

## 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 Programing"); Book b2 = new Book (102, "Java Programing"); 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:

```
class Sum extends RecursiveAction { //line n1 static final int THRESHOLD_SIZE = 3;
int stIndex, lstIndex; int [ ] data;
public Sum (int [ ]data, int start, int end) { this.data = data;
this stIndex = start; this. lstIndex = end;
}
protected void compute ( ) { int sum = 0;
if (lstIndex – stIndex <= THRESHOLD_SIZE) { for (int i = stIndex; i < lstIndex; i++) {
sum += data [i];
}
System.out.println(sum);
} else {
new Sum (data, stIndex + THRESHOLD_SIZE, lstIndex).fork( ); new Sum (data, stIndex,
Math.min (lstIndex, stIndex + THRESHOLD_SIZE)
).compute ();
}
}
}
```

and the code fragment:

```
ForkJoinPool fjPool = new ForkJoinPool ( ); int data [ ] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
fjPool.invoke (new Sum (data, 0, data.length));
and given that the sum of all integers from 1 to 10 is 55. Which statement is true?
```

- A. The program prints several values that total 55.
- B. The program prints 55.
- C. A compilation error occurs at line n1.
- D. The program prints several values whose sum exceeds 55.

**Answer:** A

**NEW QUESTION 3**

Given the code fragment:

```
for (Course a : Course.values()) {
    System.out.print(a + " Fees " + a.getCost()+" " );
}
```

Which is the valid definition of the Course enum?

```
A. enum Course { JAVA(100), J2ME(150);
    private int cost;
    public Course(int c) {
        this.cost = c;
    }
    int getCost() {
        return cost;
    }
}

B. enum Course { JAVA(100), J2ME(150);
    private static int cost;
    private Course(int c) {
        this.cost = c;
    }
    static int getCost() {
        return cost;
    }
}

C. final enum Course { JAVA(100), J2ME(150);
    private int cost;
    public Course(int c) {
        this.cost = c;
    }
    int getCost() {
        return cost;
    }
    void setCost(int c) {
        this.cost = c;
    }
}

D. enum Course { JAVA(100), J2ME(150);
    private int cost;
    Course(int c) {
        this.cost = c;
    }
    int getCost() {
        return cost;
    }
}
```

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

**Answer:** A

#### NEW QUESTION 4

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 5

Which two statements are true about the Fork/Join Framework? (Choose two.)

- A. The RecursiveTask subclass is used when a task does not need to return a result.
- B. The Fork/Join framework can help you take advantage of multicore hardware.
- C. The Fork/Join framework implements a work-stealing algorithm.
- D. The Fork/Join solution when run on multicore hardware always performs faster than standard sequential solution.

**Answer:** AC

#### NEW QUESTION 6

Given:

```
public class Product {
    public double applyDiscount(double price) {
        assert (price > 0); // line n1
        return price * 0.50;
    }
    public static void main(String[] args) {
        Product p = new Product();
        double newPrice =
            p.applyDiscount(Double.parseDouble(args[0]));
        System.out.println("New Price: " + newPrice);
    }
}
```

and the command: `java Product 0` What is the result?

- A. An AssertionError is thrown.
- B. A compilation error occurs at line n1.
- C. New Price: 0.0
- D. A NumberFormatException is thrown at run time.

**Answer:** D

#### NEW QUESTION 7

Given the code fragment:

```
List<String> words = Arrays.asList("win", "try", "best", "luck", "do");
Predicate<String> test1 = w -> {
    System.out.println("Checking...");
    return w.equals("do"); // line n1
};
Predicate test2 = (String w) -> w.length() > 3; // line n2
words.stream()
    .filter(test2)
    .filter(test1)
    .count();
```

What is the result?

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

**Answer:** A

#### NEW QUESTION 8

Given the code fragment:

```
ProductCode<Number, Integer> c1 = new ProductCode<Number, Integer>(); /* c1
instantiation */
ProductCode<Number, String> c2 = new ProductCode<Number, String>();    /* c2
instantiation */
```

You have been asked to define the ProductCode class. The definition of the ProductCode class must allow c1 instantiation to succeed and cause a compilation error on c2 instantiation.

Which definition of ProductCode meets the requirement?

```
A. class ProductCode<T, S<Integer>> {
    T c1;
    S c2;
}

B. class ProductCode<T, S extends T> {
    T c1;
    S c2;
}

C. class ProductCode<T, S> {
    T c1;
    S c2;
}

D. class ProductCode<T, S super T> {
    T c1;
    S c2;
}
```

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

**Answer: B**

#### NEW QUESTION 9

Given the code fragment:

```
Path p1 = Paths.get("/Pics/MyPic.jpeg"); System.out.println (p1.getNameCount() + ":" + p1.getName(1) +
":" + p1.getFileName());
```

Assume that the Pics directory does NOT exist.

