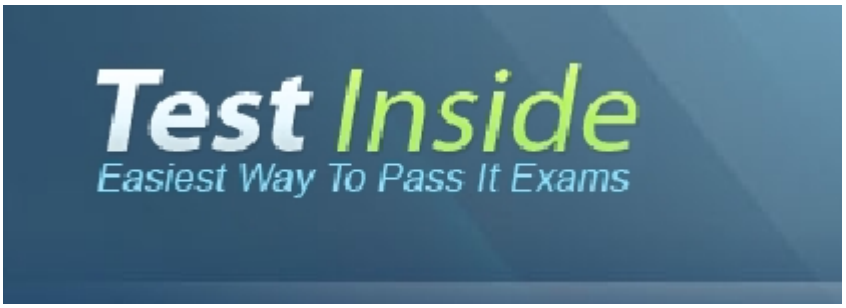
Oracle.Test-inside.1Z0-803.v2013-11-20.by.Susan.90q

Number: 1Z0-803
Passing Score: 770
Time Limit: 150 min
File Version: 12.5

http://www.gratisexam.com/

Exam Code: 1Z0-803

Exam Name: Java SE 7 Programmer I



Exam A

QUESTION 1
Given the code fragment:

```
int[][] array2D = {{0, 1, 2}, {3, 4, 5, 6}};
System.out.print(array2D[0].length + "");
System.out.print(array2D[1].getClass().isArray() + "");
System.out.println(array2D[0][1]);
```

What is the result?

A. 3false1
B. 2true3
C. 2false3
D. 3true1
E. 3false3
F. 2true1
G. 2false1

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 2

```
public class Student
{
   public String name = "";
   public int age = 0;
   public String major = "Undeclared";
   public boolean fulltime = true;

   public void display()
   {
      System.out.println("Name: " + name + " Major: " + major);
   }

   public boolean isFullTime()
   {
      return fulltime;
   }
}

class TestStudent
{
   public static void main(String[] args)
   {
      Student bob = new Student();
      Student jian = new Student();
      bob.name = "Bob";
      bob.age = 19;
      jian = bob;
      jian.name = "Jian";
      System.out.println("Bob's Name: " + bob.name);
   }
}
```

What is the result when this program is executed?

A. Bob's Name: Bob
B. Bob's Name: Jian
C. Nothing prints
D. Bob?s name

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 3
Given the code fragment:

```
String valid = "true";
if (valid)
   System.out.println("valid");
else
   System.out.println("not valid");
```

What is the result?

A. Valid
B. not valid
C. Compilation fails
D. An IllegalArgumentException is thrown at run time

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 4
Given:

```
public class ScopeTest
{
   int z;

   public static void main(String[] args)
   {
      ScopeTest myScope = new ScopeTest();
      int z = 6;
      System.out.println(z);
      myScope.doStuff();
      System.out.println(z);
      System.out.println(myScope.z);
   }

   void doStuff()
   {
      int z = 5;
      doStuff2();
      System.out.println(z);
   }

   void doStuff2()
```

```
   {
      z = 4;
   }
}
```
What is the result?

A. 6 5 6 4
B. 6 5 5 4
C. 6 5 6 6
D. 6 5 6 5

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 5
Which two are valid instantiations and initializations of a multi dimensional array?

A. int [] [] array 2D ={ { 0, 1, 2, 4} {5, 6}};
B. int [] [] array2D = new int [2] [2];
   array2D[0] [0] = 1;
   array2D[0] [1] =2;
   array2D[1] [0] =3;
   array2D[1] [1] =4;
C. int [] [] []array3D = {{0, 1}, {2, 3}, {4, 5}};
D. int [] [] [] array3D = new int [2] [2] [2];
   array3D [0] [0] = array;
   array3D [0] [1] = array;
   array3D [1] [0] = array;
   array3D [0] [1] = array;
E. int [] [] array2D = {0, 1};

Correct Answer: BD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 6
An unchecked exception occurs in a method dosomething()
Should other code be added in the dosomething() method for it to compile and execute?

A. The Exception must be caught
B. The Exception must be declared to be thrown.

C. The Exception must be caught or declared to be thrown.
D. No other code needs to be added.

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 7
Given the code fragment:

```
int b = 4;
b--;
System.out.println(--b);
System.out.println(b);
```

What is the result?

A. 2 2
B. 1 2
C. 3 2
D. 3 3

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 8
Given the code fragment:

```
interface SampleCloseable
{
   public void close() throws java.io.IOException;
}
```

Which three implementations are valid?

A.
```
public class Test implements SampleCloseable
{
   public void close() throws java.io.IOException
   {
      //do something
   }
}
```

B.
```
public class Test implements SampleCloseable
{
   public void close() throws Exception
   {
      // do something
   }
}
```

C.
```
public class Test implementations SampleCloseable
{
   public void close() throws Exception
   {
      // do something
```

```
    }
  }
```
D.
```
public class Test extends SampleCloseable
{
   public void close() throws java.io.IOException
   {
      // do something
   }
}
```

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 9
Given the code fragment:

```
int[][] array = {{0}, {0, 1}, {0, 2, 4}, {0, 3, 6, 9}, {0, 4, 8, 12, 16}};
System.out.println((array[4][1]));
System.out.println((array)[1][4]);
```

What is the result?

A. 4 Null
B. Null 4
C. An IllegalArgumentException is thrown at run time
D. 4 An ArrayIndexOutOfBoundException is thrown at run time

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 10
Given:

```
public class DoCompare1
{
   public static void main(String[] args)
   {
      String[] table = {"aa", "bb", "cc"};
      for (String ss : table)
      {
         int ii = 0;
         while (ii < table.length)
         {
            System.out.println(ss + ", " + ii);
            ii++;
         }
      }
   }
}
```

How many times is 2 printed as a part of the output?

