Dt : 2/1/2024(day-4)

*imp

Making ComputerSystem environment ready to execute JDBC Appications:

step-1 : Download and Install Database product(Oracle database)

step-2 : Perform Login Process

step-3 : Create table with name Customer70

 (id,name,city,mailid,phno)

  primary key : id

create table Customer70(id number(10),name varchar2(15),city varchar2(15),

mailid varchar2(25),phno number(15),primary key(id));

step-4 : Insert min 5 Customer details

insert into Customer70 values(1234,'Alex','Hyd','alex@gmail.com',9898981234);

insert into Customer70 values(2312,'Ram','Hyd','ram@gmail.com',7676761234);

insert into Customer70 values(3212,'Raj','Hyd','raj@gmail.com',4343431234);

step-4 : Copy DB-JAR-File from "lib" folder of Oracle to User defined folder on DeskTop

 =>DB-Jar_File is available in the following path:

   C:\oraclexe\app\oracle\product\11.2.0\server\jdbc\lib

        ojdbc6.jar - Oracle11

faq:

define JAR file?

  =>JAR srands for 'Java Archieve' and which is compressed format of more number of class

   files.

step-5 : Find PortNo and Service Name of Database product(Oracle Product)

  =>PortNo and Service Name available from 'tnsnames.ora' file of 'ADMIN" folder of 'network'

     C:\oraclexe\app\oracle\product\11.2.0\server\network\ADMIN\tnsnames.ora

        Port No   :   1521

        Service Name :   XE

  --

*imp

JDBC API:

  =>'java.sql' package is known as 'JDBC API' and which provide some 'classes and interfaces'

    to construct JDBC-Applications.

  =>'java.sql.Connection' interface is the root of JDBC-API and which is Normal Interface.

  =>The following are some important methods of 'Connection' interface:

     1.createStatement()

     2.prepareStatement()

     3.prepareCall()

     4.getAutoCommit()

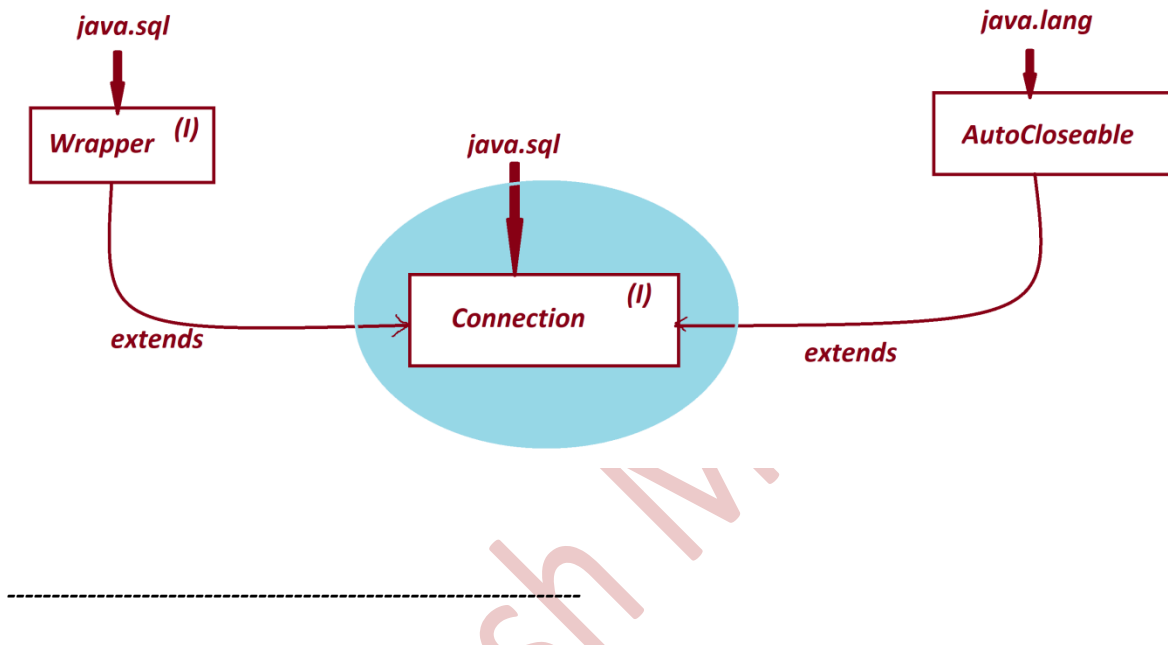   *5.setAutoCommit()*

     6.setSavepoint()

     7.releaseSavepoint()

     8.commit()

9.rollback()

10.close()

Hierarchy of 'Connection' interface:

java.sql   *java.lang*

Wrapper *(I)*   *java.sql*   **AutoCloseable**

Connection *(I)*

extends   *extends*

Lab-Assignment

ITest(I)

abstract method   *void m1(int x);*

abstract method   *void m2(int y);*

default concrete method   *void m3(int z)*
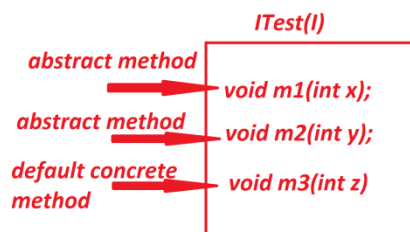
Program-1 : Create implementation object using Implementation class

Program-2 : Create implementation Object using
                'Anonymous Local Inner as implementation class of Interface'

Variables   Methods   Constructors   Blocks   Classes   Interfaces   AbstractClasses
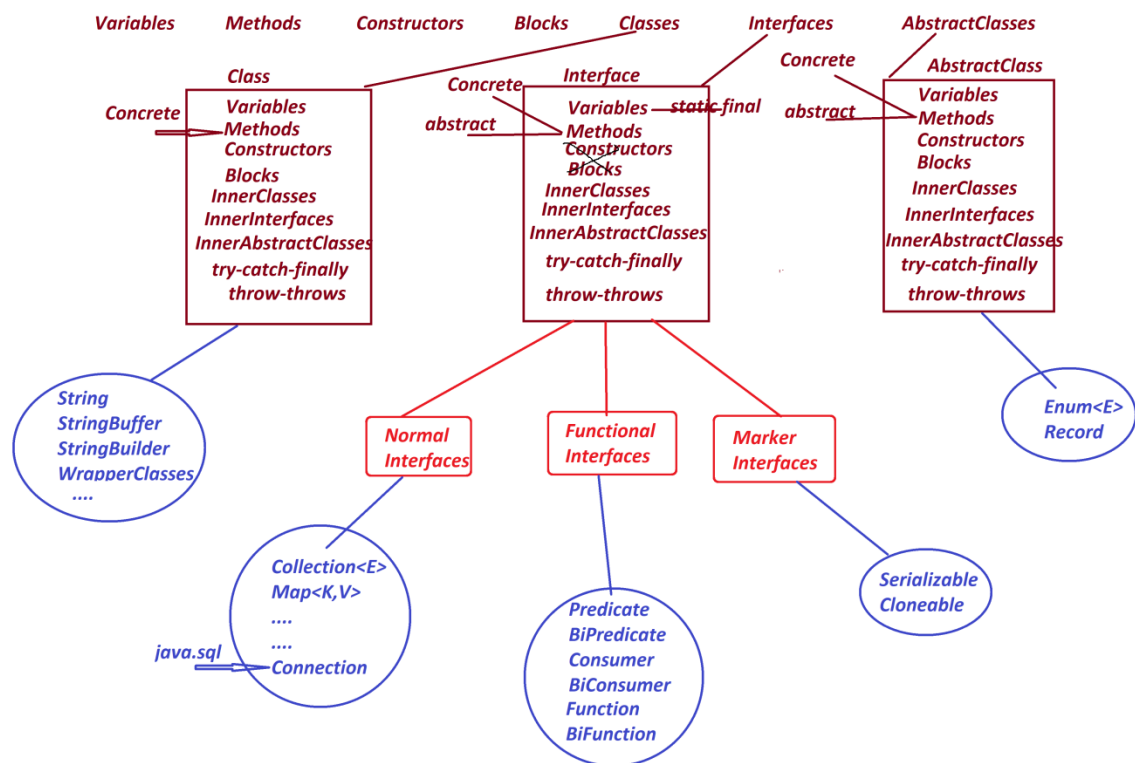
Class

Concrete   *Concrete*   **Interface**   Concrete   **AbstractClass**

Variables
Methods
Constructors
Blocks
InnerClasses
InnerInterfaces
InnerAbstractClasses
try-catch-finally
throw-throws

abstract   *Variables*   static final
Methods
Constructors
Blocks
InnerClasses
InnerInterfaces
InnerAbstractClasses
try-catch-finally
throw-throws

abstract   *Variables*
Methods
Constructors
Blocks
InnerClasses
InnerInterfaces
InnerAbstractClasses
try-catch-finally
throw-throws

String
StringBuffer
StringBuilder
WrapperClasses
....

Normal Interfaces   *Functional Interfaces*   *Marker Interfaces*

Enum<E>
Record

Collection<E>
Map<K,V>
....
....
java.sql   *Connection*

Predicate
BiPredicate
Consumer
BiConsumer
Function
BiFunction

Serializable
Cloneable