

1Z0-809 Dumps

Java SE 8 Programmer II

<https://www.certleader.com/1Z0-809-dumps.html>



NEW QUESTION 1

Given:

```
class Book { int id;
String name;
public Book (int id, String name) { this.id = id;
this.name = name;
}
public boolean equals (Object obj) { //line n1 boolean output = false;
Book b = (Book) obj;
if (this.name.equals(b.name)) output = true;
}
return output;
}
}
```

and the code fragment:

```
Book b1 = new Book (101, "Java Programming"); Book b2 = new Book (102, "Java Programming"); System.out.println (b1.equals(b2)); //line n2 Which statement is true?
```

- A. The program prints true.
- B. The program prints false.
- C. A compilation error occur
- D. To ensure successful compilation, replace line n1 with: boolean equals (Book obj) {
- E. A compilation error occur
- F. To ensure successful compilation, replace line n2 with: System.out.println (b1.equals((Object) b2));

Answer: A

NEW QUESTION 2

Given the code fragment:

```
5. IntConsumer consumer = e -> System.out.println(e);
6. Integer value = 90;
7. /* insert code fragment here */
8. consumer.accept(result);
```

Which code fragment, when inserted at line 7, enables printing 100?

- A. Function<Integer> funRef = e -> e + 10; Integer result = funRef.apply(value);
- B. IntFunction funRef = e -> e + 10; Integer result = funRef.apply (10);
- C. ToIntFunction<Integer> funRef = e -> e + 10; int result = funRef.applyAsInt (value);
- D. ToIntFunction funRef = e -> e + 10; int result = funRef.apply (value);

Answer: A

NEW QUESTION 3

Given the code fragment:

```
Stream<Path> files = Files.walk(Paths.get(System.getProperty("user.home"))); files.forEach (fName -> { //line n1
try {
Path aPath = fName.toAbsolutePath(); //line n2 System.out.println(fName + ":"
+ Files.readAttributes(aPath, Basic.File.Attributes.class).creationTime ());
} catch (IOException ex) { ex.printStackTrace();
});
});
```

What is the result?

- A. All files and directories under the home directory are listed along with their attributes.
- B. A compilation error occurs at line n1.
- C. The files in the home directory are listed along with their attributes.
- D. A compilation error occurs at line n2.

Answer: A

NEW QUESTION 4

Given the code fragment:

```
List<String> codes = Arrays.asList ("DOC", "MPEG", "JPEG"); codes.forEach (c -> System.out.print(c + " "));
String fmt = codes.stream()
.filter (s-> s.contains ("PEG"))
.reduce((s, t) -> s + t).get(); System.out.println("\n" + fmt); What is the result?
```

- A. DOC MPEG JPEG MPEGJPEG
- B. DOC MPEG MPEGJPEG MPEGMPEGJPEG
- C. MPEGJPEG MPEGJPEG
- D. The order of the output is unpredictable.

Answer: A

NEW QUESTION 5

Which class definition compiles?

```

A. class Vehicle {
    int id;
    public void start() {
        public class Engine { int eNo = id; }
    }
}

B. class Computer {
    private Card sCard = new SoundCard();
    private abstract class Card { }
    private class SoundCard extends Card { }
}

C. class Block {
    int bno;
    static class Counter {
        int locator;
        Counter() { locator = bno; }
    }
}

D. class Product {
    interface Moveable { void move(); }
    Moveable mProduct = new Moveable() {
        void move() { }
    };
}

```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

NEW QUESTION 6

Given the code fragment:

```

Path file = Paths.get ("courses.txt");
// line n1

```

Assume the courses.txt is accessible.

Which code fragment can be inserted at line n1 to enable the code to print the content of the courses.txt file?