A. Zero
B. Once

C. Twice
D. Thrice
E. Compilation fails.

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 11
Given:

```
import java.io.IOException;

public class Y
{
   public static void main(String[] args)
   {
      try
      {
         doSomething();
      }
      catch (RuntimeException e)
      {
         System.out.println(e);
      }
   }

   static void doSomething()
   {
      if (Math.random() > 0.5)
         throw new IOException();
      throw new RuntimeException();
   }
}
```

Which two actions, used independently, will permit this class to compile?

A. Adding throws IOException to the main() method signature
B. Adding throws IOException to the doSoomething() method signature
C. Adding throws IOException to the main() method signature and to the doSomething() method
D. Adding throws IOException to the doSomething() method signature and changing the catch argument to IOException
E. Adding throws IOException to the main() method signature and changing the catch argument to IOException

Correct Answer: CD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 12
Given:

```
class X
{
   String str = "default";
```

```
   X(String s)
   {
      str = s;
   }

   void print()
   {
      System.out.println(str);
   }

   public static void main(String[] args)
   {
      new X("hello").print();
   }
}
```
What is the result?

A. hello
B. default
C. Compilation fails
D. The program prints nothing
E. An exception is thrown at run time

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 13
Given:
```
public class SampleClass
{
   public static void main(String[] args)
   {
      AnotherSampleClass asc = new AnotherSampleClass();
      SampleClass sc = new SampleClass();
      // TODO code application logic here
   }
}

class AnotherSampleClass extends SampleClass
{
}
```
Which statement, when inserted into line "// TODO code application logic here ", is valid change?

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

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 14
Given the code fragment:

```
System.out.println("Result: " + 2 + 3 + 5);
System.out.println("Result: " + 2 + 1 * 5);
```

What is the result?

A. Result: 10
   Result: 30
B. Result: 10
   Result: 25
C. Result: 235
   Result: 25
D. Result: 215
   Result: 215
E. Compilation fails

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 15
Which code fragment is illegal?

A.
```
class Base1
{
   abstract class Abs1
   {
   }
}
```
B.
```
abstract class Abs1
{
   void doit()
   {
   }
}
```
C.
```
class Basel
{
   abstract class Abs1 extends Basel
   {
   }
}
```
D. abstract int var1=89;

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 16

Given the code fragment:

```
int a = 0;
a++;
System.out.println(a++);
System.out.println(a);
```

What is the result?

A. 1
   2
B. 0
   1
C. 1
   1
D. 2
   2

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 17
Given:

```
public class X
{
   public static void main(String[] args)
   {
      String theString = "Hello World";
      System.out.println(theString.charAt(11));
   }
}
```

What is the result?

A. There is no output
B. d is output
C. A StringIndexOutOfBoundsException is thrown at runtime
D. An ArrayIndexOutOfBoundsException is thrown at runtime
E. A NullPointException is thrown at runtime
F. A StringArrayIndexOutOfBoundsException is thrown at runtime

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 18
Given a java source file:

```
class X
{
   X()
   {
   }
```

```
    private void one()
    {
    }
}

public class Y extends X
{
    Y()
    {
    }

    private void two()
    {
        one();
    }

    public static void main(String[] args)
    {
        new Y().two();
    }
}
```

What changes will make this code compile?

A. adding the public modifier to the declaration ofclass x
B. adding the protected modifier to the x()constructor
C. changing the private modifier on the declaration of the one() method to protected
D. removing the Y () constructor
E. removing the private modifier from the two () method

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 19
Given:
#1
```
package handy.dandy;
public class KeyStroke {
public void typeExclamation() {
System.out.println("!")
}
}
```
#2
```
package handy; /* Line 1 */
public class Greet { /* Line 2 */
public static void main(String[] args) { /* Line 3 */
String greeting = "Hello"; /* Line 4 */
System.out.print(greeting); /* Line 5 */
Keystroke stroke = new Keystroke; /* Line 6 */
stroke.typeExclamation(); /* Line 7 */
} /* Line 8 */
} /* Line 9 */
```
What three modifications, made independently, made to class greet, enable the code to compile and run?

A. Line 6 replaced with handy.dandy.keystroke stroke = new KeyStroke ( );

B. Line 6 replaced with handy.*.KeyStroke = new KeyStroke ( );
C. Line 6 replaced withhandy.dandy.KeyStroke Stroke = new handy.dandy.KeyStroke();
D. import handy.*;addedbeforeline 1
E. import handy.dandy.*;added after line 1
F. import handy.dandy,KeyStroke;added after line 1
G. import handy.dandy.KeyStroke.typeException(); added before line 1

Correct Answer: CEF
Section: (none)
Explanation

Explanation/Reference:

QUESTION 20
Given:

```
String message1 = "Wham bam!";
String message2 = new String("Wham bam!");

if (message1 == message2)
   System.out.println("They match");
if (message1.equals(message2))
   System.out.println("They really match");
```

What is the result?

A. They match
   They really match
B. They really match
C. They match
D. Nothing Prints
E. They really match
   They really match

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 21
Given:

```
public class Speak
{ /* Line 1 */

   public static void main(String[] args)
   { /* Line 2 */
      Speak speakIt = new Tell(); /* Line 3 */
      Tell tellIt = new Tell(); /* Line 4 */
      speakIT.tellItLikeItIs(); /* Line 5 */
      (Truth) speakIt.tellItLikeItIs(); /* Line 6 */
      ((Truth) speakIt).tellItLikeItIs(); /* Line 7 */
      tellIt.tellItLikeItIs(); /* Line 8 */
      (Truth) tellIt.tellItLikeItIs(); /* Line 9 */
      ((Truth) tellIt).tellItLikeItIs(); /* Line 10 */
   }
}
```

```
class Tell extends Speak implements Truth
{
   public void tellItLikeItIs()
   {
      System.out.println("Right on!");
   }
}

interface Truth
{
   public void tellItLikeItIs();
}
```

Which three lines will compile and output "right on!"?

