Dt : 12/5/2022

Connection Pooling in JDBC:

=>The process of organizing multiple pre-initialized DataBase Connections among multiple users is known as 'Connection pooling process'.

Behaviour:

(i)User picks the DataBase connection from the Pool.

(ii)User uses the connection to perform operations on DataBase

(iii)After usage returns the Connection back to the pool.

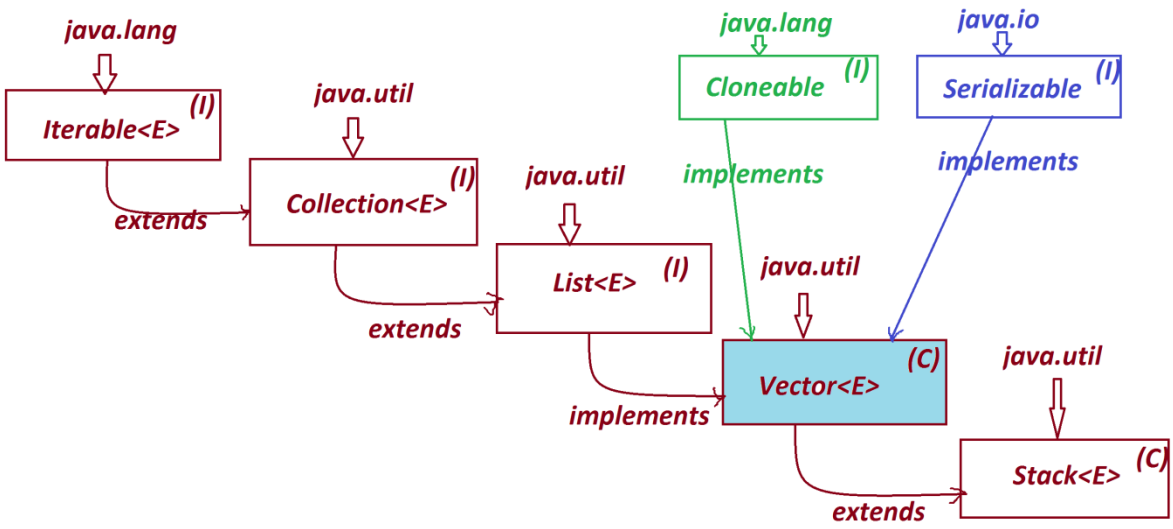
Note:

=>we take the support of Vector<E> to construct Connection Pooling Process.

=>This Vector<E> organizes elements in Sequence.

=>Vector<E> is synchronized class and Thread-Safe class.

Hierarchy of Vector<E>:



Ex_program:

ConnectionPool.java

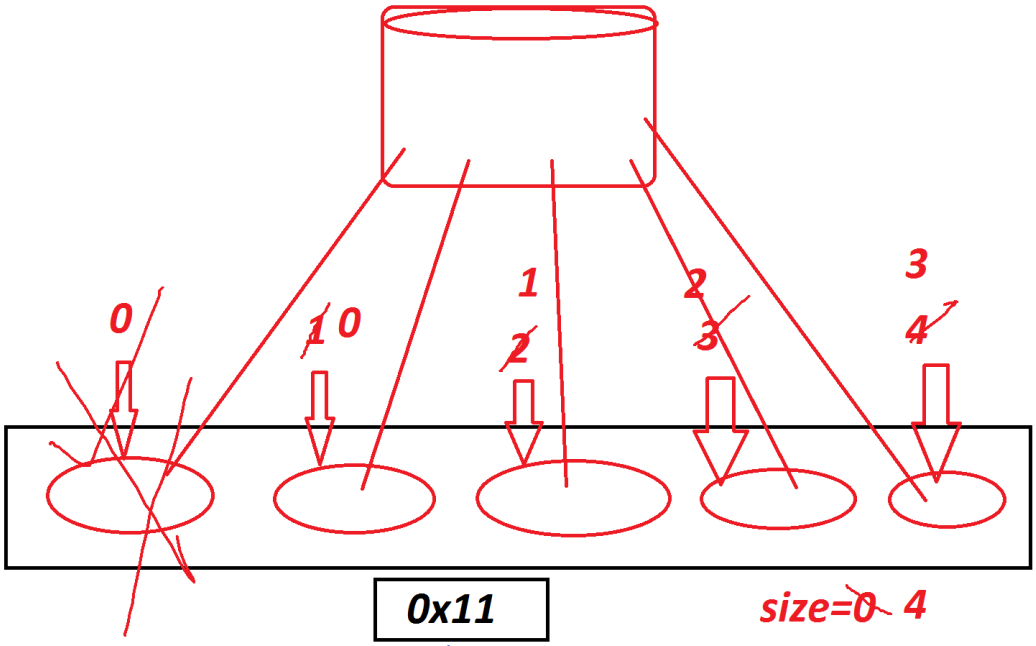
```
package test;
import java.sql.*;
import java.util.*;
public class ConnectionPool {
   public String url,uname,pword;
   public ConnectionPool(String url,String uname,String pword) {
         this.url=url;
         this.uname=uname;
         this.pword=pword;
   }
   public Vector<Connection> v = new Vector<Connection>();
   public void createConnections()
   {
```

```
        try {
                while(v.size()<5)
                {
                        System.out.println("Connection pool is Not-Full...");
                        Connection con=DriverManager.getConnection
                                        (url,uname,pword);
                        v.add(con);//Adding the connection to Pool
                        System.out.println(con);
                }//end of loop
                if(v.size()==5)
                {
                        System.out.println("Connection Pool is full...");
                }
        }catch(Exception e) {e.printStackTrace();}
}//end of method
public synchronized Connection useConnection()
{
        Connection con = v.firstElement();//First Connection is taken
        v.remove(0);//one Connection at index 0 is deleted
        return con;
}//end of method
public synchronized void returnConnection(Connection con)
{
        v.addElement(con);//Adding Connection back to pool
```

```
        System.out.println("Connection returned back to the pool...");;

  }

}
```

DBCon12.java

```
package test;
import java.util.*;
import java.sql.*;
public class DBCon12 {
     public static void main(String[] args) {
       ConnectionPool cp = new ConnectionPool

("jdbc:oracle:thin:@localhost:1521:xe","system","manager");
       cp.createConnections();
       System.out.println("Pool Size:"+cp.v.size());
       Connection con = cp.useConnection();
       System.out.println("User using : "+con);
       System.out.println("Pool Size:"+cp.v.size());
       cp.returnConnection(con);
       System.out.println("Pool Size:"+cp.v.size());
       System.out.println("====Display the Connection from
Pool====");
       cp.v.forEach((k)->
       {
        System.out.println(k);
       });
     }
}
```



Vector<Connection> v = new Vector<Connection>();

This ref will hold UnLimited Connection Objects