- A. List<String> fc = Files.list(file); fc.stream().forEach (s -> System.out.println(s));
- B. Stream<String> fc = Files.readAllLines (file); fc.forEach (s -> System.out.println(s));
- C. List<String> fc = readAllLines(file); fc.stream().forEach (s -> System.out.println(s));
- D. Stream<String> fc = Files.lines (file); fc.forEach (s -> System.out.println(s));

Answer: D

NEW QUESTION 7

Given:

```
public class Foo {
    public void methodB(String s) { System.out.println("Foo " + s ); }
}

public class Bar extends Foo {
    public void methodB(String s) { System.out.println("Bar " + s); }
}

public class Baz extends Bar {
    public void methodB(String s) { System.out.println("Baz " + s); }
}

public class Daze extends Baz{
    private Bar bb = new Bar();
    public void methodB(String s) {
        bb.methodB(s);
        super.methodB(s);
    }
}

public class TestClass {
    public static void main(String[] args) {
        Baz d = new Daze();
        d.methodB("Hello");
    }
}
```

What is the result?

- A. Bar Hello Foo Hello
- B. Bar Hello Baz Hello
- C. Baz Hello
- D. A compilation error occurs in the Daze class.

Answer: C

NEW QUESTION 8

Which two statements are true about synchronization and locks? (Choose two.)

- A. A thread automatically acquires the intrinsic lock on a synchronized statement when executed.
- B. The intrinsic lock will be retained by a thread if return from a synchronized method is caused by an uncaught exception.
- C. A thread exclusively owns the intrinsic lock of an object between the time it acquires the lock and the time it releases it.
- D. A thread automatically acquires the intrinsic lock on a synchronized method's object when entering that method.
- E. Threads cannot acquire intrinsic locks on classes.

Answer: AB

NEW QUESTION 9

Given the code fragment:

```
List<String> valList = Arrays.asList("", "George", "", "John", "Jim");
Long newVal = valList.stream() // line n1
    .filter(x -> !x.isEmpty())
    .count(); // line n2
System.out.print(newVal);
```

What is the result?

- A. A compilation error occurs at line n2.
- B. 3
- C. 2
- D. A compilation error occurs at line n1.

Answer: A

NEW QUESTION 10

Given the code fragment:

```
List<String> listVal = Arrays.asList("Joe", "Paul", "Alice", "Tom"); System.out.println (
```

// line n1

);

Which code fragment, when inserted at line n1, enables the code to print the count of string elements whose length is greater than three?

- A. listVal.stream().filter(x -> x.length()>3).count()
- B. listVal.stream().map(x -> x.length()>3).count()
- C. listVal.stream().peek(x -> x.length()>3).count().get()
- D. listVal.stream().filter(x -> x.length()>3).mapToInt(x -> x).count()

Answer: A

NEW QUESTION 10

Given:

```
public class Foo<K, V> {
    private K key;
    private V value;

    public Foo(K key, V value) { this.key = key; this.value = value; }

    public static <T> Foo<T, T> twice(T value) { return new Foo<T, T>(value, value); }

    public K getKey() { return key; }
    public V getValue() { return value; }
}
```

Which option fails?

- A. Foo<String, Integer> mark = new Foo<String, Integer> ("Steve", 100);
- B. Foo<String, String> pair = Foo.<String>twice ("Hello World!");
- C. Foo<Object, Object> percentage = new Foo<String, Integer>("Steve", 100);
- D. Foo<String, String> grade = new Foo <> ("John", "A");

Answer: A

NEW QUESTION 11

Given:

IntStream stream = IntStream.of (1,2,3); IntFunction<Integer> inFu = x -> y -> x*y; //line n1

IntStream newStream = stream.map(inFu.apply(10)); //line n2 newStream.forEach(System.out::print);

Which modification enables the code fragment to compile?

- A. Replace line n1 with: IntFunction<UnaryOperator> inFu = x -> y -> x*y;
- B. Replace line n1 with: IntFunction<IntUnaryOperator> inFu = x -> y -> x*y;
- C. Replace line n1 with: BiFunction<IntUnaryOperator> inFu = x -> y -> x*y;
- D. Replace line n2 with: IntStream newStream = stream.map(inFu.applyAsInt (10));

Answer: B

NEW QUESTION 14

Given:

```
public class Customer { private String fName; private String lName; private static int count;
```

```
public customer (String first, String last) {fName = first, lName = last;
```

```
++count;}
```

```
static { count = 0; }
```

```
public static int getCount() {return count; }
```

```
}
```

```
public class App {
```

```
public static void main (String [] args) { Customer c1 = new Customer("Larry", "Smith");
```

```
Customer c2 = new Customer("Pedro", "Gonzales"); Customer c3 = new Customer("Penny", "Jones"); Customer c4 = new Customer("Lars", "Svenson"); c4 =
```

```
null;
```

```
c3 = c2;
```

```
System.out.println (Customer.getCount());
```

```
}
```

```
}
```

What is the result?