A. Line 5
B. Line6
C. Line 7
D. Line 8
E. Line 9
F. Line 10

Correct Answer: CDF
Section: (none)
Explanation

Explanation/Reference:

QUESTION 22
Given the code fragment:

```
String h1 = "Bob";
String h2 = new String("Bob");
```

What is the best way to test that the values of h1 and h2 are the same?

A. if(h1 == h2)
B. if (h1.equals(h2))
C. if (h1 = = h2)
D. if (h1.same(h2))

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 23
Which two are valid declarations of a two-dimensional array?

A. int[][] array2D;
B. int[2][2] array2D;
C. int array2D[];
D. int[] array2D[];
E. int[][] array2D[];

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 24
Given the code fragment:

```
System.out.println("Result:" + 3 + 5);
System.out.println("result:" + (3 + 5));
```

What is the result?

A. Result: 8
   Result: 8
B. Result: 35
   Result: 8
C. Result: 8
   Result: 35
D. Result: 35
   Result: 35

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 25
Given:

```
public class Main
{
   public static void main(String[] args) throws Exception
   {
      doSomething();
   }

   private static void doSomething() throws Exception
   {
      System.out.println("Before if clause");
      if (Math.random() > 0.5)
      {
         throw new Exception();
      }
      System.out.println("After if clause");
   }
}
```

Which two are possible outputs?

A. Before if clause
   Exception in thread ?main? java.lang.Exception
   AtMain.doSomething (Main.java:8)
   At Main.main (Main.java:3)
B. Before if clause
   Exceptionin thread ?main? java.lang.Exception
   At Main.doSomething (Main.java:8)

At Main.main (Main.java:3)
After if clause

C. Exception in thread ?main? java.lang.Exception
At Main.doSomething (Main.java:8)
At Main.main (Main.java:3)

D. Before if clause
After if clause

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 26
A method doSomething() that has no exception handling code is modified to trail a method that throws a checked exception.
Which two modifications, made independently, will allow the program to compile?

A. Catch the exception in the method doSomething().
B. Declare the exception to be thrown in the doSomething() method signature.
C. Cast the exception to a RuntimeException in the doSomething() method.
D. Catch the exception in the method that calls doSomething().

Correct Answer: AB
Section: (none)
Explanation

Explanation/Reference:

QUESTION 27
Given the code fragment:

```
String color = "Red";
switch (color)
{
   case "Red":
      System.out.println("Found Red");
   case "Blue":
      System.out.println("Found Blue");
      break;
   case "White":
      System.out.println("Found White");
      break;
   default:
      System.out.println("Found Default");
}
```

What is the result?

A. Found Red
B. Found Red
Found Blue
C. Found Red
Found Blue
Found White
D. Found Red

Found Blue
Found White
Found Default

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 28
Which two may precede the word "class" in a class declaration?

A. local
B. public
C. static
D. volatile
E. synchronized

Correct Answer: BC
Section: (none)
Explanation

Explanation/Reference:

QUESTION 29
Which three are bad practices?

A. Checking for ArrayindexoutofBoundsException when iterating through an array to determinewhen all elements have been visited
B. Checking for Error and. If necessary, restartingthe program to ensure that users are unaware problems
C. Checking for FileNotFoundException to inform a user that a filename entered is not valid
D. Checking for ArrayIndexoutofBoundsExcepcion and ensuring that the program can recover if one occur
E. Checking for an IOException and ensuring that the program can recover if one occurs

Correct Answer: ABD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 30
Given:

```
public class Bark
{
   // Insert code here - Line 5
   public abstract void bark(); // Line 6

}  // Line 7

   // Line 8
   // Insert code here - Line 9
   public void bark()
```

```
   {
      System.out.println("woof");
   }
}
```
What code should be inserted?

A. 5.class Dog {
   9. public class Poodle extends Dog {
B. 5.abstract Dog {
   9. public class poodle extends Dog {
C. 5.abstract class Dog {
   9. public class Poodle extends Dog {
D. 5.abstract Dog {
   9.public class Poodle implements Dog {
E. 5. abstractDog {
   9. public class Poodle implements Dog {
F. 5.abstract class Dog {
   9.public class Poodle implements Dog {

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 31
Given:

```
class X
{
}

class Y
{
   Y()
   {
   }
}

class Z
{
   Z(int i)
   {
   }
}
```
Which class has a default constructor?

A. X only
B. Y only
C. Z only
D. X and Y
E. Y and Z
F. X and Z
G. X, Y and Z

Correct Answer: A

Section: (none)
Explanation

Explanation/Reference:

QUESTION 32
Given:
Public static void main (String [] args) {
int a, b, c = 0;
int a = b = c = 0;
int g, int h, int i = 0;
int d, e, F;
int k, l, m = 0;
Which three declarations will compile?

A. int a, b, c = 0;
B. int a = b = c = 0;
C. int g, int h, int i = 0;
D. int d, e, F;
E. int k, l, m = 0;

Correct Answer: ADE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 33
Given the code fragment:

```
int j = 0, k = 0;
for (int i = 0; i < x; i++)
{
   do
   {
      k = 0;
      while (k < z)
      {
         k++;
         System.out.print(k + " ");
      }
      System.out.println(" ");
      j++;
   }
   while (j < y);
   System.out.println("----");
}
```

What values of x, y, z will produce the following result?

```
1 2 3
1 2 3
1 2 3
1 2 3
----
1 2 3
----
```

A. X = 4, Y = 3, Z = 2

B. X = 3, Y = 2, Z = 3
C. X = 2, Y = 3, Z = 3
D. X = 4, Y = 2, Z= 3
E. X = 2, Y = 3, Z = 4

Correct Answer: E
Section: (none)
Explanation

Explanation/Reference:

QUESTION 34
Which statement initializes a stringBuilder to a capacity of 128?