What is the result?

- A. An exception is thrown at run time.
- B. 2:MyPic.jpeg: MyPic.jpeg
- C. 1:Pics:/Pics/ MyPic.jpeg
- D. 2:Pics: MyPic.jpeg

**Answer: B**

#### NEW QUESTION 10

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 10

Given:

```
public final class IceCream { public void prepare() {}
}
```

```
public class Cake {
public final void bake(int min, int temp) {} public void mix() {}
```



```
}  
public class Shop {  
    private Cake c = new Cake (); private final double discount = 0.25;  
    public void makeReady () { c.bake(10, 120); }  
}  
public class Bread extends Cake {  
    public void bake(int minutes, int temperature) {} public void addToppings() {}  
}
```

Which statement is true?

- A. A compilation error occurs in IceCream.
- B. A compilation error occurs in Cake.
- C. A compilation error occurs in Shop.
- D. A compilation error occurs in Bread
- E. All classes compile successfully.

**Answer:** D

### NEW QUESTION 13

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 15

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 20

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 25

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 26

Given the content:

```
MessagesBundle.properties file:

inquiry = How are you?

MessagesBundle_de_DE.properties file:

inquiry = Wie geht's?
```

and given the code fragment:

```
Locale currentLocale;
// line 1
ResourceBundle messages = ResourceBundle.getBundle("MessagesBundle", currentLocale);
System.out.println(messages.getString("inquiry"));
```

Which two code fragments, when inserted at line 1 independently, enable the code to print "Wie geht's?"

- A. currentLocale = new Locale ("de", "DE");
- B. currentLocale = new Locale.Builder ().setLanguage ("de").setRegion ("DE").build ();
- C. currentLocale = Locale.GERMAN;
- D. currentLocale = new Locale(); currentLocale.setLanguage ("de"); currentLocale.setRegion ("DE");
- E. currentLocale = Locale.getInstance(Locale.GERMAN,Locale.GERMANY);

**Answer:** B

#### NEW QUESTION 30

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 34

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 38

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 42

Given:

```
class Bird {
public void fly () { System.out.print(“Can fly”); }
}
class Penguin extends Bird {
public void fly () { System.out.print(“Cannot fly”); }
}
and the code fragment:
class Birdie {
public static void main (String [ ] args) { fly( ( ) -> new Bird ( ));
fly (Penguin : : new);
}
/* line n1 */
}
```

Which code fragment, when inserted at line n1, enables the Birdie class to compile?

- A. static void fly (Consumer<Bird> bird) { bird :: fly ();}
- B. static void fly (Consumer<? extends Bird> bird) {bird.accept( ) fly ();}
- C. static void fly (Supplier<Bird> bird) { bird.get( ) fly ();}
- D. static void fly (Supplier<? extends Bird> bird) { LOST

**Answer:** C

#### NEW QUESTION 47

Given:

```
public class Counter {
public static void main (String[ ] args) { int a = 10;
int b = -1;
assert (b >=1) : “Invalid Denominator”; int = a / b;
System.out.println (c);
}
}
```



What is the result of running the code with the `-ea` option?

- A. -10
- B. An `AssertionError` is thrown.
- C. A compilation error occurs.

**Answer:** C

#### NEW QUESTION 52

Given the code fragment:

```
Path path1 = Paths.get("/app/.sys/"); Path res1 = path1.resolve("log");  
Path path2 = Paths.get("/server/exe/"); Path res1 = path1.resolve("readme"); System.out.println(res1); System.out.println(res2);  
What is the result?
```

- A. `/app/sys/log/readme/server/exe`
- B. `/app/log/sys/server/exe/readme`
- C. `/app/.sys/log/readme`
- D. `/app/.sys/log/server/exe/readme`

**Answer:** C

#### NEW QUESTION 57

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 62

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 63

Given the definition of the `Emp` class:

```
public class Emp {  
    private String eName; private Integer eAge;  
    Emp(String eN, Integer eA) { this.eName = eN;  
    this.eAge = eA;  
    }  
    public Integer getEAge () {return eAge;} public String getENAME () {return eName;}  
}
```

and code fragment:

```
List<Emp>li = Arrays.asList(new Emp("Sam", 20), New Emp("John", 60), New Emp ("Jim", 51));
```

```
Predicate<Emp> agVal = s -> s.getEAge() > 50; //line n1
li = li.stream().filter(agVal).collect(Collectors.toList());
Stream<String> names = li.stream().map.(Emp::getENAME); //line n2
names.forEach(n -> System.out.print(n + " "));
```

What is the result?

- A. Sam John Jim
- B. John Jim
- C. A compilation error occurs at line n1.
- D. A compilation error occurs at line n2.

**Answer:** B

#### NEW QUESTION 68

Given the content of /resources/Message.properties: welcome1="Good day!"  
and given the code fragment: Properties prop = new Properties ();  
FileInputStream fis = new FileInputStream ("/resources/Message.properties"); prop.load(fis);  
System.out.println(prop.getProperty("welcome1")); System.out.println(prop.getProperty("welcome2", "Test")); //line n1  
System.out.println(prop.getProperty("welcome3"));  
What is the result?

- A. Good day!Testfollowed by an Exception stack trace
- B. Good day!followed by an Exception stack trace
- C. Good day!Test null
- D. A compilation error occurs at line n1.

**Answer:** C

#### NEW QUESTION 71

Given the code fragment: public class Foo {  
public static void main (String [ ] args) {  
Map<Integer, String> unsortMap = new HashMap< > ( ); unsortMap.put (10, "z");  
unsortMap.put (5, "b");  
unsortMap.put (1, "d");  
unsortMap.put (7, "e");  
unsortMap.put (50, "j");  
Map<Integer, String> treeMap = new TreeMap <Integer, String> (new Comparator<Integer> ( ) {  
@Override public int compare (Integer o1, Integer o2) {return o2.compareTo  
(o1); } } );  
treeMap.putAll (unsortMap);  
for (Map.Entry<Integer, String> entry : treeMap.entrySet ( ) ) { System.out.print (entry.getValue ( ) + " ");  
}  
}  
}  
}  
What is the result?

- A. A compilation error occurs.
- B. d b e z j
- C. j z e b d
- D. z b d e j

**Answer:** C

#### NEW QUESTION 75

Given the code fragments:  
class MyThread implements Runnable {  
private static AtomicInteger count = new AtomicInteger (0); public void run () {  
int x = count.incrementAndGet(); System.out.print (x+" ");  
}  
}  
and  
Thread thread1 = new Thread(new MyThread()); Thread thread2 = new Thread(new MyThread()); Thread thread3 = new Thread(new MyThread()); Thread [] ta =  
{thread1, thread2, thread3};  
for (int x= 0; x < 3; x++) { ta[x].start();  
}  
Which statement is true?

- A. The program prints 1 2 3 and the order is unpredictable.
- B. The program prints 1 2 3.
- C. The program prints 1 1 1.
- D. A compilation error occurs.

**Answer:** A

#### NEW QUESTION 80

Locale	Currency Symbol	Currency Code
US	\$	USD

and the code fragment?

```
double d = 15;  
Locale l = new Locale("en", "US");  
NumberFormat formatter = NumberFormat.getCurrencyInstance(l);  
System.out.println(formatter.format(d));
```

What is the result?

- A. \$15.00
- B. 15 \$
- C. USD 15.00
- D. USD \$15

**Answer:** A

#### NEW QUESTION 81

You want to create a singleton class by using the Singleton design pattern. Which two statements enforce the singleton nature of the design? (Choose two.)

- A. Make the class static.
- B. Make the constructor private.
- C. Override equals() and hashCode() methods of the java.lang.Object class.
- D. Use a static reference to point to the single instance.
- E. Implement the Serializable interface.

**Answer:** BD

#### NEW QUESTION 85

Given the code fragments:

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

and

```
List<String> strs = 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 = strs.stream()  
    .filter(cf1)  
    .filter(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 89

Given the code fragment:

```
BiFunction<Integer, Double, Integer> val = (t1, t2) -> t1 + t2; //line n1 System.out.println(val.apply(10, 10.5));  
What is the result?
```

- A. 20
- B. 20.5
- C. A compilation error occurs at line n1.
- D. A compilation error occurs at line n2.

**Answer:** C

#### NEW QUESTION 91

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 92**

Given:

```
class FuelNotAvailException extends Exception { }
class Vehicle {
void ride() throws FuelNotAvailException { //line n1 System.out.println("Happy Journey!");
}
}
class SolarVehicle extends Vehicle {
public void ride () throws Exception { //line n2 super ride ();
}
}
```

and the code fragment:

```
public static void main (String[] args) throws FuelNotAvailException, Exception
{
Vehicle v = new SolarVehicle (); v.ride();
}
```

Which modification enables the code fragment to print Happy Journey!?

- A. Replace line n1 with public void ride() throws FuelNotAvailException {
- B. Replace line n1 with protected void ride() throws Exception {
- C. Replace line n2 with void ride() throws Exception {
- D. Replace line n2 with private void ride() throws FuelNotAvailException {

**Answer:** B

**NEW QUESTION 96**

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 101**

Given:

```
class CheckClass {
public static int checkValue (String s1, String s2) { return s1.length() – s2.length();
}
}
```

and the code fragment:

```
String[] strArray = new String [] {"Tiger", "Rat", "Cat", "Lion"}
//line n1
for (String s : strArray) { System.out.print (s + " ");
}
}
```

Which code fragment should be inserted at line n1 to enable the code to print Rat Cat Lion Tiger?

- A. Arrays.sort(strArray, CheckClass : : checkValue);
- B. Arrays.sort(strArray, (CheckClass : : new) : : checkValue);
- C. Arrays.sort(strArray, (CheckClass : : new).checkValue);
- D. Arrays.sort(strArray, CheckClass : : new : : checkValue);

**Answer:** A

**NEW QUESTION 106**

Given:

```
interface P { public void method1(); }

interface Q extends P { public void method1(); }

interface R extends P { public void method2(); }

interface S { public default void method() { } }

interface T { public void method1(); public void method2(); }

interface U { public void method1(); public abstract void method2(); }
```

Which two interfaces can you use to create lambda expressions? (Choose two.)



- A. T
- B. R
- C. P
- D. S
- E. Q
- F. U

**Answer:** AF

#### NEW QUESTION 108

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 113

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 118

Given the code fragment:

```
String str = "Java is a programming language";  
ToIntFunction<String> indexVal = str::indexOf; //line n1  
int x = indexVal.applyAsInt("Java"); //line n2  
System.out.println(x);  
What is the result?
```

- A. 1
- B. A compilation error occurs at line n1.
- C. A compilation error occurs at line n2.

**Answer:** A

#### NEW QUESTION 122

Given:

```
interface Interfacel {
    public default void sayHi() {
        System.out.println("Hi Interface-1");
    }
}

interface Interface2 {
    public default void sayHi() {
        System.out.println("Hi Interface-2");
    }
}

public class MyClass implements Interfacel, Interface2 {
    public static void main(String[] args) {
        Interfacel obj = new MyClass();
        obj.sayHi();
    }
    public void sayHi() {
        System.out.println("Hi MyClass");
    }
}
```

What is the result?

- A. Hi Interface-2
- B. A compilation error occurs.
- C. Hi Interface-1
- D. Hi MyClass

**Answer:** D

#### NEW QUESTION 123

Given:

```
public interface LengthValidator {
    public boolean checkLength(String str);
}
```

and

```
public class Txt {
    public static void main(String[] args) {
        boolean res = new LengthValidator() {
            public boolean checkLength(String str) {
                return str.length() > 5 && str.length() < 10;
            }
        }.checkLength("Hello");
    }
}
```

Which interface from the java.util.function package should you use to refactor the class Txt?

- A. Consumer
- B. Predicate
- C. Supplier
- D. Function

**Answer:** C

#### NEW QUESTION 125

Given:

```
public class Canvas implements Drawable { public void draw () { }
}
public abstract class Board extends Canvas {
}
public class Paper extends Canvas { protected void draw (int color) { }
}
public class Frame extends Canvas implements Drawable { public void resize () { }
```

```
}  
public interface Drawable { public abstract void draw ();  
}
```

Which statement is true?

- A. Board does not compile.
- B. Paper does not compile.
- C. Frame does not compile.
- D. Drawable does not compile.
- E. All classes compile successfully.

**Answer:** E

#### NEW QUESTION 128

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 132

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 134

Given the code fragment:  
`List<String> colors = Arrays.asList("red", "green", "yellow"); Predicate<String> test = n -> { System.out.println("Searching...");  
return n.contains("red");  
};  
colors.stream()  
.filter(c -> c.length() > 3)  
.allMatch(test);` What is the result?

- A. Searching...
- B. Searching...Searching...
- C. Searching...Searching... Searching...
- D. A compilation error occurs.

**Answer:** A

#### NEW QUESTION 136

Given the code fragment:  
`LocalDate valentinesDay =LocalDate.of(2015, Month.FEBRUARY, 14); LocalDate nextYear = valentinesDay.plusYears(1); nextYear.plusDays(15); //line n1  
System.out.println(nextYear);` What is the result?

- A. 2016-02-14
- B. A DateTimeException is throw
- C. 2016-02-29
- D. A compilation error occurs at line n1.

**Answer:** A

#### NEW QUESTION 138

Given the code fragment:  
`List<Integer> nums = Arrays.asList (10, 20, 8); System.out.println (`

//line n1

);

Which code fragment must be inserted at line n1 to enable the code to print the maximum number in the nums list?

- A. `nums.stream().max(Comparator.comparing(a -> a)).get()`
- B. `nums.stream().max(Integer : : max).get()`
- C. `nums.stream().max()`
- D. `nums.stream().map(a -> a).max()`

**Answer:** A

#### NEW QUESTION 139

Given the definition of the Country class: `public class country {  
public enum Continent {ASIA, EUROPE} String name;  
Continent region;  
public Country (String na, Continent reg) { name = na, region = reg;  
}  
public String getName () {return name;} public Continent getRegion () {return region;}  
}`

and the code fragment:

```
List<Country> couList = Arrays.asList (  
new Country ("Japan", Country.Continent.ASIA), new Country ("Italy", Country.Continent.EUROPE),  
new Country ("Germany", Country.Continent.EUROPE)); Map<Country.Continent, List<String>> regionNames = couList.stream ()  
.collect(Collectors.groupingBy (Country ::getRegion, Collectors.mapping(Country::getName, Collectors.toList()))); System.out.println(regionNames);
```

- A. {EUROPE = [Italy, Germany], ASIA = [Japan]}
- B. {ASIA = [Japan], EUROPE = [Italy, Germany]}
- C. {EUROPE = [Germany, Italy], ASIA = [Japan]}
- D. {EUROPE = [Germany], EUROPE = [Italy], ASIA = [Japan]}

**Answer:** B

#### NEW QUESTION 144

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 145

Given the code fragment:

```
List<Integer> prices = Arrays.asList(3, 4, 5);  
prices.stream()  
    .filter(e -> e > 4)  
    .peek(e -> System.out.print("Price " + e))           // line n1  
    .map(n -> n - 1)                                     // line n2  
    .peek(n -> System.out.println(" New Price " + n));   // line n3
```

Which modification enables the code to print Price 5 New Price 4?

- A. Replace line n2 with `.map (n -> System.out.println ("New Price" + n -1))` and remove line n3
- B. Replace line n2 with `.mapToInt (n -> n - 1);`
- C. Replace line n1 with `.forEach (e -> System.out.print ("Price" + e))`
- D. Replace line n3 with `.forEach (n -> System.out.println ("New Price" + n));`

**Answer:** A

#### NEW QUESTION 147

Given:



```
class Person {
    private String firstName;
    private int salary;
    public Person(String fN, int sal) {
        this.firstName = fN;
        this.salary = sal;
    }
    public int getSalary() { return salary; }
    public String getFirstName() { return firstName; }
}
```

and the code fragment:

```
List<Person> prog = Arrays.asList(
    new Person("Smith", 1500),
    new Person("John", 2000),
    new Person("Joe", 1000));
double dVal = prog.stream()
    .filter(s -> s.getFirstName().startsWith("J"))
    .mapToInt(Person::getSalary)
    .average()
    .getAsDouble();
System.out.print(dVal);
```

What is the result?

- A. 0.0
- B. 1500.0
- C. A compilation error occur
- D. 2000.0

**Answer:** D

#### NEW QUESTION 152

Given the code fragment:

```
public static void main(String[] args) {
    Console console = System.console();
    char[] pass = console.readPassword("Enter password:"); // line n1
    String password = new String(pass); // line n2
}
```

What is the result?

- A. A compilation error occurs at line n1.
- B. A compilation error occurs at line n2.
- C. The code reads the password without echoing characters on the console.
- D. A compilation error occurs because the IOException isn't declared to be thrown or caught?

**Answer:** D

#### NEW QUESTION 155

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 159

Given the code fragment:

```
Deque<String> queue = new ArrayDeque<>();
queue.add("Susan");
queue.add("Allen");
queue.add("David");
System.out.println(queue.pop());
System.out.println(queue.remove());
System.out.println(queue);
```

What is the result?

- A. DavidDavid[Susan, Allen]
- B. SusanSusan[Susan, Allen]
- C. SusanAllen [David]
- D. DavidAllen [Susan]
- E. SusanAllen[Susan, David]

**Answer:** C

#### NEW QUESTION 161

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 165

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 169

Given the code fragment:

```
final List<String> list = new CopyOnWriteArrayList<>();
final AtomicInteger ai = new AtomicInteger(0);
final CyclicBarrier barrier = new CyclicBarrier(2, new Runnable() {
    public void run() { System.out.println(list); }
});
Runnable r = new Runnable() {
    public void run() {
        try {
            Thread.sleep(1000 * ai.incrementAndGet());
            list.add("X");
            barrier.await();
        } catch (Exception ex) {
        }
    }
};
new Thread(r).start();
new Thread(r).start();
new Thread(r).start();
new Thread(r).start();
```

What is the result ?

- A. [X][X, X][X, X, X][X, X, X, X]
- B. [X, X]
- C. [X][X, X][X, X, X]
- D. [X, X][X, X, X, X]

**Answer:** A

#### NEW QUESTION 172

Given the records from the STUDENT table:

sid	sname	semail
111	James	james@uni.com
112	Jane	jane@uni.com
114	John	john@uni.com

Given the code fragment:

```
public static void main(String[] args) throws SQLException {
    //code to load and register valid jdbc driver go here
    Connection con = DriverManager.getConnection(URL, username, password);
    Statement st = con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,
                                       ResultSet.CONCUR_UPDATABLE);

    st.execute("SELECT * FROM student");
    ResultSet rs = st.getResultSet();
    rs.absolute(3);
    rs.moveToInsertRow();
    rs.updateInt(1, 113);
    rs.updateString(2, "Jannet");
    rs.updateString(3, "jannet@uni.com");
    rs.updateRow();
    rs.refreshRow();
    System.out.println(rs.getInt(1) + " : " + rs.getString(2) + " : " + rs.getString
(3));
}
```

Assume that the URL, username, and password are valid. What is the result?

- A. The STUDENT table is not updated and the program prints: 114 : John : john@uni.com
- B. The STUDENT table is updated with the record: 113 : Jannet : jannet@uni.com and the program prints: 114 : John : john@uni.com
- C. The STUDENT table is updated with the record: 113 : Jannet : jannet@uni.com and the program prints: 113 : Jannet : jannet@uni.com
- D. A SQLException is thrown at run time.

**Answer:** A



**NEW QUESTION 174**

Given:

```
class Engine {  
    double fuelLevel;  
    Engine(int fuelLevel) { this.fuelLevel = fuelLevel; }  
    public void start() {  
        // line n1  
        System.out.println("Started");  
    }  
    public void stop() { System.out.println("Stopped"); }  
}
```

Your design requires that:

- ☐ fuelLevel of Engine must be greater than zero when the start() method is invoked.
- ☐ The code must terminate if fuelLevel of Engine is less than or equal to zero.

Which code fragment should be added at line n1 to express this invariant condition?

- A. `assert (fuelLevel) : "Terminating...";`
- B. `assert (fuelLevel > 0) : System.out.println ("Impossible fuel");`
- C. `assert fuelLevel < 0: System.exit(0);`
- D. `assert fuelLevel > 0: "Impossible fuel" ;`

**Answer: C****NEW QUESTION 175**

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 179**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 184**

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 188

Given the code fragment:

```
9. Connection conn = DriverManager.getConnection(dbURL, userName, passWord);
10. String query = "SELECT id FROM Employee";
11. try (Statement stmt = conn.createStatement()) {
12.     ResultSet rs = stmt.executeQuery(query);
13.     stmt.executeQuery("SELECT id FROM Customer");
14.     while (rs.next()) {
15.         //process the results
16.         System.out.println("Employee ID: "+ rs.getInt("id"));
17.     }
18. } catch (Exception e) {
19.     System.out.println ("Error");
20. }
```

Assume that:

The required database driver is configured in the classpath.

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

The Employee and Customer tables are available and each table has id column with a few records and the SQL queries are valid.

What is the result of compiling and executing this code fragment?

- A. The program prints employee IDs.
- B. The program prints customer IDs.
- C. The program prints Error.
- D. compilation fails on line 13.

**Answer: C**

#### NEW QUESTION 190

Which two are elements of a singleton class? (Choose two.)

- A. a transient reference to point to the single instance
- B. a public method to instantiate the single instance
- C. a public static method to return a copy of the singleton reference
- D. a private constructor to the class
- E. a public reference to point to the single instance

**Answer: BD**

#### NEW QUESTION 193

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 194

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 198

Given:

```
public interface Moveable<Integer> {  
    public default void walk (Integer distance) {System.out.println("Walking");} public void run(Integer distance);  
}
```

Which statement is true?

- A. Moveable can be used as below: Moveable<Integer> animal = n -> System.out.println("Running" + n); animal.run(100); animal.walk(20);
- B. Moveable can be used as below: Moveable<Integer> animal = n -> n + 10; animal.run(100); animal.walk(20);
- C. Moveable can be used as below: Moveable animal = (Integer n) -> System.out.println(n); animal.run(100); Moveable.walk(20);
- D. Movable cannot be used in a lambda expression.

**Answer:** A

#### NEW QUESTION 202

Given the code fragment:

```
List<String> nums = Arrays.asList("EE", "SE");
String ans = nums
    .parallelStream()
    .reduce("Java ", (a, b) -> a.concat(b));
System.out.print(ans);
```

What is the result?

- A. Java EEJava EESE
- B. Java EESE
- C. The program prints either:Java EEJava SE orJava SEJava EE
- D. Java EEJava SE

**Answer:** D

#### NEW QUESTION 204

Given the content of Operator.java, EngineOperator.java, and Engine.java files:

```
Operator.java:
public abstract class Operator {
    protected void turnON();
    protected void turnOFF();
}

EngineOperator.java:
public class EngineOperator extends Operator{
    public final void turnON() { System.out.print("ON "); }
    public final void turnOFF() { System.out.println("OFF"); }
}

Engine.java:
public class Engine{
    Operator m = new EngineOperator();
    public void operate() {
        m.turnON();
        m.turnOFF();
    }
}
```

and the code fragment:

```
Engine carEngine = new Engine();
carEngine.operate();
```

What is the result?

- A. The Engine.java file fails to compile.
- B. The EngineOperator.java file fails to compile.
- C. The Operator.java file fails to compile.
- D. ON OFF

**Answer:** A

#### NEW QUESTION 209

Given the code fragments:

```
class Caller implements Callable<String> { String str;
public Caller (String s) {this.str=s;}
public String call()throws Exception { return str.concat ("Caller");}
}
```

```
class Runner implements Runnable { String str;
public Runner (String s) {this.str=s;}
public void run () { System.out.println (str.concat ("Runner"));}
}
```

and

```
public static void main (String[] args) InterruptedException, ExecutionException
{
    ExecutorService es = Executors.newFixedThreadPool(2); Future f1 = es.submit (new Caller ("Call"));
    Future f2 = es.submit (new Runner ("Run")); String str1 = (String) f1.get();
}
```



```
String str2 = (String) f2.get(); //line n1 System.out.println(str1+ ":" + str2);
}
```

What is the result?

- A. The program prints: Run RunnerCall Caller : nullAnd the program does not terminate.
- B. The program terminates after printing: Run RunnerCall Caller : Run
- C. A compilation error occurs at line n1.
- D. An Execution is thrown at run time.

**Answer:** A

#### NEW QUESTION 213

Given the code fragments :

```
public class Product {
    String name;
    Integer price;
    Product(String name, Integer price) {
        this.name = name;
        this.price = price;
    }
    public void printVal(){ System.out.print(name + " Price:" + price + " "); }
    public void setPrice(int price) { this.price = price; }
    public Integer getPrice() { return price; }
}
```

and

```
List<Product> li = Arrays.asList(new Product("TV", 1000), new Product("Refrigerator",
2000));
Consumer<Product> raise = e -> e.setPrice(e.getPrice() + 100);
li.forEach(raise);
li.stream().forEach(Product::printVal);
```

What is the result?

- A. TV Price :110 Refrigerator Price :2100
- B. A compilation error occurs.
- C. TV Price :1000 Refrigerator Price :2000
- D. The program prints nothing.

**Answer:** C

#### NEW QUESTION 216

Given the code fragment:

```
class CallerThread implements Callable<String> { String str;
public CallerThread(String s) {this.str=s;} public String call() throws Exception { return str.concat("Call");
}
}
```

and

```
public static void main (String[] args) throws InterruptedException, ExecutionException
{
    ExecutorService es = Executors.newFixedThreadPool(4); //line n1 Future f1 = es.submit (newCallerThread("Call"));
    String str = f1.get().toString(); System.out.println(str);
}
```

Which statement is true?

- A. The program prints Call Call and terminates.
- B. The program prints Call Call and does not terminate.
- C. A compilation error occurs at line n1.
- D. An ExecutionException is thrown at run time.

**Answer:** B

#### NEW QUESTION 220

Given the code fragment:

```
10. try {
11.     Connection conn = DriverManager.getConnection(dbURL, userName, passWord);
12.     String query = "SELECT * FROM Employee WHERE ID = 110";
13.     Statement stmt = conn.createStatement();
14.     ResultSet rs = stmt.executeQuery(query);
15.     System.out.println("Employee ID: " + rs.getInt("ID"));
16. } catch (Exception se) {
17.     System.out.println("Error");
18. }
```



Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the dbURL, userName, and passWord exists The Employee table has a column ID of type integer and the SQL query matches one record. What is the result?

- A. Compilation fails at line 14.
- B. Compilation fails at line 15.
- C. The code prints the employee ID.
- D. The code prints Error.

**Answer:** A

#### NEW QUESTION 221

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 or non-final fields.
- E. You want to take advantage of multiple inheritance of type.

**Answer:** BE

#### NEW QUESTION 222

Which two code blocks correctly initialize a Locale variable? (Choose two.)

- A. Locale loc1 = "UK";
- B. Locale loc2 = Locale.getInstance("ru");
- C. Locale loc3 = Locale.getLocaleFactory("RU");
- D. Locale loc4 = Locale.UK;
- E. Locale loc5 = new Locale ("ru", "RU");

**Answer:** DE

#### NEW QUESTION 223

Given the definition of the Employee class:

```
class Employee {
    String dept, name;
    public Employee(String d, String n) {
        dept = d;
        name = n;
    }
    public String toString() {
        return getDept() + ":" + getName();
    }
    public String getDept() { return dept; }
    public String getName() { return name; }
}
```

and this code fragment:

```
List<Employee> emps = Arrays.asList(new Employee("sales", "Ada"),
    new Employee("sales", "Bob"),
    new Employee("hr", "Bob"),
    new Employee("hr", "Eva"));
Stream<Employee> s = emps.stream()
    .sorted(Comparator.comparing((Employee e) -> e.getDept())
        .thenComparing((Employee e) -> e.getName()));
List<Employee> eSorted = s.collect(Collectors.toList());
System.out.println(eSorted);
```

What is the result?

- A. [sales:Ada, hr:Bob, sales:Bob, hr:Eva]
- B. [Ada:sales, Bob:sales, Bob:hr, Eva:hr]

- C. [hr:Eva, hr:Bob, sales:Bob, sales:Ada]  
D. [hr:Bob, hr:Eva, sales:Ada, sales:Bob]

**Answer:** A

#### NEW QUESTION 226

Given the code fragment:

```
List<String> str = Arrays.asList ("my", "pen", "is", "your", "pen"); Predicate<String> test = s -> {  
int i = 0;  
boolean result = s.contains ("pen");  
System.out.print(i++) + ":"; return result;  
};  
str.stream()  
.filter(test)  
.findFirst()  
.ifPresent(System.out ::print); What is the result?
```

- A. 0 : 0 : pen  
B. 0 : 1 : pen  
C. 0 : 0 : 0 : 0 : 0 : pen  
D. 0 : 1 : 2 : 3 : 4 :  
E. A compilation error occurs.

**Answer:** A

#### NEW QUESTION 230

Given:

```
public class Emp { String fName; String lName;  
public Emp (String fn, String ln) { fName = fn;  
lName = ln;  
}  
public String getfName() { return fName; } public String getlName() { return lName; }  
}
```

and the code fragment: List<Emp> emp = Arrays.asList ( new Emp ("John", "Smith"),  
new Emp ("Peter", "Sam"),  
new Emp ("Thomas", "Wale")); emp.stream()  
//line n1

.collect(Collectors.toList());

Which code fragment, when inserted at line n1, sorts the employees list in descending order of fName and then ascending order of lName?

- A. .sorted (Comparator.comparing(Emp::getfName).reserved().thenComparing(Emp::getlName))  
B. .sorted (Comparator.comparing(Emp::getfName).thenComparing(Emp::getlName))  
C. .map(Emp::getfName).sorted(Comparator.reserveOrder())  
D. .map(Emp::getfName).sorted(Comparator.reserveOrder()).map (Emp::getlName).reserved

**Answer:** A

#### NEW QUESTION 234

Given the code fragment:

```
public static void main (String [ ] args) throws IOException {  
BufferedReader br = new BufferedReader (new InputStremReader (System.in)); System.out.print ("Enter GDP: ");  
//line 1  
}
```

Which code fragment, when inserted at line 1, enables the code to read the GDP from the user?

- A. int GDP = Integer.parseInt (br.readline());  
B. int GDP = br.read();  
C. int GDP = br.nextlnt();  
D. int GDP = Integer.parseInt (br.next());

**Answer:** A

#### NEW QUESTION 236

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 240

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 242

Which action can be used to load a database driver by using JDBC3.0?

- A. Add the driver class to the META-INF/services folder of the JAR file.
- B. Include the JDBC driver class in a jdbc.properties file.
- C. Use the java.lang.Class.forName method to load the driver class.
- D. Use the DriverManager.getDriver method to load the driver class.

**Answer:** C

#### NEW QUESTION 245

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 249

Given the Greetings.properties file, containing:

```
HELLO_MSG = Hello, everyone!
GOODBYE_MSG = Goodbye everyone!
```

and given:

```
import java.util.Enumeration;
import java.util.Locale;
import java.util.ResourceBundle;

public class ResourcesApp {
    public void loadResourceBundle() {
        ResourceBundle resource = ResourceBundle.getBundle("Greetings", Locale.US);
        System.out.println(resource.getObject(1));
    }
    public static void main(String[] args) {
        new ResourcesApp().loadResourceBundle();
    }
}
```

What is the result?

- A. Compilation fails.
- B. GOODBYE\_MSG
- C. Hello, everyone!
- D. Goodbye everyone!
- E. HELLO\_MSG

**Answer:** A

#### NEW QUESTION 254

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 259

.....



## 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>