- A. 2
- B. 3
- C. 4
- D. 5

Answer: D

NEW QUESTION 16

Given the code fragment:

```
public void recDelete (String dirName) throws IOException { File [ ] listOfFiles = new File (dirName) .listFiles();
```

```
if (listOfFiles != null && listOfFiles.length >0) {
```

```
for (File aFile : listOfFiles) { if (aFile.isDirectory ()) {
recDelete (aFile.getAbsolutePath ());
} else {
if (aFile.getName ().endsWith (".class")) aFile.delete ();
}
}
}
```

Assume that Projects contains subdirectories that contain .class files and is passed as an argument to the recDelete () method when it is invoked. What is the result?

- A. The method deletes all the .class files in the Projects directory and its subdirectories.
- B. The method deletes the .class files of the Projects directory only.
- C. The method executes and does not make any changes to the Projects directory.
- D. The method throws an IOException.

Answer: A

NEW QUESTION 21

Given:

```
1. abstract class Shape {
2. Shape () { System.out.println ("Shape"); }
3. protected void area () { System.out.println ("Shape"); } 4. }
5.
6. class Square extends Shape {
7. int side;
8. Square int side {
9. /* insert code here */
10. this.side = side;
11. }
12. public void area () { System.out.println ("Square"); }
13. }
14. class Rectangle extends Square {
15. int len, br;
16. Rectangle (int x, int y) {
17. /* insert code here */
18. len = x, br = y;
19. }
20. void area () { System.out.println ("Rectangle"); }
21. }
```

Which two modifications enable the code to compile? (Choose two.)

- A. At line 1, remove abstract
- B. At line 9, insert super ();
- C. At line 12, remove public
- D. At line 17, insert super (x);
- E. At line 17, insert super (); super.side = x;
- F. At line 20, use public void area () {

Answer: DF

NEW QUESTION 24

Given the definition of the Book class:

```
public class Book {
    private int id;
    private String name;
    public Book(int id, String name) {this.id = id; this.name = name;}
    public int getId() { return id; }
    public String getName() { return name; }
    public void setId(int id) { this.id = id; }
    public void setName(String name) { this.name = name; }
}
```

Which statement is true about the Book class?

- A. It demonstrates encapsulation.
- B. It is defined using the factory design pattern.
- C. It is defined using the singleton design pattern.
- D. It demonstrates polymorphism.
- E. It is an immutable class.

Answer: A

NEW QUESTION 25

Given the structure of the Student table: Student (id INTEGER, name VARCHAR) Given the records from the STUDENT table:

ID	NAME
102	Edwin
103	Edward
103	Edwin

Given the code fragment:

```
Connection conn = DriverManager.getConnection(dbURL, userName, passWord);
Statement st = conn.createStatement();
String query = "DELETE FROM Student WHERE id = 103";
System.out.println("Status: " + st.execute(query));
```

Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the dbURL, userName, and passWord exists. What is the result?

- A. The program prints Status: true and two records are deleted from the Student table.
- B. The program prints Status: false and two records are deleted from the Student table.
- C. A SQLException is thrown at runtime.
- D. The program prints Status: false but the records from the Student table are not deleted.

Answer: B

NEW QUESTION 26

Given the code fragments:

```
interface CourseFilter extends Predicate<String> { public default boolean test (String str) {
return str.equals ("Java");
}
}
```

and

```
List<String> str = Arrays.asList("Java", "Java EE", "Java ME"); Predicate<String> cf1 = s -> s.length() > 3;
Predicate cf2 = new CourseFilter() { //line n1 public boolean test (String s) {
return s.contains ("Java");
}
};
```

```
long c = str.stream()
.filter(cf1)
.f ilter(cf2 //line n2
.count(); System.out.println(c);
```

What is the result?

- A. 2
- B. 3
- C. A compilation error occurs at line n1.
- D. A compilation error occurs at line n2.