A. StringBuilder sb = new String("128");
B. StringBuilder sb = StringBuilder.setCapacity(128);
C. StringBuilder sb = StringBuilder.getInstance(128);
D. StringBuilder sb = new StringBuilder(128);

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 35
Given:

```
public class DoCompare4
{
   public static void main(String[] args)
   {
      String[] table = {"aa", "bb", "cc"};
      int ii = 0;
      do
         while (ii < table.length)
            System.out.println(ii++);
      while (ii < table.length);
   }
}
```

What is the result?

A. 0
B. 0
   1
   2
C. 0
   1
   2
   0
   1
   2
   0
   1
   2

D. Compilation fails

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 36
A method is declared to take three arguments. A program calls this method and passes only two arguments. What is the result?

A. Compilation fails.
B. The third argument is given the value null.
C. The third argument is given the value void.
D. The third argument is given the value zero.
E. The third argument is given the appropriate false value for its declared type.
F. An exception occurs when the method attempts to access the third argument.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 37
Given the fragment:

```
int[] array = {1, 2, 3, 4, 5};
System.arraycopy(array, 2, array, 1, 2);
System.out.print(array[1]);
System.out.print(array[4]);
```

What is the result?

A. 14
B. 15
C. 24
D. 25
E. 34
F. 35

Correct Answer: F
Section: (none)
Explanation

Explanation/Reference:

QUESTION 38
Given the following code fragment:

```
if (value >= 0)
{
if (value != 0)
System.out.print("the ");
else
```

```
System.out.print("quick ");
if (value < 10)
System.out.print("brown ");
if (value > 30)
System.out.print("fox ");
else if (value < 50)
System.out.print("jumps ");
else if (value < 10)
System.out.print("over ");
else
System.out.print("the ");
if (value > 10)
System.out.print("lazy ");
}
else
{
System.out.print("dog ");
}
System.out.print("... ");
```

What is the result if the integer value is 33?

A. The fox jump lazy ...
B. The fox lazy ...
C. Quick fox over lazy ...
D. Quick fox the ...

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 39
Which three are advantages of the Java exception mechanism?

A. Improves the program structure because the error handling code is separated from the normal program function
B. Provides a set of standard exceptions that covers all the possible errors
C. Improves the program structure because the programmer can choose where to handle exceptions
D. Improves the program structure because exceptions must be handled in the method in which they occurred
E. Allows the creation of new exceptions that are tailored to the particular program being

Correct Answer: ACE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 40
Given:

```
public class MyFor3
{
   public static void main(String[] args)
   {
```

```
        int[] xx = null;
        System.out.println(xx);
    }
}
```
What is the result?

A. null
B. compilation fails
C. Java.lang.NullPointerException
D. 0

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 41
Given:
```
public class Main {
public static void main (String[] args) {
doSomething();
}
private static void doSomething() {
doSomeThingElse();
}
private static void doSomeThingElse() {
throw new Exception();
}
}
```
Which approach ensures that the class can be compiled and run?

A. Put the throw new Exception() statement in the try catch block
B. Put the invocation of doSomethingElse() method in the try catch block
C. Put the invocation of doSomething() method in the try catch block
D. Put the invocations of both doSomething() and the doSomethingElse() methods in the try catch block

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 42
Given:
```
public class ScopeTest1
{
    public static void main(String[] args)
```

```
    {
        doStuff(); // line x1
        int x1 = x2; // line x2
        int x2 = j; // line x3
    }

    static void doStuff()
    {
        System.out.println(j); // line x4
    }

    static int j;
}
```

Which line causes a compilation error?

A. line x1
B. line x2
C. line x3
D. line x4

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 43

Given:

```
class Overloading
{
    void x(int i)
    {
        System.out.println("one");
    }

    void x(String s)
    {
        System.out.println("two");
    }

    void x(double d)
    {
        System.out.println("three");
    }

    public static void main(String[] args)
    {
        new Overloading().x(4.0);
    }
}
```

What is the result?

A. one
B. two
C. three
D. Compilation fails

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 44
Which declaration initializes a boolean variable?

A. boolean h = 1;
B. boolean k = 0;
C. boolean m = null;
D. boolean j = (1 < 5) ;

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 45
Given:

```
public class Basic
{
   private static int letter;

   public static int getLetter();

   public static void Main(String[] args)
   {
      System.out.println(getLetter());
   }
}
```

Why will the code not compile?

A. A static field cannot be private.
B. The getLetter method has no body.
C. There is no setletter method.
D. The letter field is uninitialized.
E. It contains a method named Main instead of ma

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 46
Given:

```
public class Circle
{
   double radius;
   public double area
```

```
    public Circle(double r)
    {
       radius = r;
    }

    public double getRadius()
    {
       return radius;
    }

    public void setRadius(double r)
    {
       radius = r;
    }

    public double getArea()
    {
       return /* ??? */;
    }
}

class App
{
    public static void main(String[] args)
    {
       Circle c1 = new Circle(17.4);
       c1.area = Math.PI * c1.getRadius() * c1.getRadius();
    }
}
```

This class is poorly encapsulated. You need to change the circle class to compute and return the area instead. What three modifications are necessary to ensure that the class is being properly encapsulated?

