Vendor: Oracle

Exam Code: 1Z0-808

Exam Name: Java SE 8 Programmer I

Question 101—Question 110

QUESTION 101
Which code fragment cause a compilation error?

A.
```
float flt = 100F;
```
B.
```
float flt = (float) 1_11.00;
```
C.
```
float flt = 100;
```
D.
```
double y1 = 203.22;
float flt = y1;
```
E.
```
int y2 = 100;
float flt = (float) y2;
```

Answer: D
Explanation:
Exception in thread "main" java.lang.Error: Unresolved compilation problem:
Type mismatch: cannot convert from double to float

QUESTION 102
Given:

```
class X {
static void m (int[] i) {
i[0] += 7;
}
```

```
public static void main (String[] args) {
int[] j = new int[1];
j[0] = 12;
m(j);
System.out.println(j[0]);
}
}
```

What is the result?

A. 7
B. 12
C. 19
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: C

QUESTION 103
Given:

```
1. public class SampleClass {
2. public static void main (String[] args) {
3. AnotherSampleClass asc = new AnotherSampleClass();
4. SampleClass sc = new SampleClass();
5. // insert code here
6. }
7. }
8. class AnotherSampleClass extends SampleClass {
9. }
```

Which statement, when inserted into line 5, enables the code to compile?

A. asc = sc;
B. sc = asc;
C. asc = (Object) sc;
D. asc = sc.clone;

Answer: B

QUESTION 104
Which statement will empty the contents of a StringBuilder variable named sb?

A. sb.deleteAll;
B. sb.delete(0, sb.size());
C. sb.delete(0, sb.length());
D. sb.removeAll();

Answer: C

QUESTION 105
Given:

```
class MarksOutOfBoundsException extends IndexOutOfBoundsException { }
public class GradingProcess {
void verify(int marks) throws IndexOutOfBoundsException { if (marks >
100) {
throw new MarksOutOfBoundsException();
}
if (marks > 50) {
System.out.print("Pass");
} else {
System.out.print("Fail");
}
}

public static void main(String[] args) {
int marks = Integer.parseInt(args[2]);
try {
new GradingProcess().verify(marks);
} catch (Exception e) {
System.out.print(e.getClass());
}
}
}
```

And the command line invocation:

```
java GradingProcess 89 50 104
```

What is the result?

A. Pass
B. Fail
C. class MarksOutOfBoundsException
D. class IndexOutOfBoundsException
E. class Excpetion

Answer: C

QUESTION 106
Given:

```
1. interface Pet { }
2. class Dog implements Pet { }
3. class Beagle extends Dog { }
```

Which three are valid?

A. Pet a = new Dog();
B. Pet b = new Pet();
C. Dog f = new Pet();
D. Dog d = new Beagle();
E. Pet e = new Beagle();
F. Beagle c = new Dog();

Answer: ADE
Explanation:
B and C aren't valid because Pet is abstract and it can't be instantiated.
F isn't valid because a Dog isn't a Beagle.

QUESTION 107
Given the code fragment:

```
StringBuilder sb = new StringBuilder();
sb.append("World");
```

Which fragment prints Hello World?

A.
```
sb.insert(0, "Hello ");
System.out.println(sb);
```
B.
```
sb.append(0, "Hello ");
System.out.println(sb);
```
C.
```
sb.add(0, "Hello ");
System.out.println(sb);
```
D.
```
sb.set(0, "Hello ");
System.out.println(sb);
```

Answer: A

QUESTION 108
Given:

```
package pkg1;
class Bb { }
public class Ee {
private Ee() { }
}
package pkg2;
final class Ww;
package pkg3;
public abstract class Dd { void m() { } }
```

And,

```
1. package pkg4;
2. import pkg1.*;
3. import pkg2.*;
4. import pkg3.*;
5. // insert a class definition here
```

Which two class definitions, when inserted independently at line 5, enable the code to

compile?

