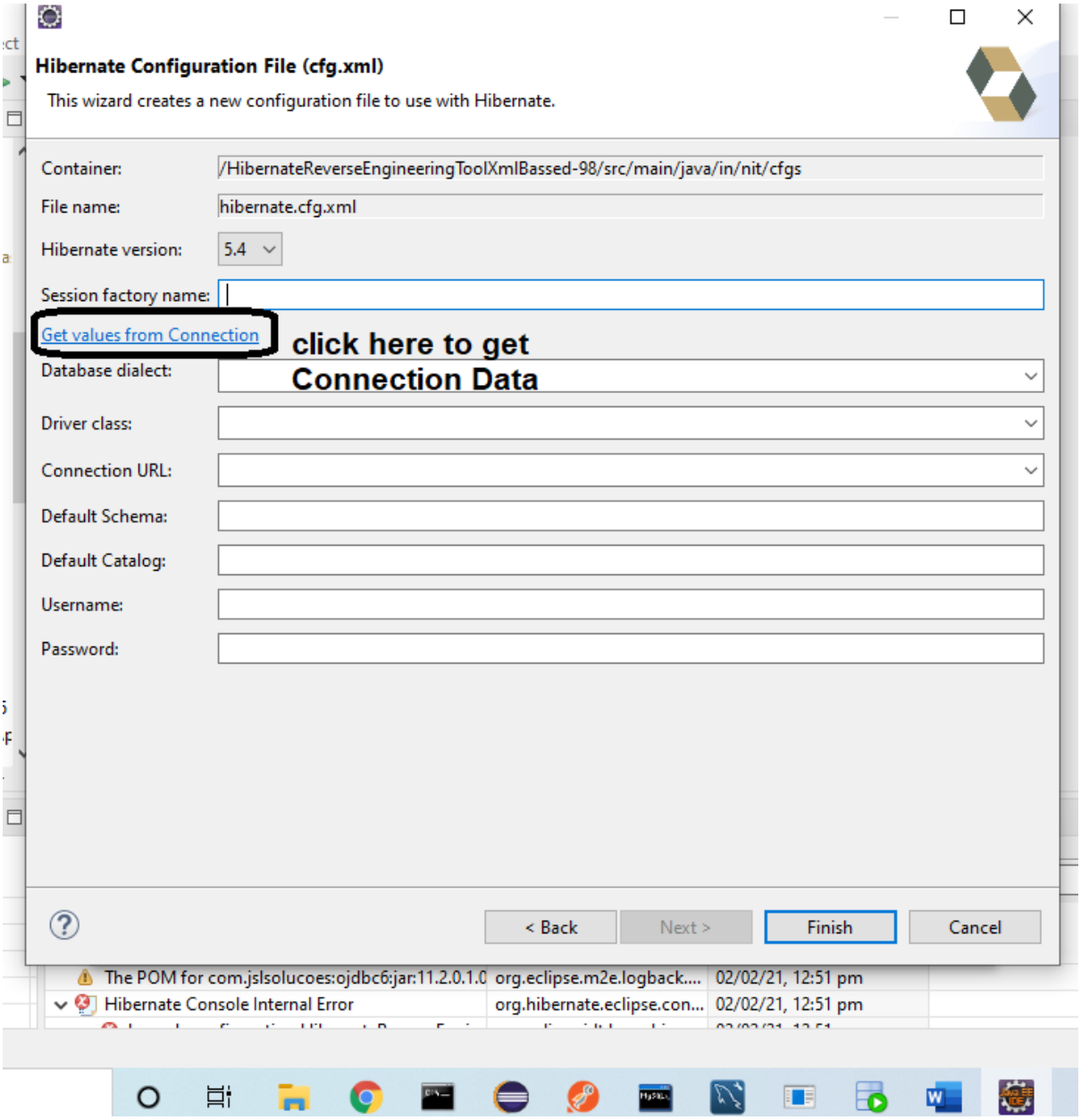
Step12:


Create Hibernate Console Configuration
This wizard allows you to create a configuration for Hibernate Console.
Name: cfgsfile20
Main
Options
Classpath
Mappings
Common
Type:
Core
Annotations (jdk 1.5+)
JPA (jdk 1.5+)
Hibernate Version: 5.4
Setup configuration file
Do you want to create a new *.cfg.xml or use an existing one ?
Create new...
Use existing...
Cancel
Setup...
Configuration file:
Setup...
Persistence unit:
Browse...
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

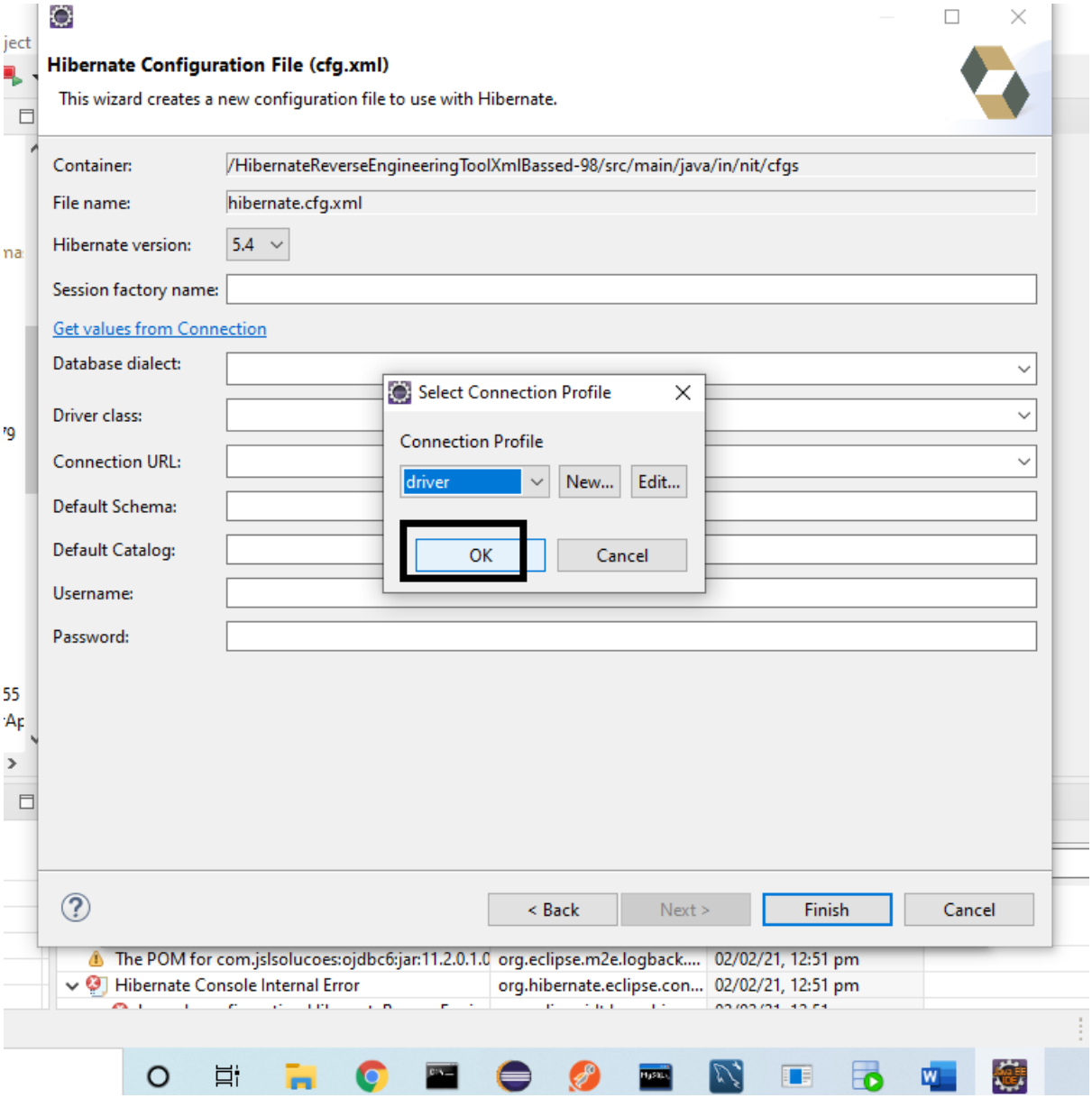
Step13:


Create Hibernate Configuration file (cfg.xml)
To load Configuration from 'hibernate.cfg.xml' you'll have to pass file name to Configuration.configure() method
Enter or select the parent folder:
HibernateReverseEngineeringToolXmlBassed-98/src/main/java/in/nit/cfgs
HibernateReverseEngineeringTool-97 [Webservice master]
HibernateReverseEngineeringToolXmlBassed-98
.settings
src
in
main
java
in
nit
cfgs
dao
domain
test
test
target
IntegrationOfHibernateWithReSTAPI-96
where you want to save configuration file location
File name:
hibernate.cfg.xml
Advanced >>
< Back
Next >
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

Step14:

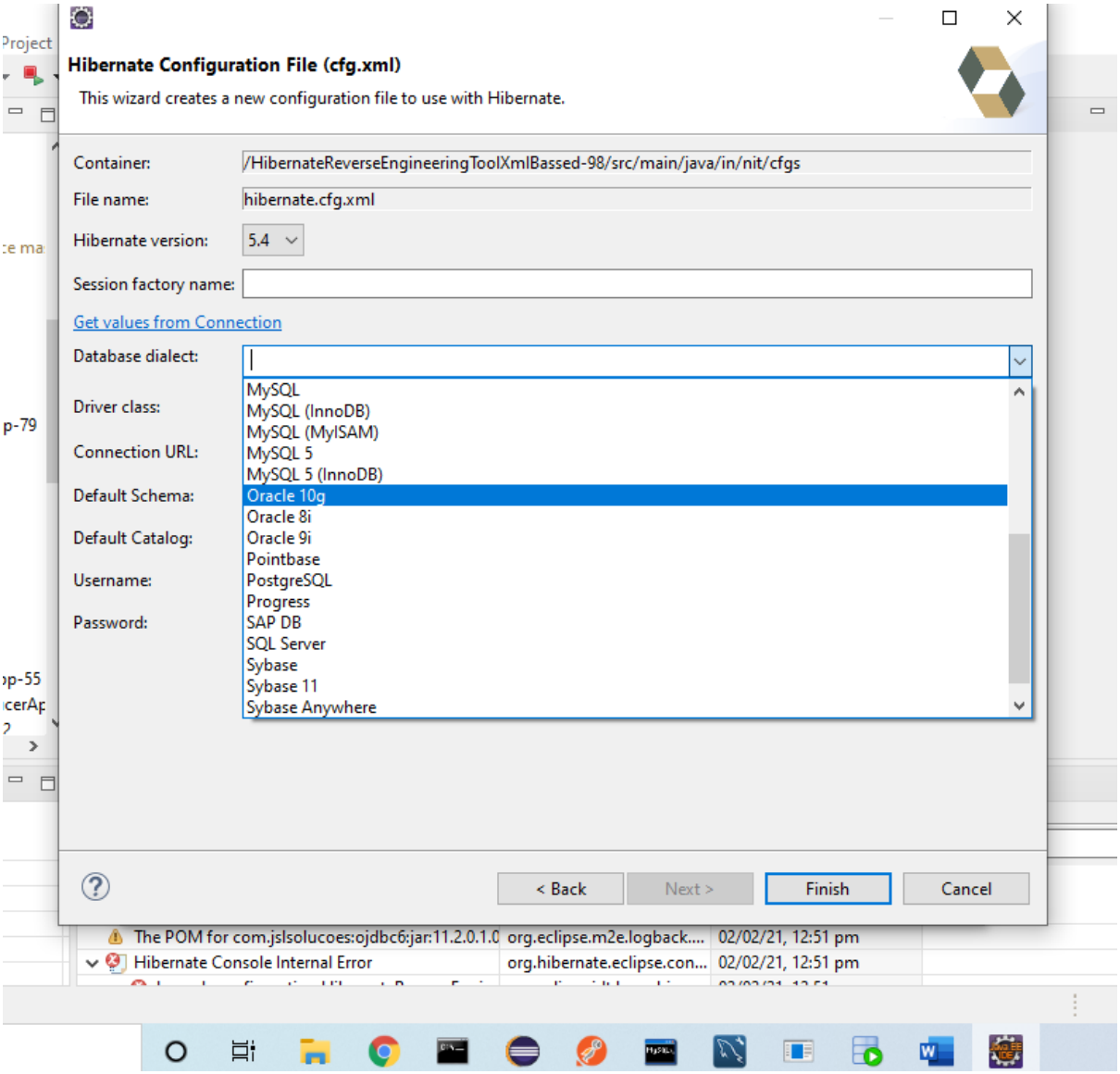

Hibernate Configuration File (cfg.xml)
This wizard creates a new configuration file to use with Hibernate.
Container:
/HibernateReverseEngineeringToolXmlBassed-98/src/main/java/in/nit/cfgs
File name:
hibernate.cfg.xml
Hibernate version:
5.4
Session factory name:
Get values from Connection
click here to get
Connection Data
Database dialect:
Driver class:
Connection URL:
Default Schema:
Default Catalog:
Username:
Password:
< Back
Next >
Finish
Cancel
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

Step15:



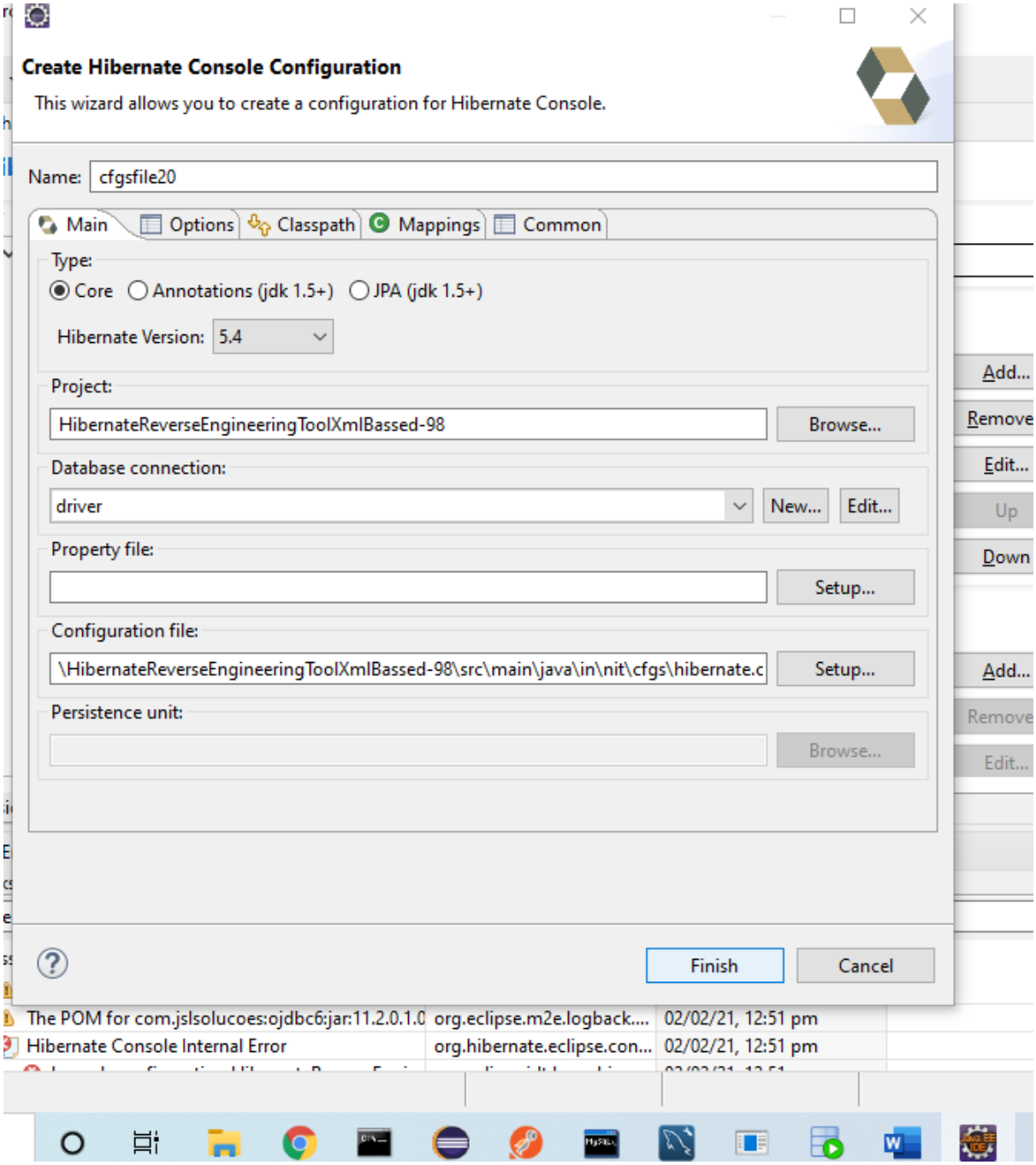
Step16:

Select dialect

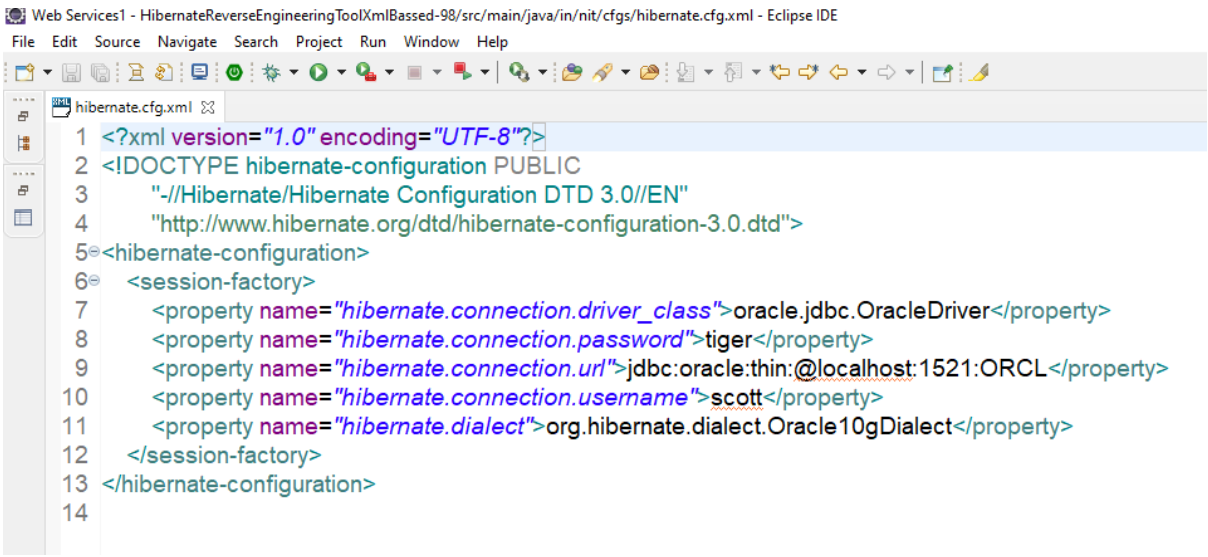


Step17:

Finish


Create Hibernate Console Configuration
This wizard allows you to create a configuration for Hibernate Console.
Name: cfgsfile20
Main
Options
Classpath
Mappings
Common
Type:
Core
Annotations (jdk 1.5+)
JPA (jdk 1.5+)
Hibernate Version: 5.4
Project:
HibernateReverseEngineeringToolXmlBassed-98
Browse...
Database connection:
driver
New...
Edit...
Property file:
Setup...
Configuration file:
\HibernateReverseEngineeringToolXmlBassed-98\src\main\java\in\nit\cfgs\hibernate.c
Setup...
Persistence unit:
Browse...
Finish
Cancel
Add...
Remove
Edit...
Up
Down
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
Hibernate Console Internal Error
org.hibernate.eclipse.con...
02/02/21, 12:51 pm

Step18:

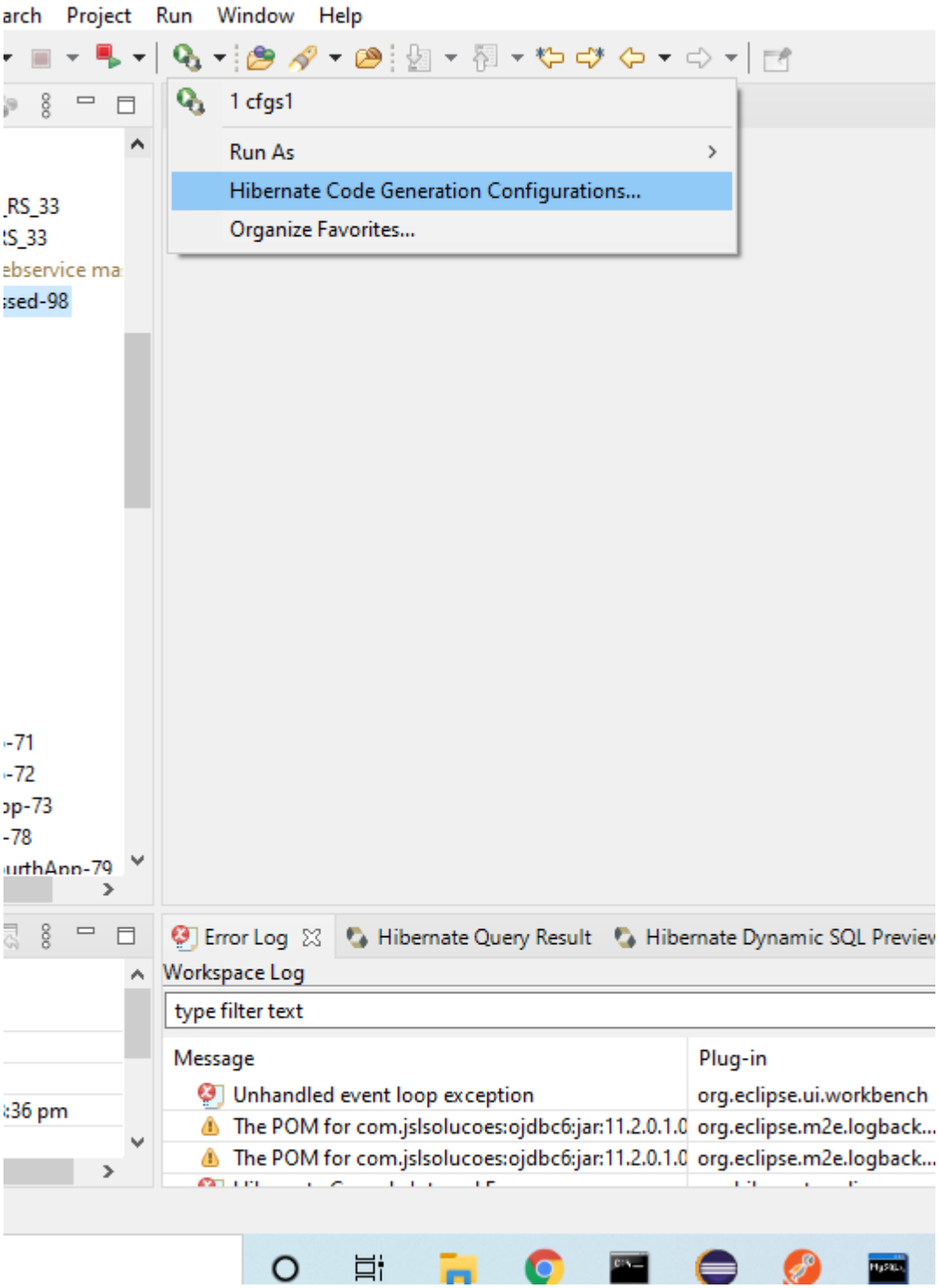



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
                "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
                "http://www.hibernate.org/dtd/hibernate-configuration-
3.0.dtd">
<hibernate-configuration>
   <session-factory>
      <property
name="hibernate.connection.driver_class">oracle.jdbc.OracleDriver</pr
operty>
      <property name="hibernate.connection.password">tiger</property>
      <property
name="hibernate.connection.url">jdbc:oracle:thin:@localhost:1521:ORC
L</property>
      <property
name="hibernate.connection.username">scott</property>
      <property
name="hibernate.dialect">org.hibernate.dialect.Oracle10gDialect</prope
rty>
   </session-factory>
</hibernate-configuration>
```

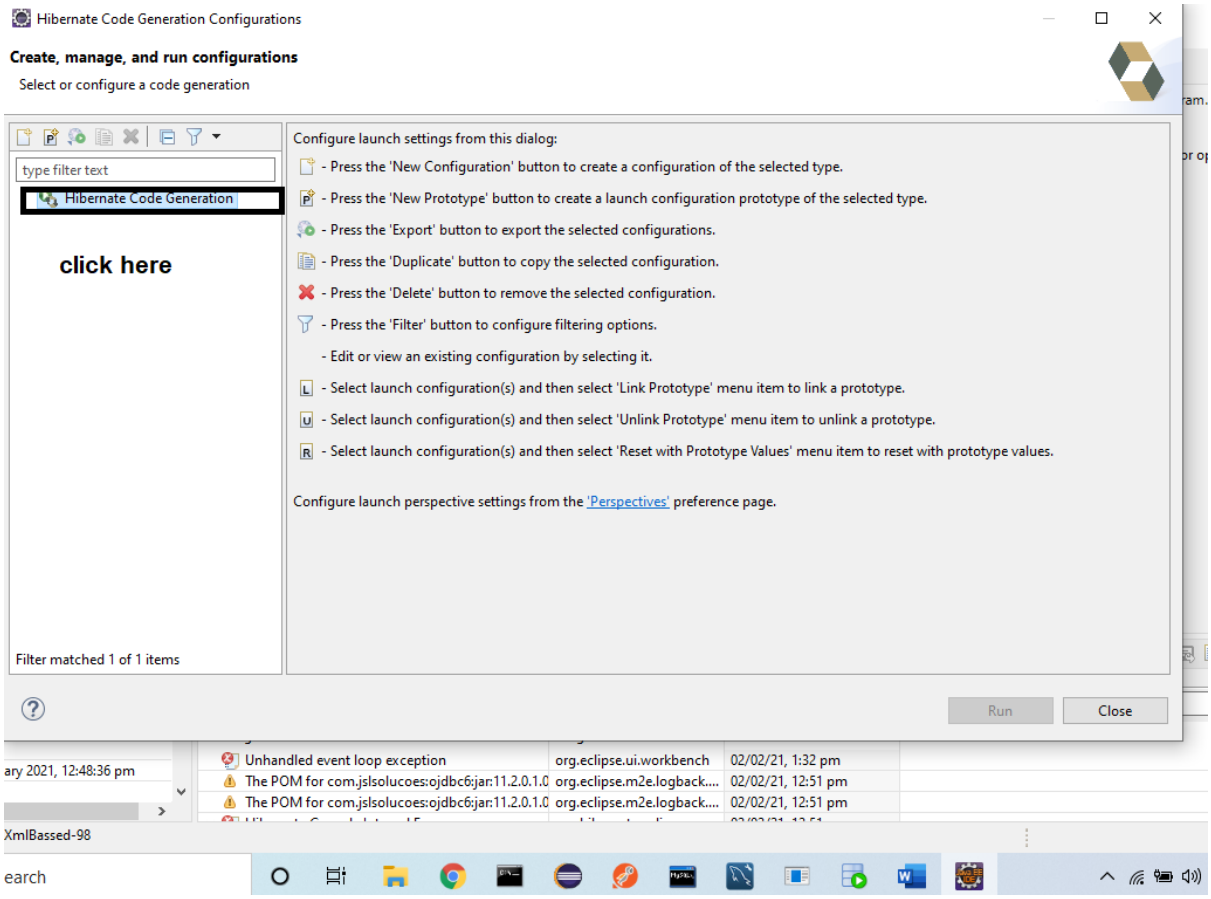
Configuration file generation Completed.

Mapping File Generation

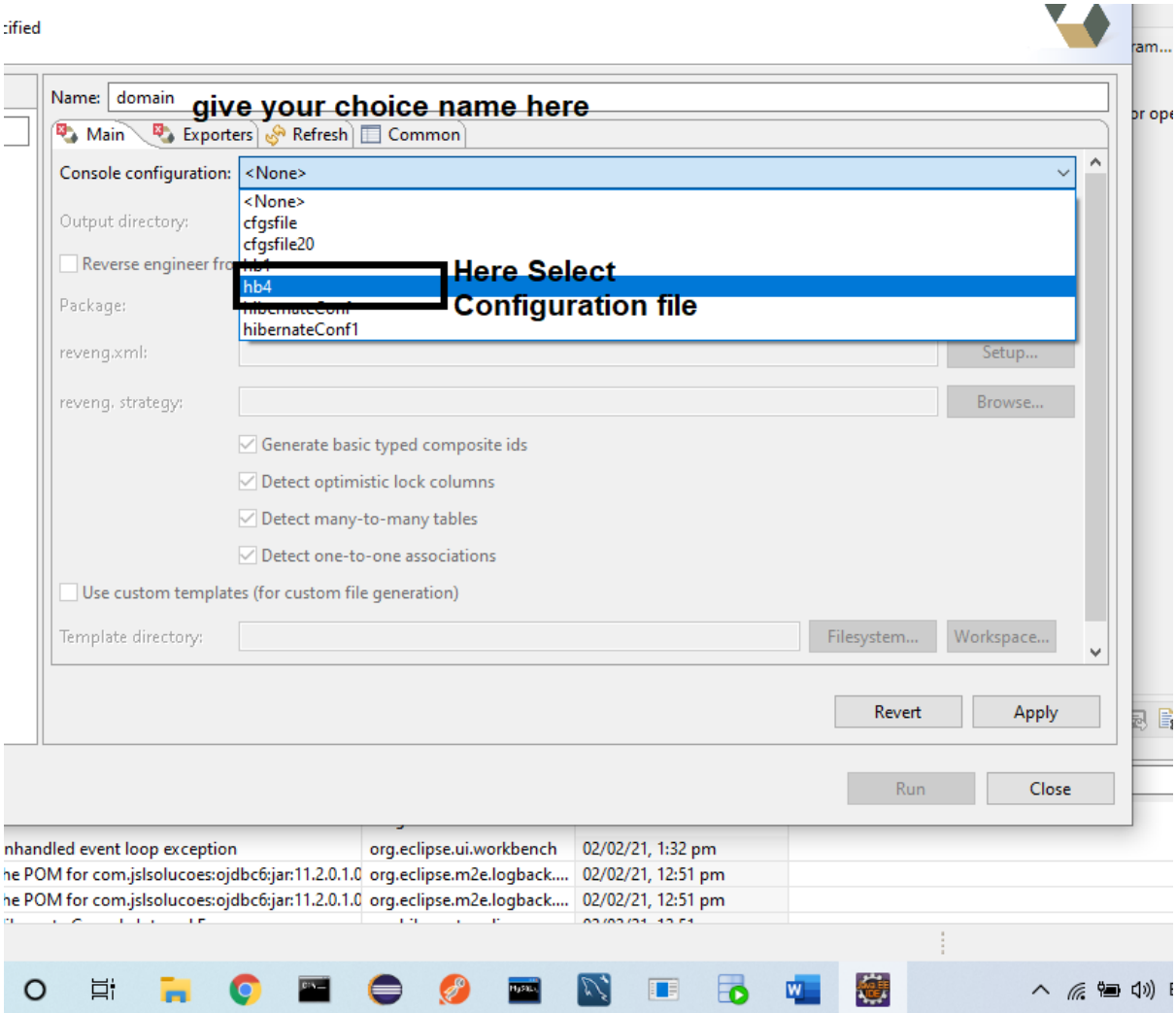
Step1:


arch Project Run Window Help
1 cfgs1
Run As
Hibernate Code Generation Configurations...
Organize Favorites...
Error Log
Hibernate Query Result
Hibernate Dynamic SQL Preview
Workspace Log
type filter text
Message
Plug-in
Unhandled event loop exception
org.eclipse.ui.workbench
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback...
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback...

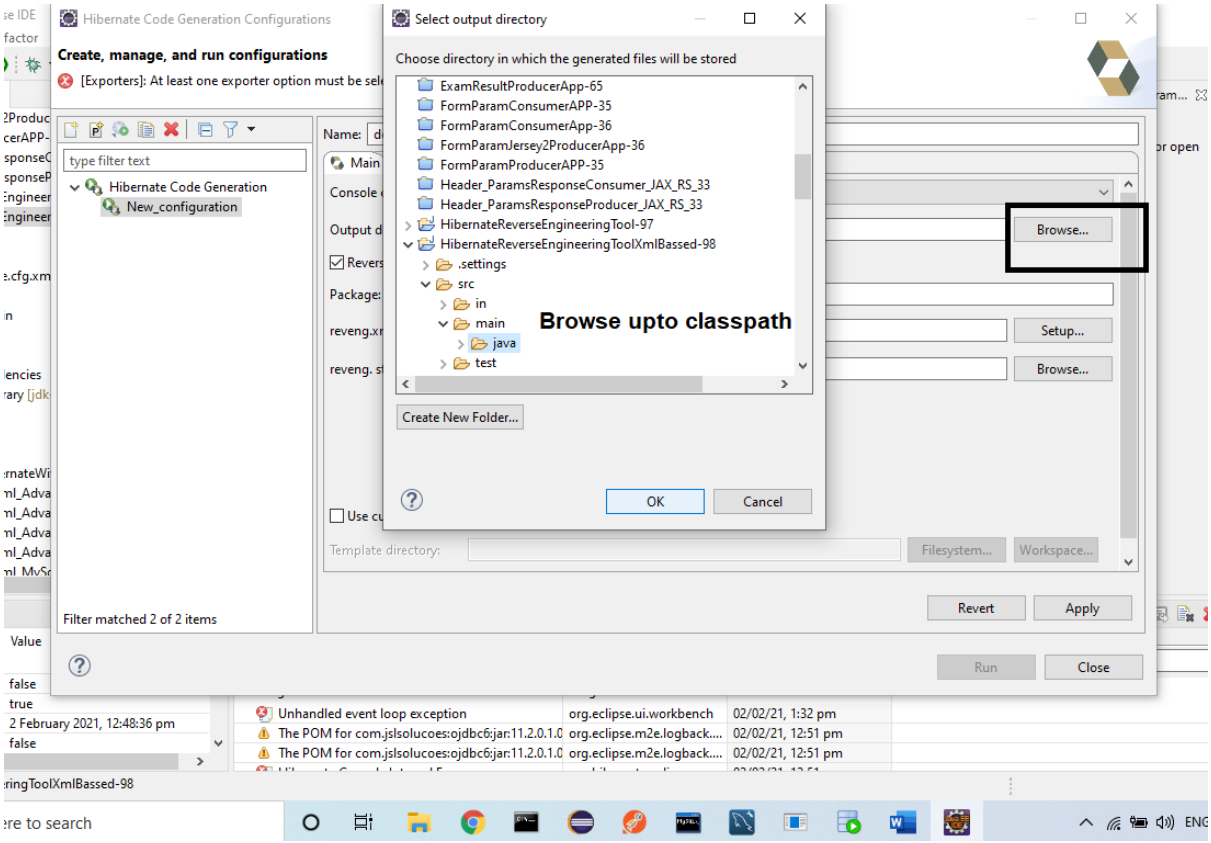
Step2:


Hibernate Code Generation Configurations
Create, manage, and run configurations
Select or configure a code generation
type filter text
Hibernate Code Generation
click here
Filter matched 1 of 1 items
Configure launch settings from this dialog:
- Press the 'New Configuration' button to create a configuration of the selected type.
- Press the 'New Prototype' button to create a launch configuration prototype of the selected type.
- Press the 'Export' button to export the selected configurations.
- Press the 'Duplicate' button to copy the selected configuration.
- Press the 'Delete' button to remove the selected configuration.
- Press the 'Filter' button to configure filtering options.
- Edit or view an existing configuration by selecting it.
- Select launch configuration(s) and then select 'Link Prototype' menu item to link a prototype.
- Select launch configuration(s) and then select 'Unlink Prototype' menu item to unlink a prototype.
- Select launch configuration(s) and then select 'Reset with Prototype Values' menu item to reset with prototype values.
Configure launch perspective settings from the 'Perspectives' preference page.
Run
Close
Unhandled event loop exception
org.eclipse.ui.workbench
02/02/21, 1:32 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
ary 2021, 12:48:36 pm
XmlBassed-98
earch

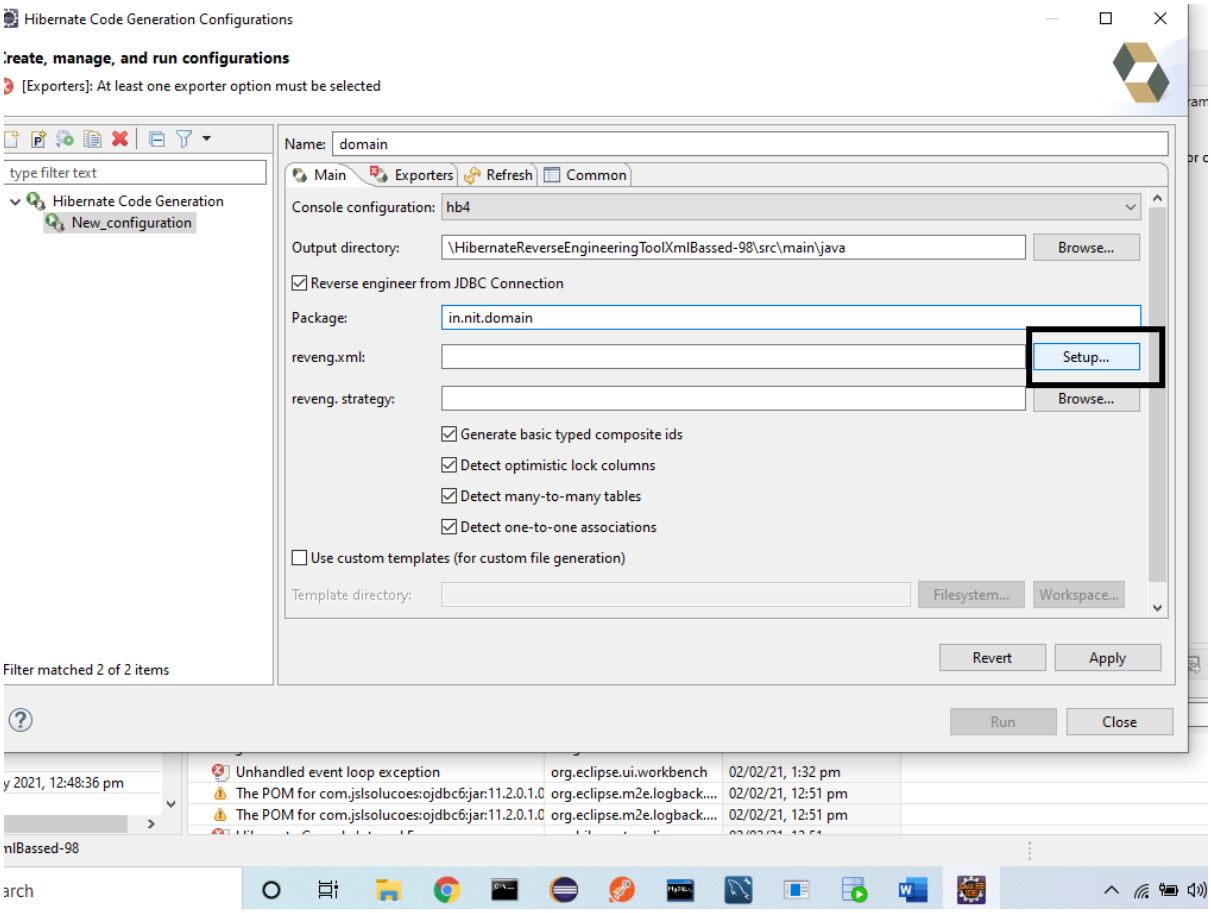
Step3:


Name: domain
give your choice name here
Main
Exporters
Refresh
Common
Console configuration: <None>
<None>
cfgsfile
cfgsfile20
hb4
hibernateConf1
Here Select
Configuration file
Output directory:
Reverse engineer fr
Package:
reveng.xml:
Setup...
reveng. strategy:
Browse...
Generate basic typed composite ids
Detect optimistic lock columns
Detect many-to-many tables
Detect one-to-one associations
Use custom templates (for custom file generation)
Template directory:
Filesystem...
Workspace...
Revert
Apply
Run
Close
nhandled event loop exception
org.eclipse.ui.workbench
02/02/21, 1:32 pm
he POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
he POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm

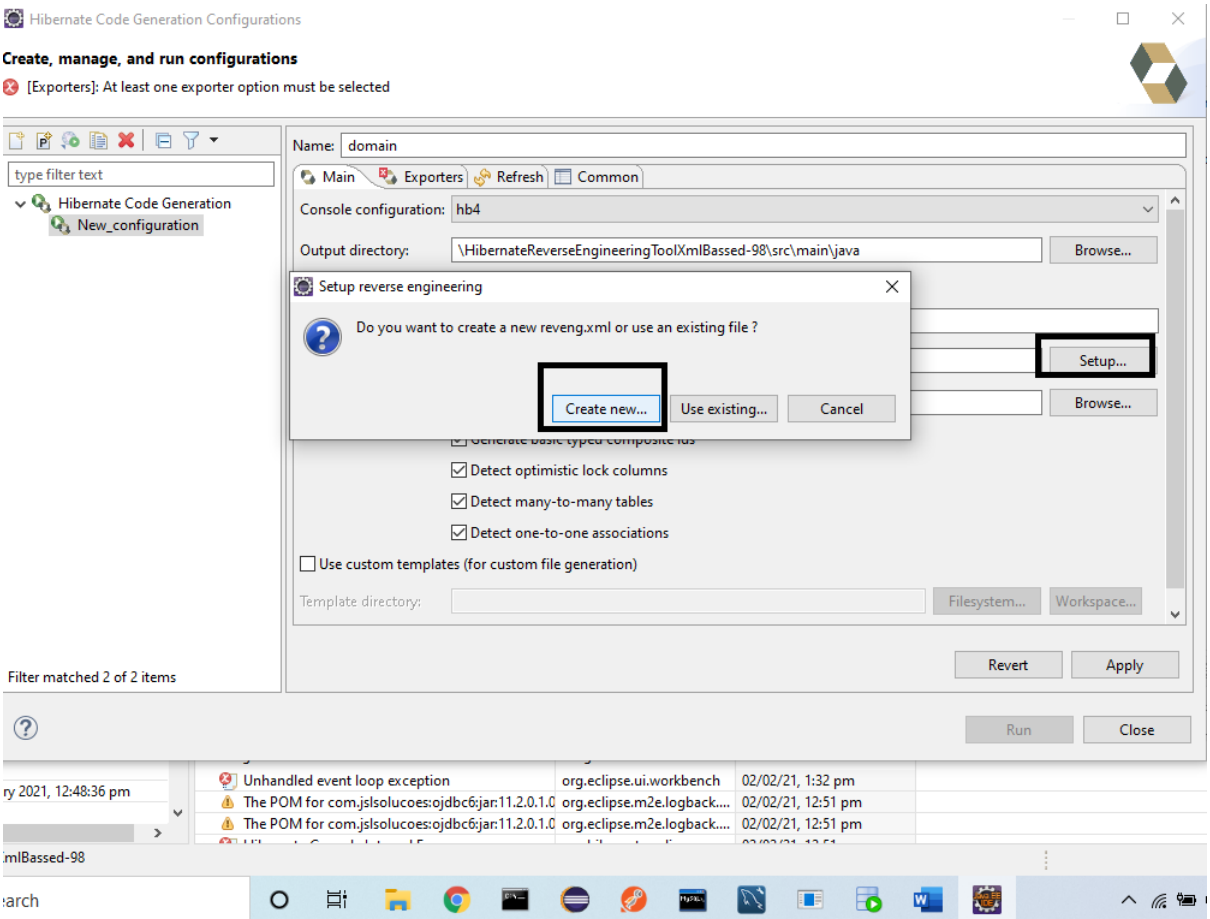
Step4:


Hibernate Code Generation Configurations
Create, manage, and run configurations
[Exporters]: At least one exporter option must be sel
type filter text
Hibernate Code Generation
New_configuration
Filter matched 2 of 2 items
Name:
Main
Console
Output d
Revers
Package:
Use cu
Template directory:
Filesystem...
Workspace...
Revert
Apply
Run
Close
Select output directory
Choose directory in which the generated files will be stored
ExamResultProducerApp-65
FormParamConsumerAPP-35
FormParamConsumerApp-36
FormParamJersey2ProducerApp-36
FormParamProducerAPP-35
Header_ParamsResponseConsumer_JAX_RS_33
Header_ParamsResponseProducer_JAX_RS_33
HibernateReverseEngineeringTool-97
HibernateReverseEngineeringToolXmlBassed-98
.settings
src
in
main
java
test
Browse upto classpath
Create New Folder...
OK
Cancel
Browse...
Setup...
Unhandled event loop exception
org.eclipse.ui.workbench
02/02/21, 1:32 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
2 February 2021, 12:48:36 pm

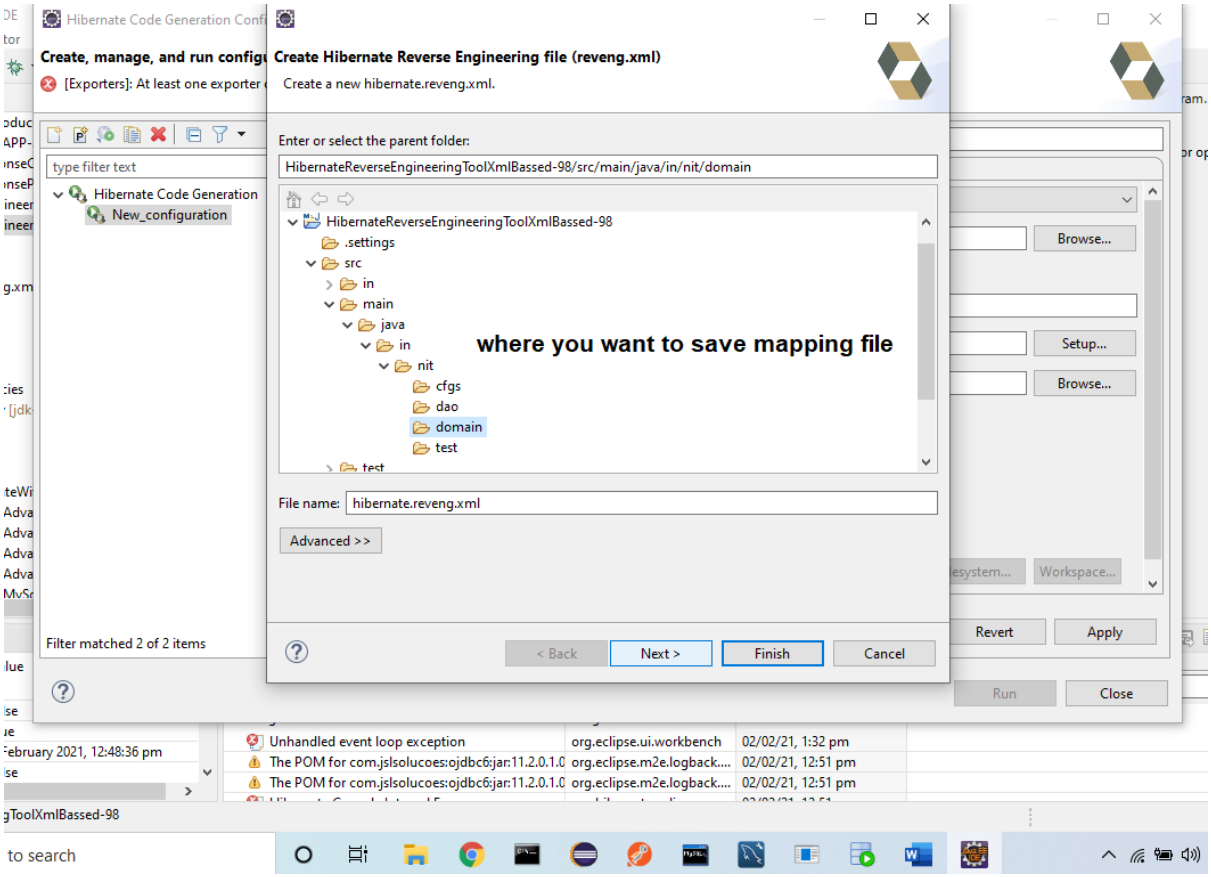
Step5:


Hibernate Code Generation Configurations
reate, manage, and run configurations
[Exporters]: At least one exporter option must be selected
type filter text
Hibernate Code Generation
New_configuration
Filter matched 2 of 2 items
Name: domain
Main
Exporters
Refresh
Common
Console configuration: hb4
Output directory: \HibernateReverseEngineeringToolXmlBassed-98\src\main\java
Browse...
Reverse engineer from JDBC Connection
Package: in.nit.domain
reveng.xml:
Setup...
reveng. strategy:
Browse...
Generate basic typed composite ids
Detect optimistic lock columns
Detect many-to-many tables
Detect one-to-one associations
Use custom templates (for custom file generation)
Template directory:
Filesystem...
Workspace...
Revert
Apply
Run
Close
Unhandled event loop exception
org.eclipse.ui.workbench
02/02/21, 1:32 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm

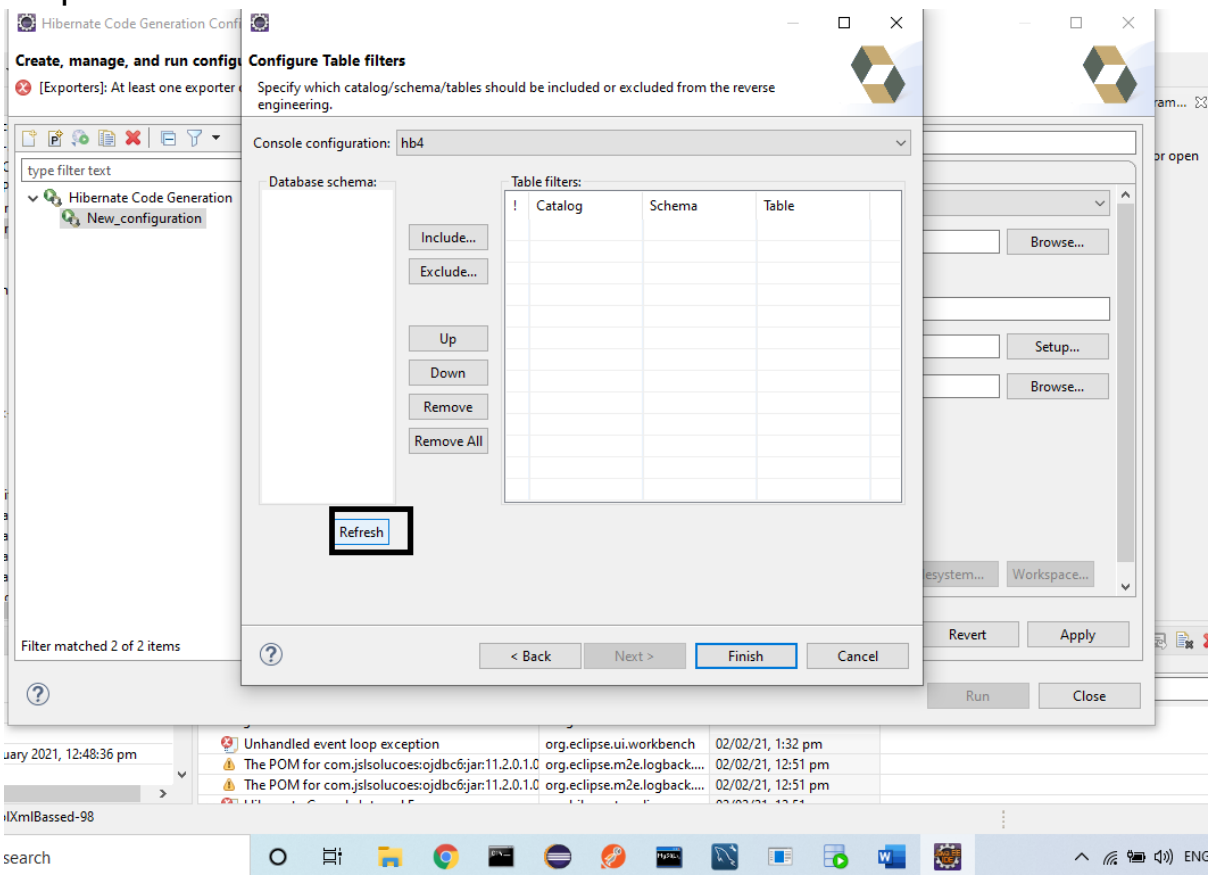
Step6:


Hibernate Code Generation Configurations
Create, manage, and run configurations
[Exporters]: At least one exporter option must be selected
type filter text
Hibernate Code Generation
New_configuration
Name: domain
Main
Exporters
Refresh
Common
Console configuration: hb4
Output directory: \HibernateReverseEngineeringToolXmlBassed-98\src\main\java
Browse...
Setup reverse engineering
Do you want to create a new reveng.xml or use an existing file ?
Create new...
Use existing...
Cancel
Setup...
Browse...
Detect optimistic lock columns
Detect many-to-many tables
Detect one-to-one associations
Use custom templates (for custom file generation)
Template directory:
Filesystem...
Workspace...
Filter matched 2 of 2 items
Revert
Apply
Run
Close
Unhandled event loop exception
org.eclipse.ui.workbench
02/02/21, 1:32 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm

Step7:

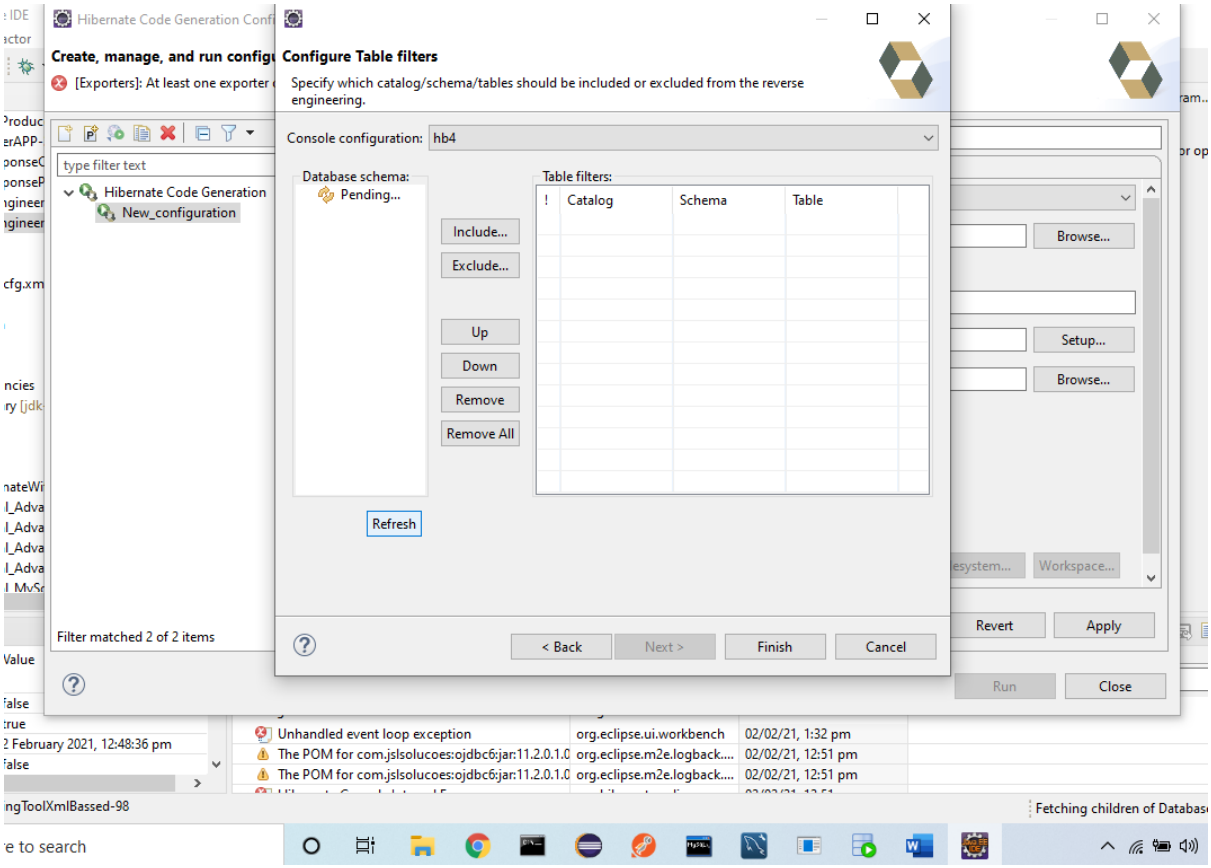

Create Hibernate Reverse Engineering file (reveng.xml)
Create a new hibernate.reveng.xml.
Enter or select the parent folder:
HibernateReverseEngineeringToolXmlBassed-98/src/main/java/in/nit/domain
where you want to save mapping file
File name: hibernate.reveng.xml
Advanced >>
< Back
Next >
Finish
Cancel

Step8:


Configure Table filters
Specify which catalog/schema/tables should be included or excluded from the reverse engineering.
Console configuration: hb4
Database schema:
Table filters:
Catalog
Schema
Table
Include...
Exclude...
Up
Down
Remove
Remove All
Refresh
< Back
Next >
Finish
Cancel

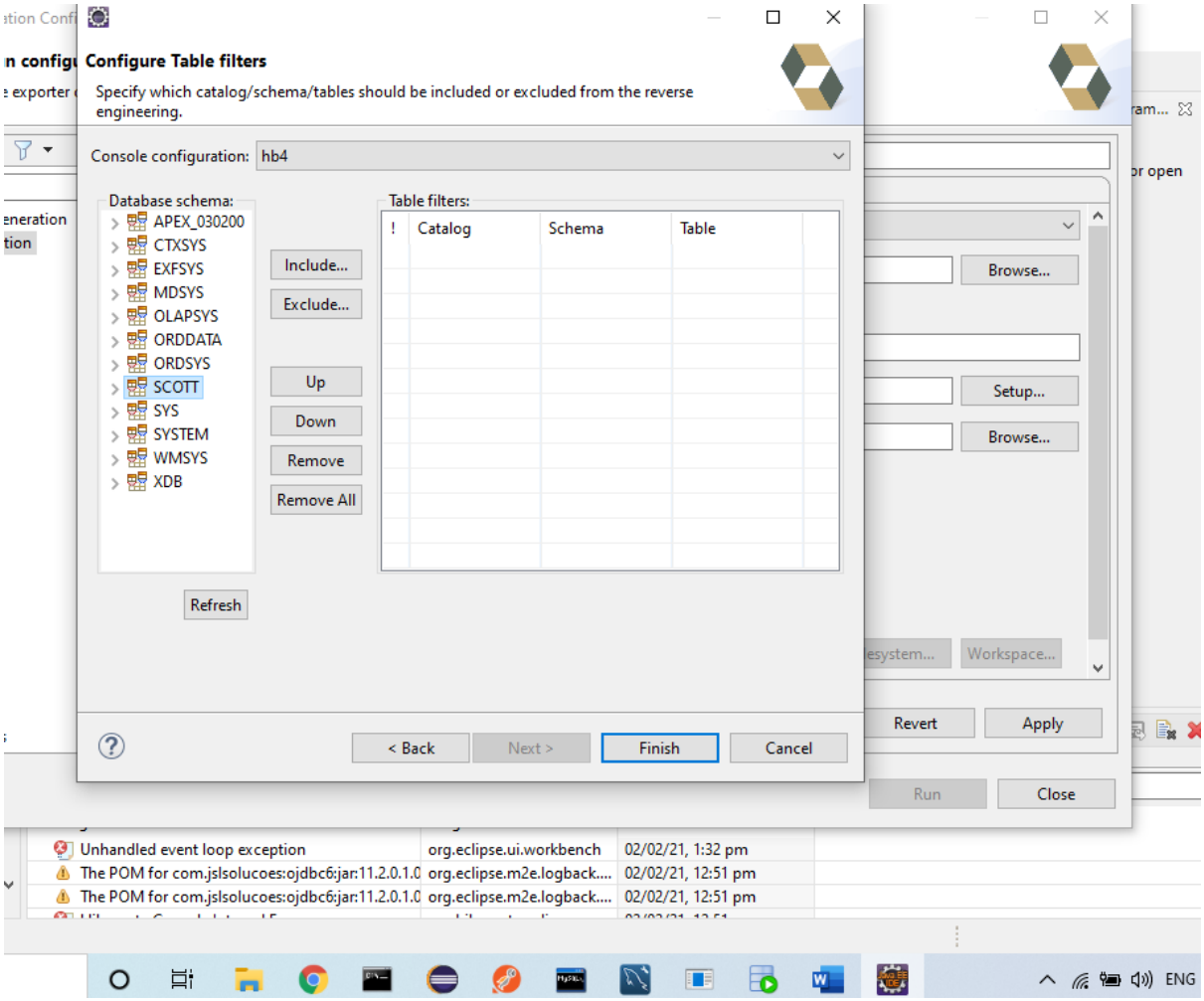
Step9:

Wait few Minutes



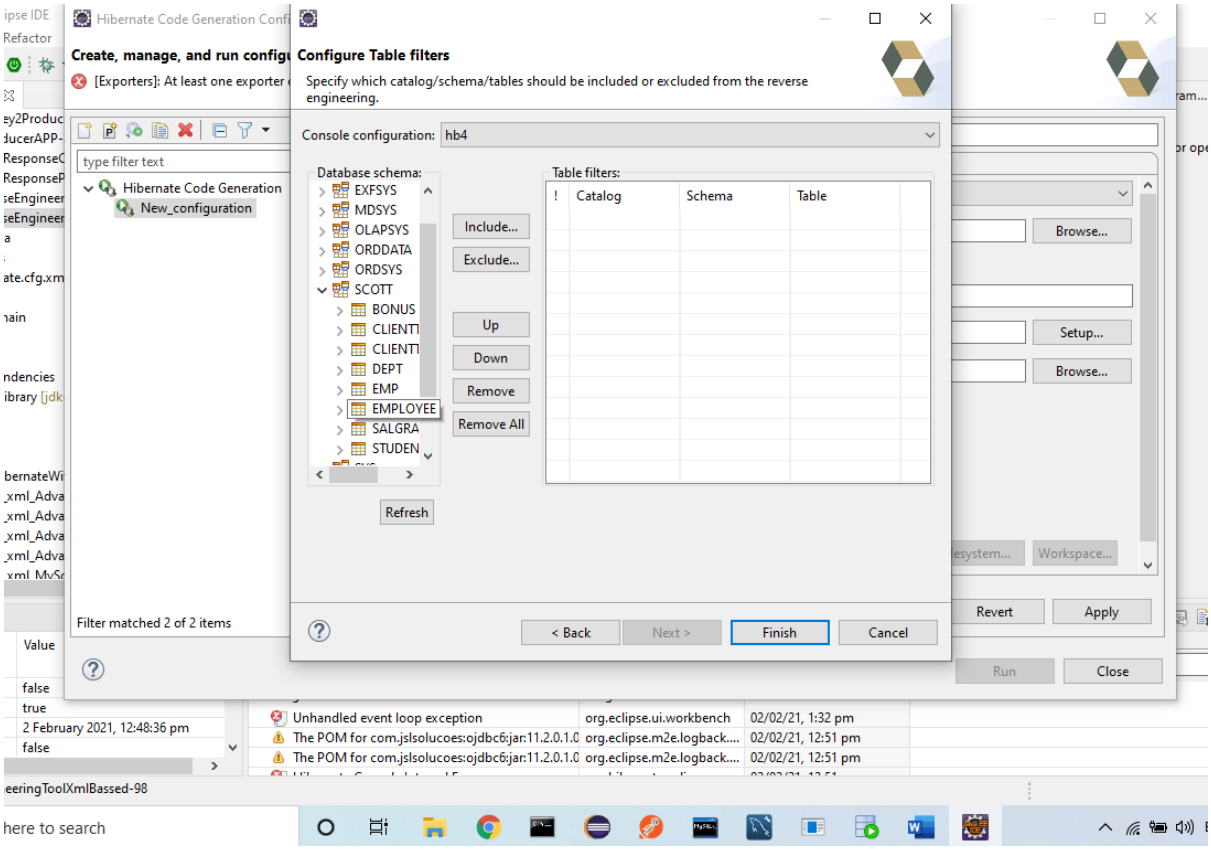
Step10:

Select user



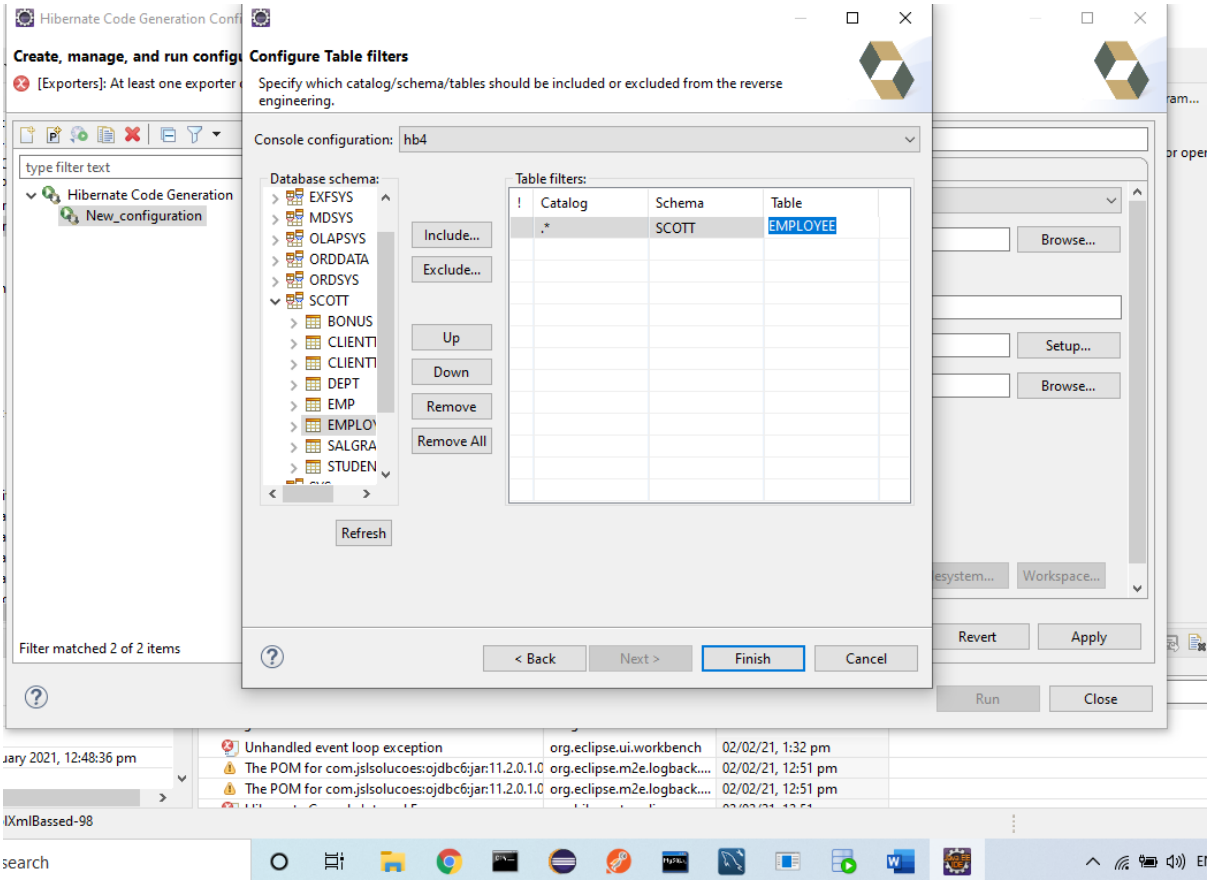
Step11:

Here Select table and include



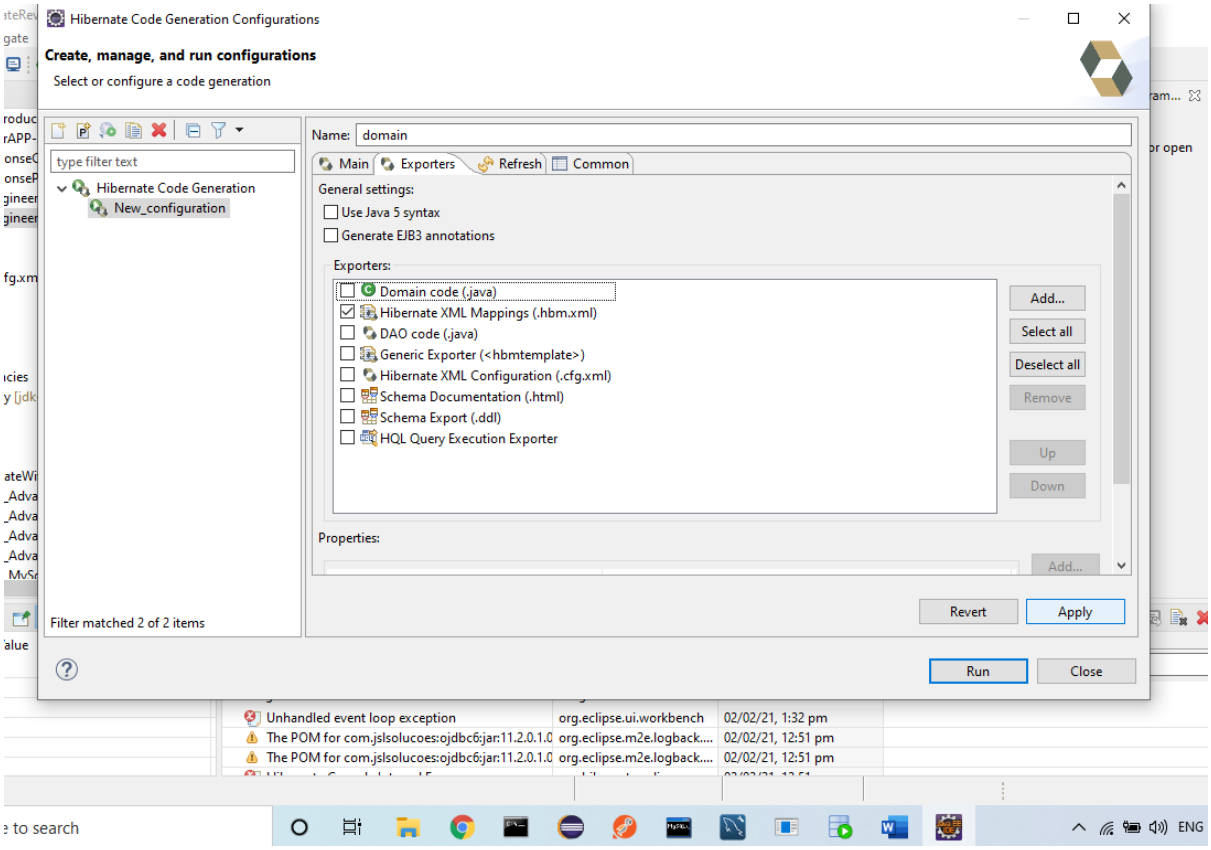
Step12:

Select table finish

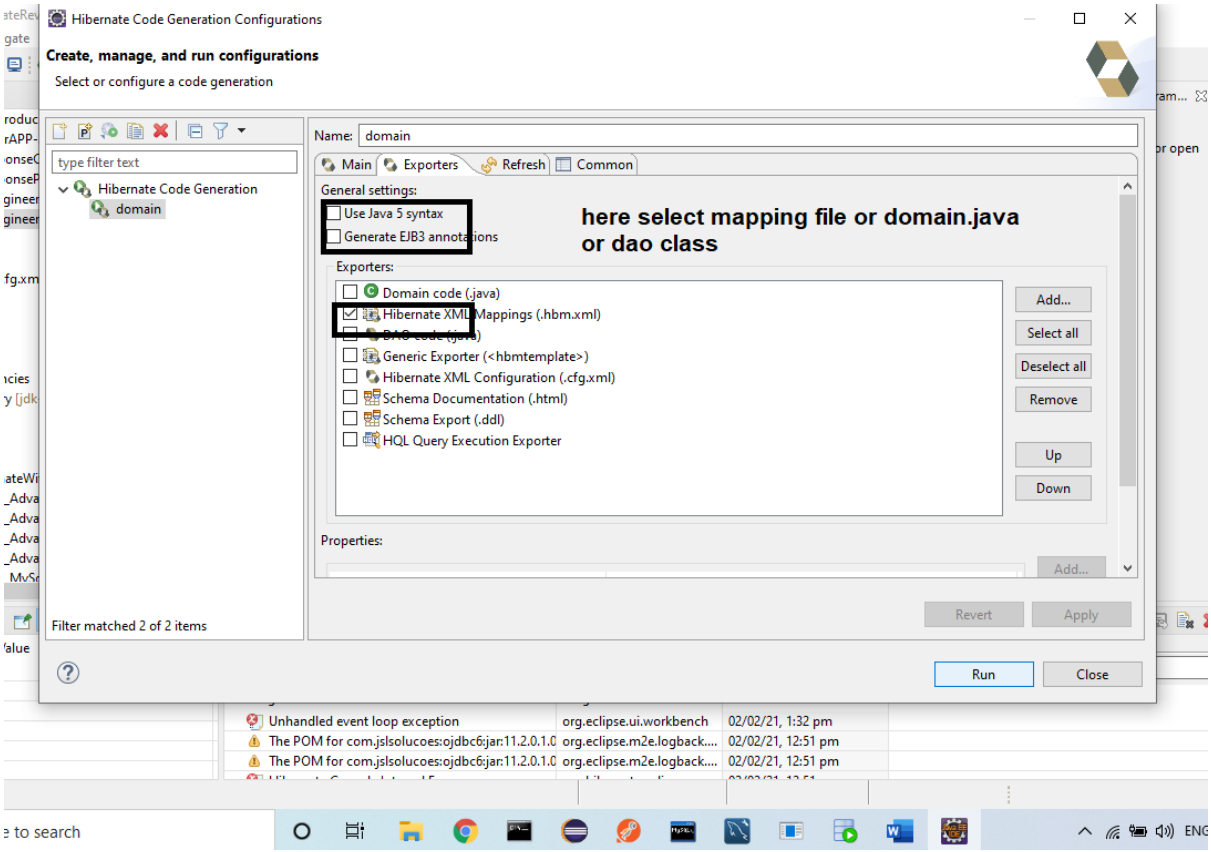


Step13:

Go to Exports tab

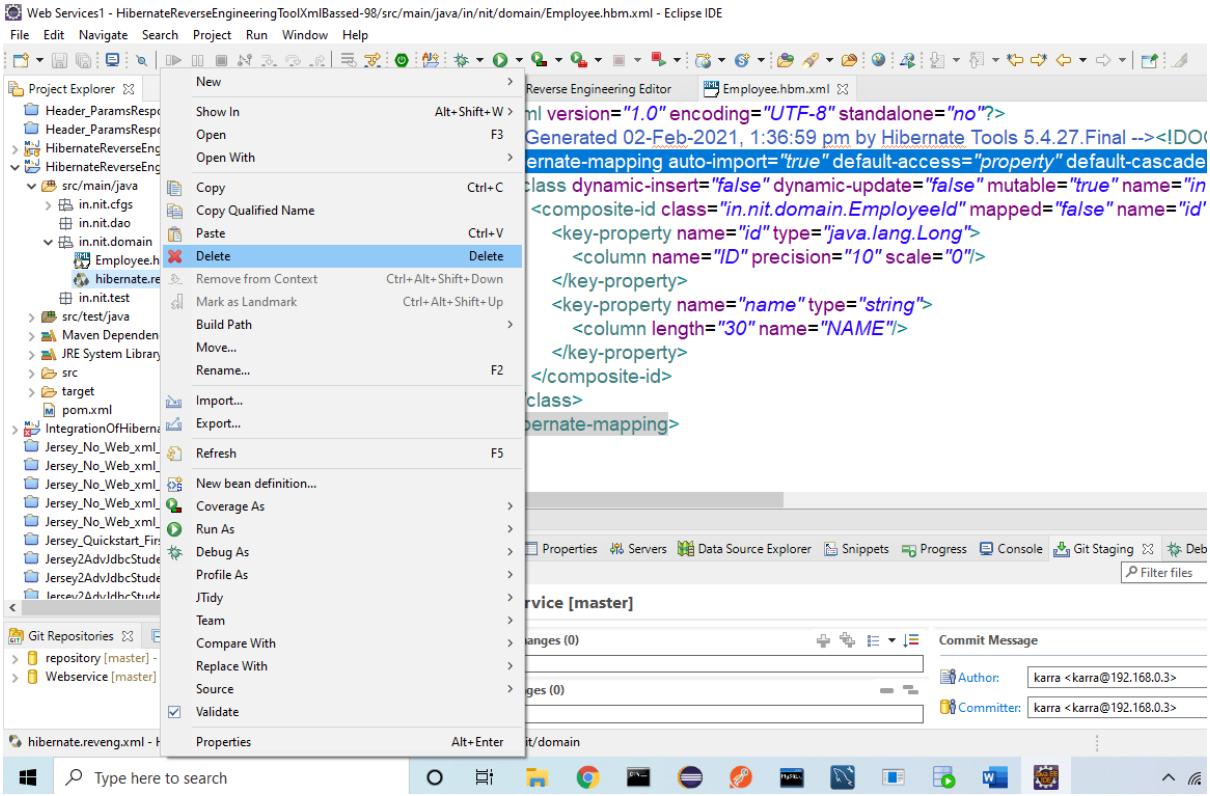

Hibernate Code Generation Configurations
Create, manage, and run configurations
Select or configure a code generation
type filter text
Hibernate Code Generation
New_configuration
Filter matched 2 of 2 items
Name: domain
Main
Exporters
Refresh
Common
General settings:
Use Java 5 syntax
Generate EJB3 annotations
Exporters:
Domain code (.java)
Hibernate XML Mappings (.hbm.xml)
DAO code (.java)
Generic Exporter (<hbmtemplate>)
Hibernate XML Configuration (.cfg.xml)
Schema Documentation (.html)
Schema Export (.ddl)
HQL Query Execution Exporter
Add...
Select all
Deselect all
Remove
Up
Down
Properties:
Add...
Revert
Apply
Run
Close
Unhandled event loop exception
org.eclipse.ui.workbench
02/02/21, 1:32 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
The POM for com.jslsolucoes:ojdbc6:jar:11.2.0.1.0
org.eclipse.m2e.logback....
02/02/21, 12:51 pm
to search
ENG

Step14:



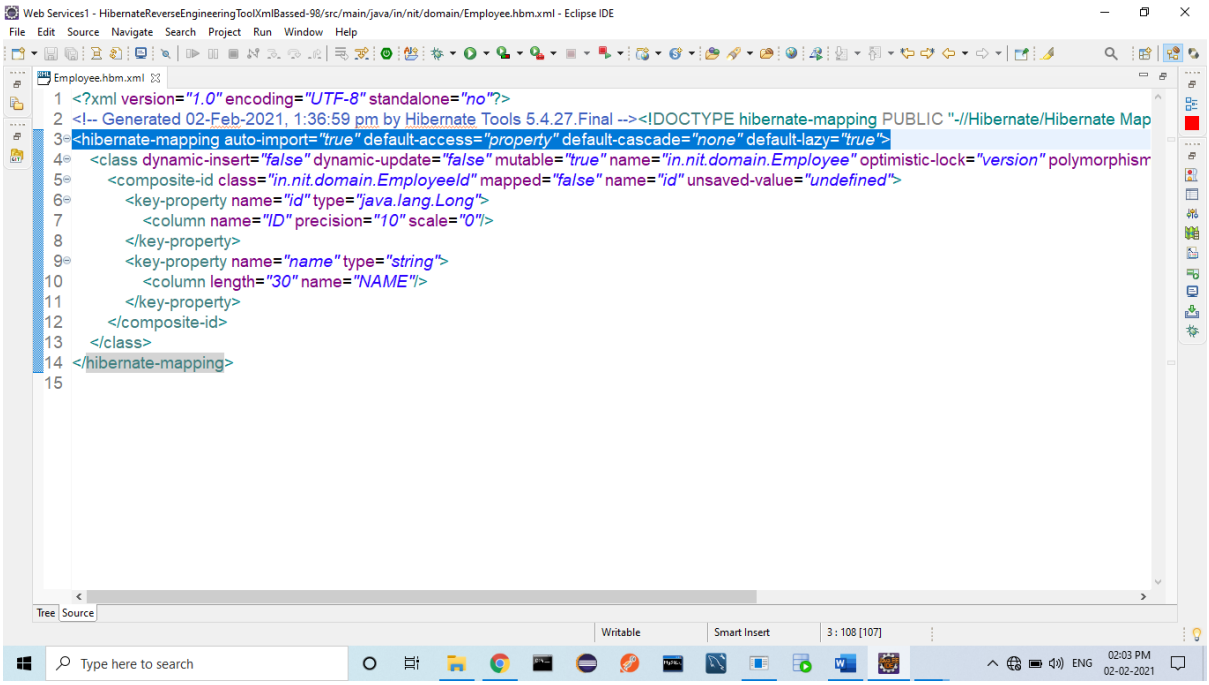
Step15:

If any extra configurations coming delete it



Step16:

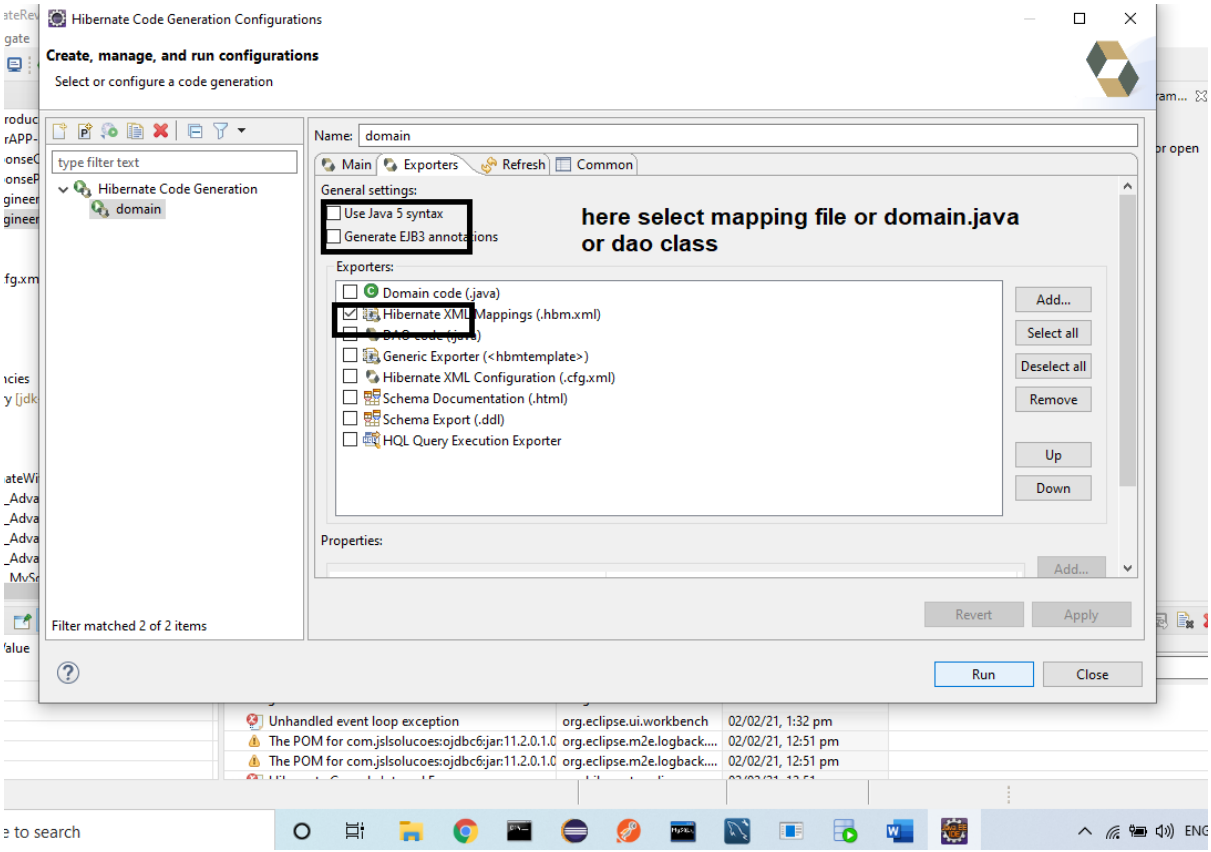
Mapping file code:



```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!-- Generated 02-Feb-2021, 1:36:59 pm by Hibernate Tools
5.4.27.Final --><!DOCTYPE hibernate-mapping PUBLIC "-
//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<hibernate-mapping auto-import="true" default-access="property"
default-cascade="none" default-lazy="true">
    <class dynamic-insert="false" dynamic-update="false" mutable="true"
name="in.nit.domain.Employee" optimistic-lock="version"
polymorphism="implicit" schema="SCOTT" select-before-update="false"
table="EMPLOYEE">
        <composite-id class="in.nit.domain.EmployeeId" mapped="false"
name="id" unsaved-value="undefined">
            <key-property name="id" type="java.lang.Long">
                <column name="ID" precision="10" scale="0"/>
            </key-property>
            <key-property name="name" type="string">
                <column length="30" name="NAME"/>
            </key-property>
        </composite-id>
    </class>
</hibernate-mapping>

Mapping file generation completed
```

Note: Here you can select domain class, mapping file, dao class Etc……..



Sample Example:

Step in pom.xml file:

```
<properties>
        <project.build.sourceEncoding>UTF-
8</project.build.sourceEncoding>
        <maven.compiler.source>15</maven.compiler.source>
        <maven.compiler.target>15</maven.compiler.target>
    </properties>

    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.11</version>
            <scope>test</scope>
        </dependency>
        <!--
https://mvnrepository.com/artifact/org.hibernate/hibernate-core -->
        <dependency>
```

```
            <groupId>org.hibernate</groupId>
            <artifactId>hibernate-core</artifactId>
            <version>5.4.20.Final</version>
        </dependency>
        <!--
https://mvnrepository.com/artifact/com.jslsolucoes/ojdbc6 -->
        <dependency>
            <groupId>com.jslsolucoes</groupId>
            <artifactId>ojdbc6</artifactId>
            <version>11.2.0.1.0</version>
        </dependency>
        <!--
https://mvnrepository.com/artifact/org.projectlombok/lombok -->
        <dependency>
            <groupId>org.projectlombok</groupId>
            <artifactId>lombok</artifactId>
            <version>1.18.18</version>
            <scope>provided</scope>
        </dependency>

        </dependencies>
```

Step2:

```
SQL> create table employee(id number(10),name varchar2(30));

Table created.

SQL> commit;

Commit complete.

SQL> select * from employee;

        ID NAME
---------- ------------------------------
   2344478 Sankar

SQL> desc employee;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                                 NUMBER(10)
 NAME                                               VARCHAR2(30)

SQL> _
```

Step3: Create all Required packages:

Step4:

Configuration file generated

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
                "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
                "http://www.hibernate.org/dtd/hibernate-configuration-
3.0.dtd">
<hibernate-configuration>
   <session-factory>
      <property
name="hibernate.connection.driver_class">oracle.jdbc.OracleDriver</pr
operty>
      <property name="hibernate.connection.password">tiger</property>
      <property
name="hibernate.connection.url">jdbc:oracle:thin:@localhost:1521:ORC
L</property>
      <property
name="hibernate.connection.username">scott</property>
      <property
name="hibernate.dialect">org.hibernate.dialect.Oracle10gDialect</prope
rty>
      <mapping class="in.nit.domain.Employee"></mapping>
   </session-factory>
</hibernate-configuration>
```

Step5:

Domain class generated
Note : if any extra classes and any configurations coming delete it.

```
package in.nit.domain;

import javax.persistence.Entity;
import javax.persistence.Id;

@Entity
public class Employee implements java.io.Serializable {
   @Id
```

```
        private Integer id;
        private String name;

        public Integer getId() {
            return this.id;
        }

        public void setId(Integer id) {
            this.id = id;
        }

        public String getName() {
            return this.name;
        }

        public void setName(String name) {
            this.name = name;
        }



    }
```

Step6:

Dao class:

Interface:

```
package in.nit.dao;

import in.nit.domain.Employee;

public interface EmployeeDao {
public int save(Employee emp);
}
```

Impl:

```
package in.nit.dao;

import org.hibernate.Session;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

import in.nit.domain.Employee;

public class EmployeeDaoImpl implements EmployeeDao {

	@Override
	public int save(Employee emp) {
		Transaction tx=null;
		int id=0;
		//create hibernate SessionFactory, Session Objects
		Session ses=new
Configuration().configure("/in/nit/cfgs/hibernate.cfg.xml").buildSessi
onFactory().openSession();
		tx=ses.beginTransaction();

		//Save Object
		try {
			id=(Integer)ses.save(emp);
			tx.commit();
		}
		catch(Exception e) {
			tx.rollback();
			e.printStackTrace();
		}
		return id;
	}

}
```

Factory:

```
package in.nit.dao;

public class EmployeeDaoFactory {
```

```
        public static EmployeeDao getInstane() {

                return new EmployeeDaoImpl();
        }
}
```

Step7:

SaveTest

```
package in.nit.test;

import in.nit.dao.EmployeeDao;
import in.nit.dao.EmployeeDaoFactory;
import in.nit.domain.Employee;

public class SaveTest {

        public static void main(String[] args) {
                EmployeeDao dao=null;
                Employee emp=null;
                //get dao
                dao=EmployeeDaoFactory.getInstane();
      //invoke method
                emp=new Employee();
                emp.setId(2344478);
                emp.setName("Sankar");
                int eno=dao.save(emp);
                System.out.println("Record is Saved in Database"
+eno);
        }

}
```

Output is:

Record is Saved in Database 34445

----------The End----------