A. Change the access modifier of the setRadius () method to private
B. Change the getArea () method
   public double getArea () { return area; }
C. When the radius is set in the Circle constructor and the setRadius () method, recomputed the area and store it into the area field
D. Change the getRadius () method:
   public double getRadius () {
   area = Math.PI * radius * radius;
   return radius;
   }

Correct Answer: ABC
Section: (none)
Explanation

Explanation/Reference:

QUESTION 47
Given a code fragment:

```
    StringBuilder sb = new StringBuilder();
    String h1 = "HelloWorld";
    sb.append("Hello").append("world");
    if (h1 == sb.toString())
    {
       System.out.println("They match");
    }
```

```
if (h1.equals(sb.toString()))
{
   System.out.println("They really match");
}
```
What is the result?

A. They match
   They really match
B. They really match
C. They match
D. Nothing is printed to the screen

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 48
Given the following code:
```
public class Simple
{ /* Line 1 */
   public float price; /* Line 2 */

   public static void main(String[] args)
   { /* Line 3 */
      Simple price = new Simple(); /* Line 4 */
      price = 4; /* Line 5 */
   } /* Line 6 */
} /* Line 7 */
```
What will make this code compile and run?

A. Change line 2 to the following:
   Publicint price
B. Change line 4 to the following:
   int price = new simple ();
C. Change line 4 to the following:
   Floatprice = new simple ();
D. Change line 5 to the following:
   Price = 4f;
E. Change line 5 to the following:
   price.price = 4;
F. Change line 5 to the following:
   Price= (float) 4:
G. Change line 5 to the following:
   Price= (Simple) 4;
H. The code compiles and runs properly; no changes are necessary

Correct Answer: E
Section: (none)
Explanation

Explanation/Reference:

QUESTION 49
Given:

```
public class DoWhile
{
   public static void main(String[] args)
   {
      int ii = 2;
      do
      {
         System.out.println(ii);
      }
      while (--ii);
   }
}
```

What is the result?

A. 2
   1
B. 2
   1
   0
C. null
D. an infinite loop
E. compilation fails

Correct Answer: E
Section: (none)
Explanation

Explanation/Reference:

QUESTION 50
You are writing a method that is declared not to return a value. Which two are permitted in the method body?

A. omission of the return statement
B. return null;
C. return void;
D. return;

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 51
Identify two benefits of using ArrayList over array in software development.

A. reduces memory footprint
B. implements the Collection API
C. is multi.thread safe
D. dynamically resizes based on the number of elements in the list

Correct Answer: BD

Section: (none)
Explanation

Explanation/Reference:
Explanation: ArrayList supports dynamic arrays that can grow as needed. In Java, standard arrays are of a fixed length. After arrays are created, they cannot grow or shrink, which means that you must know in advance how many elements an array will hold. But, sometimes, you may not know until run time precisely how large of an array you need. To handle this situation, the collections framework defines ArrayList. In essence, an ArrayList is a variable-length array of object references. That is, an ArrayList can dynamically increase or decrease in size. Array lists are created with an initial size. When this size is exceeded, the collection is automatically enlarged. When objects are removed, the array may be shrunk.

QUESTION 52
Which three are valid types for switch in Java 7 version?

A. int
B. float
C. double
D. Integer
E. String
F. Float

Correct Answer: ADE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 53
Give:

```
public class MyFive
{
   public static void main(String[] args)
   {
      short ii;
      short jj = 0;
      for (ii = KK; ii > 6; ii -= 1)
      { // line x //
         jj++;
      }
      System.out.println("jj = " + jj);
   }
}
```

What value should replace KK in line x to cause jj = 5 to be output?

A. -1
B. 1
C. 5
D. 8
E. 11

Correct Answer: E
Section: (none)
Explanation

Explanation/Reference:

QUESTION 54
Given the code fragment:

```
Boolean b1 = true;
Boolean b2 = false;
int 1 = 0;
while (foo) {}
```

Which one is valid as a replacement for foo?

A. b1.compareTo(b2)
B. i = 1
C. i == 2? -1:0
D. "foo".equals("bar")

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 55

```
public class SuperTest
{
   public static void main(String[] args)
   {
      statement1
      statement2
      statement3
   }
}

class Shape
{
   public Shape()
   {
      System.out.println("Shape: constructor");
   }

   public void foo()
   {
      System.out.println("Shape: foo");
   }
}

class Square extends Shape
{
   public Square()
   {
      super();
   }

   public Square(String label)
   {
      System.out.println("Square: constructor");
   }

   public void foo()
```

```
    {
        super.foo();
    }

    public void foo(String label)
    {
        System.out.println("Square: foo");
    }
}
```

What should statement1, statement2, and statement3, be respectively, in order to produce the result?
Shape: constructor
Shape: foo
Square: foo

A. Square square = new Square ("bar");
   square.foo ("bar");
   square.foo();
B. Square square = new Square ("bar");
   square.foo ("bar");
   square.foo ("bar");
C. Square square = new Square ();
   square.foo ();
   square.foo("bar");
D. Square square = new Square();
   square.foo("bar");
   square.foo();
E. Square square = new Square();
   square.foo();
   square.foo("bar");

Correct Answer: E
Section: (none)
Explanation

Explanation/Reference:

QUESTION 56
Give:
Public Class Test {
}
Which two packages are automatically imported into the java source file by the java compiler?

A. java.lang
B. java.awt
C. javax.net
D. java.*
E. The package with no name

Correct Answer: AE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 57

Given:

```
public class X implements Z
{
   public String toString()
   {
      return "I am X";
   }

   public static void main(String[] args)
   {
      Y myY = new Y();
      X myX = myY;
      Z myZ = myX;
      System.out.println(myZ);
   }
}

class Y extends X
{
   public String toString()
   {
      return "I am Y";
   }
}

interface Z
{
}
```

What is the reference type of myZ and what is the type of the object it references?