A. class Cc extends Bb { }
B. class Cc extends Ww { }
C. class Cc extends Ee { }
D. class Cc extends Dd { }

Answer: AD Only D because Bb is default access

QUESTION 109
Given:

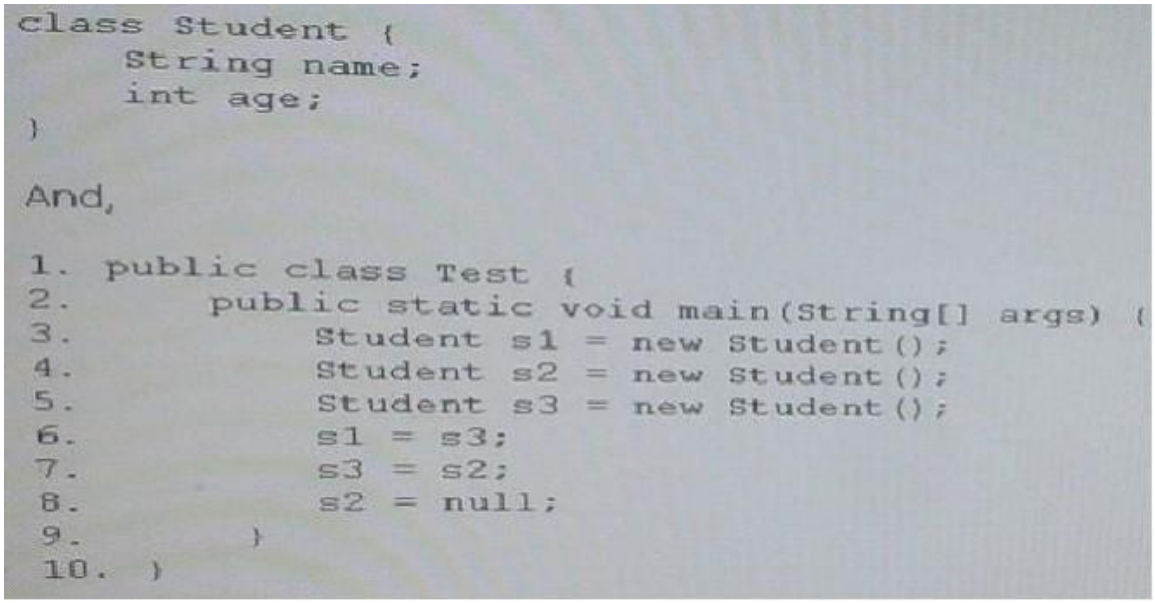
```
1. public class Simple {
2. public float price;
3. public static void main (String [] args) {
4. Simple price = new Simple();
5. price = 4;
6. }
7. }
```

Which will make this code compile and run?

A. Change line 5 to:
price = 4f;
B. Change line 5 to:
price.price = 4;
C. Change line 5 to:
price = (float) 4;
D. Change line 5 to:
price = (Simple) 4;
E. The code compiles and runs properly; no changes are necessary.

Answer: B

QUESTION 110
Given the code fragment:
Which statement is true?



```
class Student {
    String name;
    int age;
}

And,

1. public class Test {
2.     public static void main(String[] args) {
3.         Student s1 = new Student();
4.         Student s2 = new Student();
5.         Student s3 = new Student();
6.         s1 = s3;
7.         s3 = s2;
8.         s2 = null;
9.     }
10. }
```

A. After line 8, three objects are eligible for garbage collection.
B. After line 8, two objects are eligible for garbage collection.
C. After line 8, one object is eligible for garbage collection.
D. After line 8, none of the objects are eligible for garbage collection.

Answer: C
Explanation:
Run the code

```
class Student {
    String name;
    int age;

    // Called by the garbage collector on an object when garbage collection determines
    // that there are no more references to the object.
    @Override
    protected void finalize () {
        System.out.print("Finalized Object\n");
    }
}

public class Main {

    public static void main(String[] args) throws
```

```
InterruptedException {
        Student s1 = new Student();
        Student s2 = new Student();
        Student s3 = new Student();

        s1 = s3;
        s3 = s2;
        s2 = null;

        System.gc();
        Thread.sleep(3000);

        System.out.println(s1.name);
        System.out.println(s3.name);
    }
}
```

The output is
Finalized Object
null
null

In other words,
s2 has been garbage collected.
s1 and s3 haven't been garbage collected.