Q) What is functionality of getConnection() method?

A.

Syntax:-

Public static Connection getConnection(String dburl,String uname,String pwd);

This method present in the DriverManager class.

Based on “jdbc:oracle:thin” of DB url, The driver manager pick up registered “jdbc oracle driver”.

1. Based on “localhost:1521:xe” of DB url , The driver manager identifies the logical db of oracle s/w of current machine. It establish the connection between oracle s/w and java application.
2. This method returns either “object of implementation class of Connection Interface” or null.

Q) How do u know Implementation class Name of Connection Interface?

A)

Connection con= DriverManager.getConnection(“dburl”,”dbname”,”pwd”);

String s= con.getClass();

S.o.p(s)

O/p:-

**T4CConnection**.

**Note:-** This class name is changed from driver s/w to driver s/w. Therefore Remembering all class names is difficult. Therefore It is better to refer the implementation class object with Connection type variable.

**Real life Example:**- Instead remembering /referring all students with their original names. The faculty refers them with common name “My Dear students”.

===== Traditional Method-1 to register the JDBC Driver S/w with DriverManager ===== =

Oracle.jdbc.driver.OracleDriver driver = new oracle.jdbc.driver.OracleDriver();

DriverManager.registerDriver(driver);

=====Traditional Method-1 to register the JDBC Driver S/w with DriverManager =====

Class.forName(“oracle.jdbc.driver.OracleDriver”);

forName() method make JVM to load the OracleDriver class. In that loading process, the static block of OracleDriver class is executed. This block contains logic to create object of OracleDriver and to place object at DriverManager by invoke the registerDriver() method.

Static{

Try{

If(defaultDriver == null){

defaultDriver=new oracle.jdbc.driver.OracleDriver();

DriverManager.registerDriver(defaultDriver);

}

}

Catch(Exception e) {

----

}

}

=====Modern Approach to register the JDBC Driver S/w with DriverManager =====

JDBC 4.x onwards, “The auto loading of jdbc driver class feature” is given. Just we have to add jdbc driver software related jar file to “CLASSPATH” variable. “jdbc driver class” will be automatically loaded.

Try{

Connection con=DriverManager.getConnection(“DBurl”,”uname”,”pwd”);

If(con==null){

System.out.println(“Connection not established”);

}

Else{

System.out.println(“Connection established”);

}

}

When DriverManager.getConnection() is invoked,

Step1: DriverManager Class is loaded into JVM.

Step2: In the process of loading, The static block of DriverManager is executed. This block contains logic to search for “java.sql.Driver file” in META-INF\search folder of jar file.

Step3: From this file, It collects JDBC driver class name.

Step4: It invokes the Class.forName() for loading JDBC Driver class.

Step5: In the process of Loading JDBC Driver class, the static block of JDBC Driver class is executed . This block contains logic for registering JDBC Driver s/w with DriverManager.