A. Reference type isZ; object type isZ.
B. Reference type isY;object type isY.
C. Reference type isZ;object type is Y.
D. Reference type is X; object type is Z.

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 58

Given:

```
class SampleClass
{
}

class AnotherSampleClass extends SampleClass
{
}

class Test
{
   public static void main(String[] args)
   {
      SampleClass sc = new SampleClass();
      AnotherSampleClass asc = new AnotherSampleClass();
      sc = asc;
```

```
        System.out.println("sc: " + sc.getClass());
        System.out.println("asc: " + asc.getClass());
    }
}
```

What is the result?

A. sc: class.Object
   asc: class.AnotherSampleClass
B. sc: class.SampleClass
   asc: class.AnotherSampleClass
C. sc: class.AnotherSampleClass
   asc: class.SampleClass
D. sc: class AnotherSampleClass
   asc: class AnotherSampleClass

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 59
Given the code fragment:

```
public static void main(String[] args)
{
    String[] table = {"aa", "bb", "cc"};
    int ii = 0;
    for (String ss : table)
    {
        while (ii < table.length)
        {
            System.out.println(ii);
            ii++;
            break;
        }
    }
}
```

How many times is 2 printed?

A. zero
B. once
C. twice
D. thrice
E. it is not printed because compilation fails

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 60
Given:
public class SampleClass

```
{
   public static void main(String[] args)
   {
      SampleClass sc, scA, scB;
      sc = new SampleClass();
      scA = new SampleClassA();
      scB = new SampleClassB();
      System.out.println(
            "Hash is : " +
                  sc.getHash() + ", " + scA.getHash() + ", " + scB.getHash()
      );
   }

   public int getHash()
   {
      return 111111;
   }
}

class SampleClassA extends SampleClass
{
   public long getHash()
   {
      return 44444444;
   }
}

class SampleClassB extends SampleClass
{
   public long getHash()
   {
      return 999999999;
   }
}
```

What is the result?

A. Compilation fails
B. An exception is thrown at runtime
C. There is no result because this is not correct way to determine the hash code
D. Hash is: 111111, 44444444, 999999999

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 61
Which two will compile, and can be run successfully using the command:
Java Fred1 hello walls

A.
```
class Fred1
{
   public static void main(String args)
   {
      System.out.println(args[1]);
   }
}
```

B.
```
class Fred1
{
   public static void main(String[] args)
   {
      System.out.println(args[2]);
   }
}
```
C.
```
class Fred1
{
   public static void main(String[] args)
   {
      System.out.println(args);
   }
}
```
D.
```
class Fred1
{
   public static void main(String[] args)
   {
      System.out.println(args[1]);
   }
}
```

Correct Answer: CD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 62
Given:
```
public abstract class Wow
{
   private int wow;

   public wow(int wow)
   {
      this.wow = wow;
   }

   public void wow()
   {
   }

   private void wowza()
   {
   }
}
```
What is true about the class Wow?

A. It compiles without error.
B. It does not compile because an invalid method declaration
C. It does not compile because an abstract class cannot have instance variables.
D. It does not compile because an abstract class must have at least one abstract method.
E. It does not compile because an abstract class must have a constructor with no arguments.

Correct Answer: B
Section: (none)

Explanation

Explanation/Reference:

QUESTION 63
Given:

```
class X
{
   static void m(int i)
   {
   }

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

What is the result?

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

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 64
Which two statements are true?

A. An abstract class can implement an interface.
B. An abstract class can be extended by an interface.
C. An interface CANNOT be extended by another interface.
   D An interface can be extended by an abstract class.
D. An abstract class can be extended by a concrete class.
E. An abstract class CANNOT be extended by an abstract class.

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 65
Given:

```
class Overloading
{
   int x(double d)
```

```
    {
        System.out.println("one");
        return 0;
    }

    String x(double d)
    {
        System.out.println("two");
        return null;
    }

    double x(double d)
    {
        System.out.println("three");
        return 0.0;
    }

    public static void main(String[] args)
    {
        new Overloading().x(4.0)
    }
}
```

What is the result?

A. One
B. Two
C. Three
D. Compilation fails

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 66
The catch clause argument is always of type_____.

A. Exception
B. Exception but NOT including RuntimeException
C. Throwable
D. RuntimeException
E. CheckedException
F. Error

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 67
Given the code fragment:

```
ArrayList<Integer> list = new ArrayList<>(1);//line 1
list.add(1001);//line 2
```

```
        list.add(1002);//line 3
        System.out.println(list.get(list.size()));//line 4
```
What is the result?

A. Compilation fails due to an error on line 1.
B. An exception is thrown at run time due to error on line 3
C. An exception is thrown at run time due to error on line 4
D. 1002

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
Explanation:
The code compiles fine.
At runtime an IndexOutOfBoundsException is thrown when the second list item is added.

QUESTION 68
View the Exhibit.