Answer: B

NEW QUESTION 27

Given that /green.txt and /colors/yellow.txt are accessible, and the code fragment: Path source = Paths.get("/green.txt);

Path target = Paths.get("/colors/yellow.txt);

Files.move(source, target, StandardCopyOption.ATOMIC_MOVE); Files.delete(source);

Which statement is true?

- A. The green.txt file content is replaced by the yellow.txt file content and the yellow.txt file is deleted.
- B. The yellow.txt file content is replaced by the green.txt file content and an exception is thrown.
- C. The file green.txt is moved to the /colors directory.
- D. A FileAlreadyExistsException is thrown at runtime.

Answer: D

NEW QUESTION 32

Which statement is true about the DriverManager class?

- A. It returns an instance of Connection.
- B. it executes SQL statements against the database.
- C. It only queries metadata of the database.
- D. it is written by different vendors for their specific database.

Answer: A

Explanation:

The DriverManager returns an instance of Doctrine\DBAL\Connection which is a wrapper around the underlying driver connection (which is often a PDO instance).

NEW QUESTION 37

Given the code fragment:

```
//line n1
System.out.println(iP);
```

Which code fragment, when inserted at line n1, enables the code to print /First.txt?

- A. Path iP = new Paths ("/First.txt");
- B. Path iP = Paths.toPath ("/First.txt");
- C. Path iP = new Path ("/First.txt");
- D. Path iP = Paths.get ("/", "First.txt");

Answer: D

NEW QUESTION 38

Given the code fragment:

```
final String str1 = "Java";
StringBuffer strBuf = new StringBuffer("Course");
UnaryOperator<String> u = (str2) -> str1.concat(str2); // line n1
UnaryOperator<String> c = (str3) -> str3.toLowerCase();
System.out.println(u.apply(c.apply(strBuf))); // line n2
```

What is the result?

- A. A compilation error occurs at line n1.
- B. courseJava
- C. Javacourse
- D. A compilation error occurs at line n2.

Answer: A

NEW QUESTION 40

In 2015, daylight saving time in New York, USA, begins on March 8th at 2:00 AM. As a result, 2:00 AM becomes 3:00 AM.

Given the code fragment:

```
ZoneId zone = ZoneId.of("America/New_York");
ZonedDateTime dt = ZonedDateTime.of(LocalDate.of(2015, 3, 8), LocalTime.of(1, 0),
zone);
ZonedDateTime dt2 = dt.plusHours(2);
System.out.print(DateTimeFormatter.ofPattern("H:mm - ").format(dt2));
System.out.println("difference: " + ChronoUnit.HOURS.between(dt, dt2));
```

Which is the result?

- A. 3:00 – difference: 2
- B. 2:00 – difference: 1
- C. 4:00 – difference: 3
- D. 4:00 – difference: 2

Answer: B

NEW QUESTION 44

Given the code fragment:

```
ZonedDateTime depart = ZonedDateTime.of(2015, 1, 15, 3, 0, 0, 0, ZoneID.of("UTC-7"));
```

```
ZonedDateTime arrive = ZonedDateTime.of(2015, 1, 15, 9, 0, 0, 0, ZoneID.of("UTC-5"));
```

```
long hrs = ChronoUnit.HOURS.between(depart, arrive); //line n1 System.out.println("Travel time is" + hrs + "hours");
```

What is the result?

- A. Travel time is 4 hours
- B. Travel time is 6 hours
- C. Travel time is 8 hours
- D. An exception is thrown at line n1.

Answer: A

NEW QUESTION 45

Given the code fragment: Stream<List<String>> iStr= Stream.of (Arrays.asList ("1", "John"), Arrays.asList ("2", null)0; Stream<<String> nInSt = iStr.flatMapToInt ((x) -> x.stream ()); nInSt.forEach (System.out :: print); What is the result?

- A. 1John2null
- B. 12

- C. A NullPointerException is thrown at run time.
D. A compilation error occurs.

Answer: D

NEW QUESTION 47

Given:
class Student {
String course, name, city;
public Student (String name, String course, String city) { this.course = course; this.name = name; this.city = city;
}
public String toString() {
return course + ":" + name + ":" + city;
}
}
and the code fragment: List<Student> stds = Arrays.asList(
new Student ("Jessy", "Java ME", "Chicago"), new Student ("Helen", "Java EE", "Houston"), new Student ("Mark", "Java ME", "Chicago")); stds.stream()
.collect(Collectors.groupingBy(Student::getCourse))
.forEach(src, res) -> System.out.println(src)); What is the result?

- A. [Java EE: Helen:Houston][Java ME: Jessy:Chicago, Java ME: Mark:Chicago]
B. Java EEJava ME
C. [Java ME: Jessy:Chicago, Java ME: Mark:Chicago] [Java EE: Helen:Houston]
D. A compilation error occurs.

Answer: B

NEW QUESTION 52

Given that these files exist and are accessible:

```
/company/emp/info.txt
/company/emp/benefits/b1.txt
```

and given the code fragment:

```
// line n1
stream.forEach(s -> System.out.print(s));
```

Which code fragment can be inserted at line n1 to enable the code to print only /company/emp?

- A. Stream<Path> stream = Files.list (Paths.get ("/company"));
B. Stream<Path> stream = Files.find(Paths.get ("/company"), 1,(p,b) -> b.isDirectory (), FileVisitOption.FOLLOW_LINKS);
C. Stream<Path> stream = Files.walk (Paths.get ("/company"));
D. Stream<Path> stream = Files.list (Paths.get ("/company/emp"));

Answer: B

NEW QUESTION 53

Given:
Item table
• ID, INTEGER: PK
• DESCRIP, VARCHAR(100)
• PRICE, REAL
• QUANTITY< INTEGER

And given the code fragment:
9. try {
10. Connection conn = DriverManager.getConnection(dbURL, username, password);
11. String query = "Select * FROM Item WHERE ID = 110";
12. Statement stmt = conn.createStatement();
13. ResultSet rs = stmt.executeQuery(query);
14. while(rs.next()) {
15. System.out.println("ID: " + rs.getInt("Id"));
16. System.out.println("Description: " + rs.getString("Descrip"));
17. System.out.println("Price: " + rs.getDouble("Price"));
18. System.out.println(Quantity: " + rs.getInt("Quantity"));
19. }
20. } catch (SQLException se) {
21. System.out.println("Error");
22. }

Assume that:

The required database driver is configured in the classpath.
The appropriate database is accessible with the dbURL, userName, and passWord exists. The SQL query is valid.
What is the result?

- A. An exception is thrown at runtime.
B. Compilation fails.
C. The code prints Error.
D. The code prints information about Item 110.

Answer: D

NEW QUESTION 58

You have been asked to create a ResourceBundle which uses a properties file to localize an application. Which code example specifies valid keys of menu1 and menu2 with values of File Menu and View Menu?

- A. <key name = 'menu1'>File Menu</key><key name = 'menu2'>View Menu</key>
- B. <key>menu1</key><value>File Menu</value><key>menu2</key><value>View Menu</value>
- C. menu1, File Menu, menu2, View Menu Menu
- D. menu1 = File Menu menu2 = View Menu

Answer: D

NEW QUESTION 59

Assume customers.txt is accessible and contains multiple lines. Which code fragment prints the contents of the customers.txt file?

- A. Stream<String> stream = Files.find (Paths.get ("customers.txt")); stream.forEach((String c) -> System.out.println(c));
- B. Stream<Path> stream = Files.find (Paths.get ("customers.txt")); stream.forEach(c) -> System.out.println(c));
- C. Stream<Path> stream = Files.list (Paths.get ("customers.txt")); stream.forEach(c) -> System.out.println(c));
- D. Stream<String> lines = Files.lines (Paths.get ("customers.txt")); lines.forEach(c) -> System.out.println(c));

Answer: A

NEW QUESTION 64

Given the code fragment:

```
Map<Integer, Integer> mVal = new HashMap<>();
mVal.put(1, 10);
mVal.put(2, 20);
//line n1
c.accept(1, 2);
mVal.forEach(c);
```

Which statement can be inserted into line n1 to print 1,2; 1,10; 2,20;?

