Batch Google Classroom
SpringHibernate_11May21_2PMto4PM

Batch WhatsApp Group
https://chat.whatsapp.com/LdhFmOTRjow2NiSTmedNiA

Batch Link
meet.google.com/atv-jmfx-crd

____

Day1

Advanced Java
Book1-WCD(Web Component Developer)
JDBC
Servlet
JSP
XML

Adv Java Book2

Module:Hibernate & Spring Frameworks[Adv Java Book2]
Duration:40 hrs (20 days)

Today we starting with module Hibernate

Hibernate is a Framework??

Framework is semi-developed code which can be customized to develop our application faster

we dont to develop our application from scratch

we acheive RAD[Rapid Application Development]

****Hibernate is a Database Framework
It saves an object into DB
Employee emp=new Employee(101,'Robert',23000);
Here emp object is stored in heap(temporarily stored)

Object Persistence
----
we want to store object state permanently
Employee emp=new Employee(101,"RObert",78900);
session.save(emp);
we are trying save an object in to the database

**Hibernate is a Framework used for Object Persistence in the database.

Object Persistence Mechanisms
----
1)JDBC:-
  -----
application connects to the db via DriverManager/

DataSource[Connection Pooling]

appln fires query using Statement,PreparedStatement,
CallableStatement(stored procedure)

To retrive data ResultSet/RowSet is used

Pros:-
simple to code

Cons:-
-repeated coding
-transaction & concurrency needs to be done manually
-caching & dynamic update require extra efforts

```
Connection con=DriverManager.getConnection
(dburl,dbuser,dbpass);
```

Here conn object will be created & then returned

```
con.close();//here conn object gets destroyed
```

```
DataSource ds=ctx.lookup("myds");
Connection con=ds.getConnection();
```

Here precreated conn object is returned

```
con.close();//here conn is object is added back to the free pool
```

PS executes dynamic query(query which accepts runtime input)
insert into emp values (?,?,?) --->placeholder

PreparedStatement executes precompiled queries
frequently required static query is compiled only once & native code
is cached & reused.Hence PS offer better performance than statement

"select * from emp"(query)---->native db lang

To fetch data frm ResultSet--->3 objects
Connection,Statement,ResultSet

To fetch data using RowSet--->only Rowset(conn,fire query,data retrive
data)

Rowset
1)3-in-1(conn,fire query,data retrive data)
2)Scrollable and updatable is by default
3)supports disconnected architecture

```
Rowset 2 types --->JdbcRowSet (connected architecture)
                   CachedRowSet(disconnected architecture)
```

____

2)Serialization

```
  --------------
  Serialization is process of converting object state into    a
bytestream.
  This bytestream can then be stored into file/transfer   across
network

Pros:-
        automated object peristence in a file
[
we just need to pass object to a method writeObject()
class need to implements Serializable
]
Cons:-
  object--->bytestream---->file [slow]
  hence not suitable for large scale data

//Marker or Tagging interface
public class Employee implements Serializable{
        int empId;
        String empName;
        double salary;
        Address resAddr;

}
public class Address implements Serializable{
        String street,city,state,country;
}

Employee emp=new Employee(101,"RObert",78900,new Address(...));
FileOutputStream fos=new FileOutputStream("emp.txt");
ObjectOutputStream oos=new ObjectOutputStream(fos);
oos.writeObject(emp);
-----------------------------------------------------------
What is a marker/tagging interface?
It is an empty interface,which does not have any method
whenever any class implements it,the class need not impl any method
MI is used to grant a special permission/feature to its implementing
class
_______________________________________________________

3)XML:- Extensible Markup language
  ----
we can store an object in to an xml file using XMLStreamWriter

file---->sequential manner
db------>tabular manner
xml----->structured manner

emp.dat
-------
101 Robert 78900......

employee.xml
------------
<Employee>
        <id>101</id>
        <name>RObert</name>
        <salary>78900</salary>
</Employee>
```

Pros:-
stores structured data in a text format[hence best suited for transferring data between heterogeneous systems]

Cons:-

        Not good for large scale data

____