```
public class Hat
{
   public int ID = 0;
   public String name = "hat";
   public String size = "One Size Fit All";
   public String color = "";

   public String getName()
   {
      return name;
   }

   public void setName(String name)
   {
      this.name = name;
   }
}

class TestHat
{
   public static void main(String[] args)
   {
      Hat blackCowboyHat = new Hat();
   }
}
```

Which statement sets the name of the Hat instance?

A. blackCowboyHat.setName = "Cowboy Hat";
B. setName("Cowboy Hat");
C. Hat.setName("Cowboy Hat");
D. blackCowboyHat.setName("Cowboy Hat");

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 69

```
public class Two
{
   public static void main(String[] args)
   {
      try
      {
         doStuff();
         System.out.println("1");
      }
      catch (Exception e)
      {
         System.out.println("2");
      }
   }

   public static void doStuff()
   {
      if (Math.random() > 0.5)
         throw new RuntimeException();
      doMoreStuff();
      System.out.println("3 ");
   }

   public static void doMoreStuff()
   {
      System.out.println("4");
   }
}
```

Which two are possible outputs?

A. 2
B. 4 3 1
C. 1 2 3
D. 1 3 4

Correct Answer: AB
Section: (none)
Explanation

Explanation/Reference:

QUESTION 70

Given:

```
public class MyFor
{
   public static void main(String[] args)
   {
      for (int ii = 0; ii < 4; ii++)
      {
         System.out.println("ii = " + ii);
         ii = ii + 1;
      }
   }
}
```

What is the result?

A. ii = 0
   ii = 2
B. ii = 0
   ii = 1
   ii = 2
   ii = 3
C. ii =
D. Compilation fails.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 71
Given the code fragment:

```
int[][] array2d = new int[2][3];
System.out.println("Loading the data.");
for (int x = 0; x < array2d.length; x++)
{
   for (int y = 0; y < array2d[0].length; y++)
   {
      System.out.println(" x = " + x);
      System.out.println(" y = " + y);
   // insert load statement here.
   }
}
System.out.println("Modify the data. ");
for (int x = 0; x < array2d.length; x++)
{
   for (int y = 0; y < array2d[0].length; y++)
   {
      System.out.println(" x = " + x);
      System.out.println(" y = " + y);
      // insert modify statement here.
   }
}
```

Which pair of load and modify statement should be inserted in the code? The load statement should set the array's x row and y column value to the sum of x and y
The modify statement should modify the array's x row and y column value by multiplying it by 2

A. Load statement: array2d(x,y) = x + y;
   Modify statement: array2d(x,y) = array2d(x,y) * 2
B. B. Load statement: array2d[x y] = x + y;
   Modify statement: array2d[x y] = array2d[x y] * 2
C. Load statement: array2d[x,y] = x + y;
   Modify statement: array2d[x,y] = array2d[x,y] * 2
D. Load statement: array2d[x][y] = x + y;
   Modify statement: array2d[x][y] = array2d[x][y] * 2
E. Load statement: array2d[[x][y]] = x + y;
   Modify statement: array2d[[x][y]] = array2d[[x][y]] * 2

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 72
Given:

```
public class DoBreak1
{
   public static void main(String[] args)
   {
      String[] table = {
            "aa", "bb", "cc",
            "dd"
      };
      for (String ss : table)
      {
         if ("bb".equals(ss))
         {
            continue;
         }
         System.out.println(ss);
         if ("cc".equals(ss))
         {
            break;
         }
      }
   }
}
```

What is the result?

A. aa
   cc
B. aa
   bb
   cc
C. cc
   dd
D. cc
E. Compilation fails.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 73

```
1. class StaticMethods {
2. static void one() {
3. two();
4. StaticMethods.two();
5. three();
6. StaticMethods.four();
7. }
8. static void two() { }
9. void three() {
10. one();
11. StaticMethods.two();
```

```
12. four();
13. StaticMethods.four();
14. }
15. void four() { }
16. }
```
Which three lines are illegal?

A. line 3
B. line 4
C. line 5
D. line 6
E. line 10
F. line 11
G. line 12
H. line 13

Correct Answer: CDH
Section: (none)
Explanation

QUESTION 74
Which is a valid abstract class?

A.
```
public abstract class Car {
protected void accelerate();
}
```
B.
```
public interface Car {
protected abstract void accelerate();
}
```
C.
```
public abstract class Car {
protected final void accelerate();
}
```
D.
```
public abstract class Car {
protected abstract void accelerate();
}
```
E.
```
public abstract class Car {
protected abstract void accelerate() {
//more car can do
}}
```

Correct Answer: D
Section: (none)
Explanation

QUESTION 75
```
public class Student
{
   public String name =
          "";
   public int age = 0;
   public String major = "Undeclared";
   public boolean fulltime = true;

   public void display()
```

```
    {
        System.out.println("Name: " + name + " Major: " + major);
    }

    public boolean isFullTime()
    {
        return fulltime;
    }
}

class TestStudent
{
    public static void main(String[] args)
    {
        Student bob = new Student();
        bob.name = "Bob";
        bob.age = 18;
        bob.year = 1982;
    }
}
```

What is the result?

A. year is set to 1982.
B. bob.year is set to 1982
C. A runtime error is generated.
D. A compile time error is generated.

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 76
Given the code fragment:

```
String name = "Spot";
int age = 4;
String str = "My dog " + name + " is " + age;
System.out.println(str);
StringBuilder sb = new StringBuilder();
```

Using StringBuilder, which two code fragments are the best potion to build and print the following string My dog Spot is 4

A. sb.append("My dog " + name + " is " + age);
   System.out.println(sb);
B. sb.insert("My dog ").append(name + " is " + age); System.out.println(sb);
C. sb.insert("My dog ").insert(name).insert(" is ").insert(age); System.out.println(sb);
D. sb.append("My dog ").append(name).append(" is ").append(age); System.out.println(sb);

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 77

Given:

```
public class Main
{
   public static void main(String[] args)
   {
      try
      {
         doSomething();
      }
      catch (SpecialException e)
      {
         System.out.println(e);
      }
   }

   static void doSomething()
   {
      int[] ages = new int[4];
      ages[4] = 17;
      doSomethingElse();
   }

   static void doSomethingElse()
   {
      throw new SpecialException("Thrown at end of doSomething() method");
   }
}

class SpecialException extends RuntimeException
{
   public SpecialException(String string)
   {
   }
}
```

What is the output?

A. SpecialException: Thrown at end of doSomething() method
B. Error in thread "main" java.lang.
   ArrayIndexOutOfBoundseror
C. C. Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 4
   at Main.doSomething(Main.java:12)
   at Main.main(Main.java:4)
D. D. SpecialException: Thrown at end of doSomething() method at
   Main.doSomethingElse(Main.java:16)
   at Main.doSomething(Main.java:13)
   at Main.main(Main.java:4)

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 78
View the exhibit:

```
public class Student
{
   public String name = "";
```

```
    public int age = 0;
    public String major = "Undeclared";
    public boolean fulltime = true;

    public void display()
    {
        System.out.println("Name: " + name + " Major: " + major);
    }

    public boolean isFullTime()
    {
        return fulltime;
    }
}
```

Which line of code initializes a student instance?

A. Student student1;
B. Student student1 = Student.new();
C. Student student1 = new Student();
D. Student student1 = Student();

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 79

```
int[] array = {1, 2, 3, 4, 5};
for (int i : array)
{
    if (i < 2)
    {
        keyword1;
    }
    System.out.println(i);
    if (i == 3)
    {
        keyword2;
    }
}
```

What should keyword1 and keyword2 be respectively, in oreder to produce output 2345?

A. continue, break
B. break, break
C. break, continue
D. continue, continue

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 80

```
int i, j=0;
```

```
i = (3* 2 +4 +5 ) ;
j = (3 * ((2+4) + 5));
System.out.println("i:"+ i + "\nj":+j);
```
What is the result?

A. i:16 j:16
B. 16
C. i:15 j:33
D. 33
E. i:16 j:33
F. 15
G. i:15 j:16
H. 23

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 81
```
boolean log3 = (5.0 != 6.0) && (4 != 5);
boolean log4 = (4 != 4) || (4 == 4);
System.out.println("log3:" + log3 + "\nlog4:" + log4);
```
What is the result?

A. log3:false
log4:true
B. log3:true
log4:true
C. log3:true
log4:false
D. log3:false
log4:false

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 82
Which statement will emoty 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();

Correct Answer: C
Section: (none)
Explanation

QUESTION 83

```
class StaticField
{
   static int i = 7;

   public static void main(String[] args)
   {
      StaticField obj = new StaticField();
      obj.i++;
      StaticField.i++;
      obj.i++;
      System.out.println(StaticField.i + " " + obj.i);
   }
}
```

What is the result?

A. 10 10
B. 8 9
C. 9 8
D. 7 10

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 84
Which two are valid array declaration?

A. Object array[];
B. Boolean array[3];
C. int[] array;
D. Float[2] array;

Correct Answer: AC
Section: (none)
Explanation

QUESTION 85
Given:

```
class Overloading
{
   int x(double d)
   {
      System.out.println("one");
      return 0;
   }

   String x(double d)
   {
      System.out.println("two");
      return null;
   }
```

```
    double x(double d)
    {
        System.out.println("three");
        return 0.0;
    }

    public static void main(String[] args)
    {
        new Overloading().x(4.0);
    }
}
```
What is the result?

A. one
B. two
C. three
D. Compilation fails.

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 86
Given:
```
public class MainMethod
{
    void main()
    {
        System.out.println("one");
    }

    static void main(String args)
    {
        System.out.println("two");
    }

    public static void main(String[] args)
    {
        System.out.println("three");
    }

    void main(Object[] args)
    {
        System.out.println("four");
    }
}
```
What is printed out when the program is excuted?

A. one
B. two
C. three
D. four

Correct Answer: C
Section: (none)

Explanation

Explanation/Reference:

QUESTION 87
Given:

```
public class ScopeTest
{
   int j, k;

   public static void main(String[] args)
   {
      new ScopeTest().doStuff();
   }

   void doStuff()
   {
      int x = 5;
      doStuff2();
      System.out.println("x");
   }

   void doStuff2()
   {
      int y = 7;
      System.out.println("y");
      for (int z = 0; z < 5; z++)
      {
         System.out.println("z");
         System.out.println("y");
      }
   }
}
```

Which two items are fields?

A. j
B. k
C. x
D. y
E. z

Correct Answer: AB
Section: (none)
Explanation

Explanation/Reference:

QUESTION 88
A method is declared to take three arguments. A program calls this method and passes only two arguments. What is the results?

A. Compilation fails.
B. The third argument is given the value null.
C. The third argument is given the value void.
D. The third argument is given the value zero.

E. The third argument is given the appropriate falsy value for its declared type. F) An exception occurs when the method attempts to access the third argument.

Correct Answer: A
Section: (none)
Explanation

QUESTION 89

```
public class ForTest
{
   public static void main(String[] args)
   {
      int[] arrar = {1, 2, 3};
      for (foo){ }
   }
}
```

Which three are valid replacements for foo so that the program will compiled and run?

A. int i: array
B. int i = 0; i < 1; i++
C. ;;
D. ; i < 1; i++
E. ; i < 1;

Correct Answer: ABC
Section: (none)
Explanation

Explanation/Reference:

QUESTION 90

Given:

```
public class SampleClass
{
   public static void main(String[] args)
   {
      AnotherSampleClass asc = new AnotherSampleClass();
      SampleClass sc = new SampleClass();
      sc = asc;
      System.out.println("sc: " + sc.getClass());
      System.out.println("asc: " + asc.getClass());
   }
}

class AnotherSampleClass extends SampleClass
{
}
```

What is the result?

A. sc: class Object
   asc: class AnotherSampleClass
B. sc: class SampleClass
   asc: class AnotherSampleClass
C. sc: class AnotherSampleClass
   asc: class SampleClass

D. sc: class AnotherSampleClass
asc: class AnotherSampleClass

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference: