Quiz-3(OOP with Java) SET-A

Time: 30 Minutes   **Max Marks: 10**
Multiple Choices [0.25X16]

Q1. Which class object is used for dynamic queries in JDBC API
        A. Statement   B. Connection   C. PreparedStatement   D. Forname

Q2. All collection classes are available in which package
        A. java.io   B. java.lang   C. java.awt   D. java.util

Q3. Which method is used to perform DDL statements in JDBC
        A. execute()   B. executeUpdate()   C. executeQuery()   D. None of Them

Q4. Which one is called as JDBC-Net driver in JDBC
        A. Type-1 driver   B. Type-2 driver   C. Type-3 driver   D. Type-4 driver

Q5. You need to store elements in a collection that guarantees that no duplicates are stored and all elements can be accessed in natural order. Which class provides that capability?
        A.java.util.HashMap   B.java.util.TreeSet   C.java.util.ArrayList
        D.java.util.Collections

Q6. Which interface provides the capability to store objects using a key-value pair?
        A.   Java.util.Map   B.   Java.util.Set   C.   Java.util.List   D.   Java.util.Collection

Q7. Which of these packages contains all the classes and methods required for even handling in Java?
        A.   java.applet   B.   java.awt   C.   java.event   D.   java.awt.event

Q8. Which of these methods are used to register a keyboard event listener?
        a) KeyListener()   b) addKistener()   c) addKeyListener()   d) eventKeyboardListener()

 Q9. What is a listener in context to event handling?
        a) A listener is a variable that is notified when an event occurs.
        b) A listener is a object that is notified when an event occurs.
        c) A listener is a method that is notified when an event occurs.
        d) None of the mentioned.

Q10. Which of these events is generated when a button is pressed?
        a) ActionEvent   b) KeyEvent   c) WindowEvent   d) AdjustmentEvent

Q11. Which of these methods of Thread class is used to find out the priority given to a thread?
        a) get()   b) ThreadPriority()   c) getPriority()   d) getThreadPriority()
Q12. Which method registers a thread in a thread scheduler?
        a)   run();   b)   construct();   c)   start();   d)   register();
Q13. What is multithreaded programming?
        a) It's a process in which two different processes run simultaneously.
        b) It's a process in which two or more parts of same process run simultaneously.
        c) Its a process in which many different process are able to access same information.
        d) Its a process in which a single process can access information from many sources.

Q14. What is sent to the user via HTTP, invoked using the HTTP protocol on the user's computer, and run on the user's computer as an application?

    A. A Java application   B.   A Java applet   C.   A Java servlet   D. None of them.

Q15. Which file stream is serialized before writing to file.

    A. FileReader   B. FileWriter   C. ObjectOutputStream   D. None of them.

Q16. Which file stream can be used to open a file to read strings

    A.   FileReader   B. BufferedWriter   C. BufferedReader   D. None of them.

Correct the error if any and find the output of the following program segments also show workout   [1.5X4]

Q1.	Q2.
| ```java
class multithreaded_programing {
    public static void main(String args[]) {
        Thread t = Thread.currentThread();
        System.out.println(t.getPriority());
    }
}
``` | ```java
import java.util.*;
  class Output {
     public static void main(String args[]) {
        ArrayList obj = new ArrayList();
        obj.add("A");
        obj.add(0, "B");
        System.out.println(obj.size());
     }
  }
``` |
| Q3. | Q4. |
| ```java
class MyThread extends Thread
{
  public static void main(String [] args)
  {
    MyThread t = new MyThread();
    t.run();
  }
  public void run()
  {
    for(int i=1; i < 3; ++i)
    {
      System.out.print(i + "..");
    }
  }
}
``` | ```java
import java.util.*;
  class Linkedlist {
     public static void main(String args[]) {
        LinkedList obj = new LinkedList();
        obj.add("A");
        obj.add("B");
        obj.add("C");
        obj.addFirst("D");
        System.out.println(obj);
     }
  }
``` |