- A. BiConsumer<Integer,Integer> c = (i, j) -> {System.out.print (i + "," + j+ " ");};
- B. BiFunction<Integer, Integer, String> c = (i, j) -> {System.out.print (i + "," + j+ " ");};
- C. BiConsumer<Integer, Integer, String> c = (i, j) -> {System.out.print (i + "," + j+ " ");};
- D. BiConsumer<Integer, Integer, Integer> c = (i, j) -> {System.out.print (i + "," + j+ " ");};

Answer: B

NEW QUESTION 65

The data.doc, data.txt and data.xml files are accessible and contain text. Given the code fragment:

```
Stream<Path> paths = Stream.of (Paths. get("data.doc"),
Paths. get("data.txt"),
Paths. get("data.xml"));
paths.filter(s-> s.toString().endsWith("txt")).forEach( s -> {
try { Files.readAllLines(s)
.stream()
.forEach(System.out::println); //line n1
} catch (IOException e) { System.out.println("Exception");
}
});
```

What is the result?

- A. The program prints the content of data.txt file.
- B. The program prints: Exception<<The content of the data.txt file>> Exception
- C. A compilation error occurs at line n1.
- D. The program prints the content of the three files.

Answer: A

NEW QUESTION 67

Given:

```
class Vehicle { int vno;
String name;
public Vehicle (int vno, String name) { this.vno = vno.;
this.name = name;
}
public String toString () { return vno + ":" + name;
}
}
```

and this code fragment:

```
Set<Vehicle> vehicles = new TreeSet <> (); vehicles.add(new Vehicle (10123, "Ford")); vehicles.add(new Vehicle (10124, "BMW")); System.out.println(vehicles);
```

What is the result?

- A. 10123 Ford10124 BMW

- B. 10124 BMW10123 Ford
- C. A compilation error occurs.
- D. A ClassCastException is thrown at run time.

Answer: D

NEW QUESTION 69

Given the code fragments:

```
public class Video {
    public void play() throws IOException {
        System.out.print("Video played.");
    }
}

public class Game extends Video {
    public void play() throws Exception {
        super.play();
        System.out.print("Game played.");
    }
}
```

and

```
try {
    new Game().play();
} catch (Exception e) {
    System.out.print(e.getClass());
}
```

What is the result?

- A. Video played.Game played.
- B. A compilation error occurs.
- C. class java.lang.Exception
- D. class java.io.IOException

Answer: C

NEW QUESTION 71

Given the code fragment:

```
List<Integer> list1 = Arrays.asList(10, 20); List<Integer> list2 = Arrays.asList(15, 30);
//line n1
```

Which code fragment, when inserted at line n1, prints 10 20 15 30?

- A. Stream.of(list1, list2).flatMap(list -> list.stream()).forEach(s -> System.out.print(s + " "));
- B. Stream.of(list1, list2).flatMap(list -> list.intStream()).forEach(s -> System.out.print(s + " "));
- C. list1.stream().flatMap(list2.stream()).flatMap(e1 -> e1.stream()).forEach(s -> System.out.println(s + " "));
- D. Stream.of(list1, list2).flatMapToInt(list -> list.stream()).forEach(s -> System.out.print(s + " "));

Answer: A

NEW QUESTION 76

Given the code fragment:

```
List<String> qwords = Arrays.asList("why ", "what ", "when ");
BinaryOperator<String> operator = (s1, s2) -> s1.concat(s2); // line n1
String sen = qwords.stream()
    .reduce("Word: ", operator);
System.out.println(sen);
```

What is the result?

- A. Word: why what when
- B. Word: why Word: why what Word: why what when
- C. Word: why Word: what Word: when
- D. Compilation fails at line n1.

Answer: A

NEW QUESTION 79

Given the definition of the Vehicle class: class Vehicle {
String name;
void setName (String name) { this.name = name;
}
String getName() { return name;
}
}

Which action encapsulates the Vehicle class?

- A. Make the Vehicle class public.
- B. Make the name variable public.
- C. Make the setName method public.
- D. Make the name variable private.
- E. Make the setName method private.
- F. Make the getName method private.

Answer: D

NEW QUESTION 84

Given the code fragments: class TechName {
String techName;
TechName (String techName) { this.techName=techName;
}
}
and
List<TechName> tech = Arrays.asList (new TechName("Java-"),
new TechName("Oracle DB-"), new TechName("J2EE-")
);
Stream<TechName> stre = tech.stream();
//line n1

Which should be inserted at line n1 to print Java-Oracle DB-J2EE-?

- A. stre.forEach(System.out::print);
- B. stre.map(a-> a.techName).forEach(System.out::print);
- C. stre.map(a-> a).forEachOrdered(System.out::print);
- D. stre.forEachOrdered(System.out::print);

Answer: B

NEW QUESTION 86

Given the code fragment:

```
try {
    Properties prop = new Properties();
    prop.put("user", userName);
    prop.put("password", passWord);
    Connection conn = DriverManager.getConnection(dbURL, prop);
    if(conn != null){
        System.out.print("Connection Established");
    }
} catch (Exception e) {
    System.out.print(e);
}
```

and the information:

- The required database driver is configured in the classpath.
- The appropriate database is accessible with the dbURL, username, and passWord exists. What is the result?

- A. A ClassNotFoundException is thrown at runtime.
- B. The program prints nothing.
- C. The program prints Connection Established.
- D. A SQLException is thrown at runtime.

Answer: C

NEW QUESTION 89

Which statement is true about java.util.stream.Stream?

- A. A stream cannot be consumed more than once.
- B. The execution mode of streams can be changed during processing.
- C. Streams are intended to modify the source data.

D. A parallel stream is always faster than an equivalent sequential stream.

Answer: B

NEW QUESTION 93

Given the content:

```
MessagesBundle.properties file:

username = Enter User Name
password = Enter Password

MessagesBundle_fr_FR.properties file:

username = Entrez le nom d'utilisateur
password = Entrez le mot de passe
```

and the code fragment:

```
Locale currentLocale = new Locale.Builder().setRegion("FR").setLanguage("fr").build();
ResourceBundle messages = ResourceBundle.getBundle("MessagesBundle", currentLocale);
Enumeration<String> names = messages.getKeys();
while (names.hasMoreElements()) {
    String key = names.nextElement();
    String name = messages.getString(key);
    System.out.println(key + " = " + name);
}
```

What is the result?

- A. username = Entrez le nom d'utilisateur password = Entrez le mot de passe
- B. username = Enter User Name password = Enter Password
- C. A compilation error occurs.
- D. The program prints nothing.

Answer: A

NEW QUESTION 97

Given the code fragment:

```
List<String> li = Arrays.asList("Java", "J2EE", "J2ME", "JSTL", "JSP", "Oracle DB");
Predicate<String> val = p -> p.contains("J");
List<String> neLi = li.stream().filter(x -> x.length() > 3)
    .filter(val).collect(Collectors.toList());
System.out.println(neLi);
```

What is the result?

- A. A compilation error occurs.
- B. [Java, J2EE, J2ME, JSTL, JSP]
- C. null
- D. [Java, J2EE, J2ME, JSTL]

Answer: A

NEW QUESTION 102

Given the code fragment:

```
Deque<Integer> nums = new ArrayDeque<>();
nums.add(1000);
nums.push(2000);
nums.add(3000);
nums.push(4000);
Integer i1 = nums.remove();
Integer i2 = nums.pop();
System.out.println(i1 + " : " + i2);
```

What is the result?

- A. 4000 : 2000

- B. 4000 : 1000
- C. 1000 : 4000
- D. 1000 : 2000

Answer: B

NEW QUESTION 105

Which two reasons should you use interfaces instead of abstract classes? (Choose two.)

- A. You expect that classes that implement your interfaces have many common methods or fields, or require access modifiers other than public.
- B. You expect that unrelated classes would implement your interfaces.
- C. You want to share code among several closely related classes.
- D. You want to declare non-static on non-final fields.
- E. You want to take advantage of multiple inheritance of type.

Answer: BE

NEW QUESTION 110

Given:

```
class UserException extends Exception { }
class AgeOutOfLimitException extends UserException { } and the code fragment:
class App {
public void doRegister(String name, int age) throws UserException, AgeOutOfLimitException { if (name.length () < 6) {
throw new UserException ();
} else if (age >= 60) {
throw new AgeOutOfLimitException ();
} else {
System.out.println("User is registered.");
}
}
public static void main(String[] args) throws UserException { App t = new App ();
```

- A. t.doRegister("Mathew", 60);}What is the result?
- B. User is registered.
- C. An AgeOutOfLimitException is thrown.
- D. A UserException is thrown.
- E. A compilation error occurs in the main method.

Answer: B

NEW QUESTION 115

Given:

```
class Product {
    String pname;
    public Product(String pname) {
        this.pname = pname;
    }
}
```

and the code fragment:

```
Product p1 = new Product("PowerCharger");
Product p2 = p1;
System.out.println(p1.equals(p2));
Product p3 = new Product("PowerCharger");
System.out.println(p1.equals(p3));
```

What is the result?

- A. true>true
- B. false>true
- C. false>false
- D. true>false

Answer: B

NEW QUESTION 119

Given:

```
class ImageScanner implements AutoCloseable { public void close () throws Exception { System.out.print ("Scanner closed.");
}
public void scanImage () throws Exception { System.out.print ("Scan.");
throw new Exception("Unable to scan.");
```

```

}
}
class ImagePrinter implements AutoCloseable { public void close () throws Exception { System.out.print ("Printer closed.");
}
public void printImage () {System.out.print("Print."); }
}

```

and this code fragment:

```

try (ImageScanner ir = new ImageScanner(); ImagePrinter iw = new ImagePrinter()) { ir.scanImage();
iw.printImage();
} catch (Exception e) { System.out.print(e.getMessage());
}

```

What is the result?

- A. Scan.Printer close
- B. Scanner close
- C. Unable to scan.
- D. Scan.Scanner close
- E. Unable to scan.
- F. Sca
- G. Unable to scan.
- H. Sca
- I. Unable to sca
- J. Printer closed.

Answer: A

NEW QUESTION 122

Given:

```

class DataConverter {
    public void copyFlatFilesToTables() { }
    public void close() throws Exception {
        throw new RuntimeException(); // line n1
    }
}

```

and the code fragment:

```

public static void main(String[] args) throws Exception {
    try (DataConverter dc = new DataConverter()) // line n2
    { dc.copyFlatFilesToTables(); }
}

```

What is the result?

- A. A compilation error occurs at line n2.
- B. A compilation error occurs because the try block doesn't have a catch or finally block.
- C. A compilation error occurs at line n1.
- D. The program compiles successfully.

Answer: B

NEW QUESTION 123

Given the code fragments:

```

class R implements Runnable {
    public void run() { System.out.println("Run..."); }
}

class C implements Callable<String> {
    public String call() throws Exception { return "Call..."; }
}

```

and

```
ExecutorService es = Executors.newSingleThreadExecutor();
es.execute(new R()); // line n1
Future<String> f1 = es.submit(new C()); // line n2
System.out.println(f1.get());
es.shutdown();
```

What is the result?

- A. The program prints Run... and throws an exception.
- B. A compilation error occurs at line n1.
- C. Run...Call...
- D. A compilation error occurs at line n2.

Answer: B

NEW QUESTION 128

Given the code fragments:

```
public static Optional<String> getCountry(String loc) {
    Optional<String> couName = Optional.empty();
    if ("Paris".equals(loc))
        couName = Optional.of("France");
    else if ("Mumbai".equals(loc))
        couName = Optional.of("India");
    return couName;
}
```

and

```
Optional<String> city1 = getCountry("Paris");
Optional<String> city2 = getCountry("Las Vegas");
System.out.println(city1.orElse("Not Found"));
if (city2.isPresent())
    city2.ifPresent(x -> System.out.println(x));
else
    System.out.println(city2.orElse("Not Found"));
```

What is the result?

- A. FranceOptional[NotFound]
- B. Optional [France] Optional [NotFound]
- C. Optional[France] Not Found
- D. FranceNot Found

Answer: D

NEW QUESTION 131

Given the code fragment:

```
List<Integer> li = Arrays.asList(10, 20, 30);
Function<Integer, Integer> fn = f1 -> f1 + f1;
Consumer<Integer> conVal = s -> System.out.print("Val:" + s + " ");
li.stream().map(fn).forEach(conVal);
```

What is the result?

- A. Val:20 Val:40 Val:60
- B. Val:10 Val:20 Val:30
- C. A compilation error occurs.
- D. Val: Val: Val:

Answer: B

NEW QUESTION 132

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your 1Z0-809 Exam with Our Prep Materials Via below:

<https://www.certleader.com/1Z0-809-dumps.html>