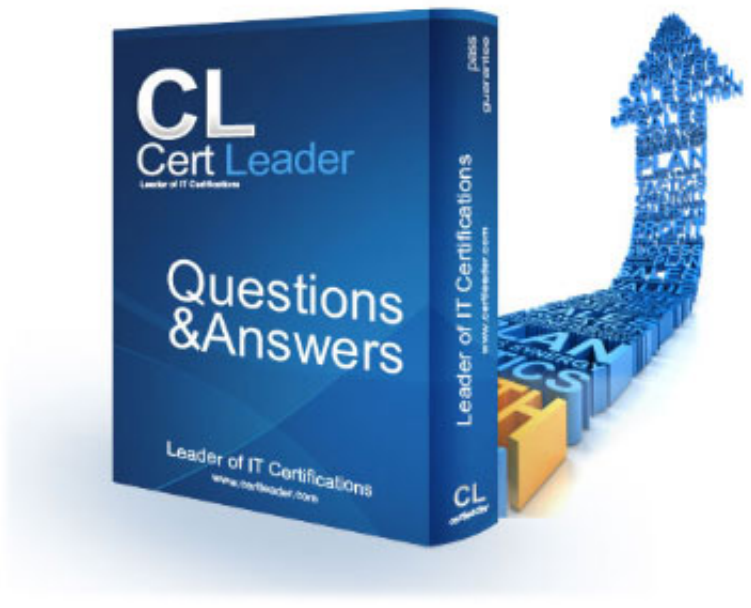
1Z0-803 Dumps

Java SE 7 Programmer I

https://www.certleader.com/1Z0-803-dumps.html



NEW QUESTION 1
Which code fragment cause a compilation error?

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

Answer: B

NEW QUESTION 2
Given the code format:

```
class DBConfiguration {
    String user;
    String password;
}

And:

4. public class DBHandler {
5.     DBConfiguration configureDB(String uname, String password) {
6.         // insert code here
7.     }
8.     public static void main(String[] args) {
9.         DBHandler r = new DBHandler();
10.        DBConfiguration dbConf = r.configureDB("manager", "manager");
11.    }
12. }
```

Which code fragment must be inserted at line 6 to enable the code to compile?

A. DBConfiguration f; return f;
B. Return DBConfiguration;
C. Return new DBConfiguration;
D. Retutn 0;

Answer: B

NEW QUESTION 3
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 Objectasc: class AnotherSampleClass
B. sc: class SampleClassasc: class AnotherSampleClass
C. sc: class AnotherSampleClass asc: class SampleClass
D. sc: class AnotherSampleClassasc: class AnotherSampleClass

Answer: D

NEW QUESTION 4
Given the code fragment:
System.out.println(2 + 4 * 9 - 3); //Line 21
System.out.println((2 + 4) * 9 - 3); // Line 22
System.out.println(2 + (4 * 9) - 3); // Line 23
System.out.println(2 + 4 * (9 - 3)); // Line 24
System.out.println((2 + 4 * 9) - 3); // Line 25 Which line of codes prints the highest number?

A. Line 21
B. Line 22
C. Line 23
D. Line 24
E. Line 25

Answer: B

Explanation: The following is printed: 35
51
35
26
35

NEW QUESTION 5
Which two are valid array declaration?

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

Answer: AC

NEW QUESTION 6
Given the code fragment:

```
if (aVar++ < 10) {
    System.out.println(aVar + " Hello World!");
} else {
    System.out.println(aVar + " Hello Universe!");
}
```

What is the result if the integer aVar is 9?

A. 10 Hello world!
B. 10 Hello universe!
C. 9 Hello world!
D. Compilation fails.

Answer: A

NEW QUESTION 7
Given the code fragment

```
class Test2 {
int fvar;
static int cvar;
    public static void main(String[] args) {
    Test2 t = new Test2();
    // insert code here to write field variables
     }
   }
```

Which code fragments, inserted independently, enable the code compile?

A. t.fvar = 200;
B. cvar = 400;
C. fvar = 200; cvar = 400;
D. this.fvar = 200; this.cvar = 400;
E. t.fvar = 200; Test2.cvar = 400;
F. this.fvar = 200; Test2.cvar = 400;

Answer: B

NEW QUESTION 8
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;
    }

}
```

Given:

```
public 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

Answer: B

Explanation: After the statement jian = bob; the jian will reference the same object as bob.

NEW QUESTION 9
Given:

```
public class Series {
    public static void main(String[] args) {
        int arr[] = {1, 2, 3};

        for (int var : arr) {
            int i = 1;
            while (i <= var);
                System.out.println(i++);
        }
    }
}
```

What is the result?

A. 111
B. 123
C. 234
D. Compilation fails
E. The loop executes infinite times

Answer: E

NEW QUESTION 10
Given:

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

Answer: B

NEW QUESTION 10
Given:
public class Test1 {
static void doubling (Integer ref, int pv) { ref =20;
pv = 20;
}
public static void main(String[] args) { Integer iObj= new Integer(10);
int iVar = 10; doubling(iObj++, iVar++);
System.out.println(iObj+ ", "+iVar); What is the result?

A. 11, 11
B. 10, 10
C. 21, 11
D. 20, 20
E. 11, 12

Answer: A

Explanation: The code doubling(iObj++, iVar++); increases both variables from to 10 to 11.

NEW QUESTION 14
Given:

```
Given:
class X {
     public void mX() {
          System.out.println("Xm1");
     }
}
class Y extends X {
     public void mX() {
          System.out.println("Xm2");
     }
     public void mY() {
          System.out.println("Ym");
     }
}

public class Test {
     public static void main(String[] args) {
       X xRef = new Y();
       Y yRef = (Y) xRef;
       yRef.mY();
       xRef.mX();
     }
}
```

A. YmXm2
B. YmXm1
C. Compilation fails
D. A ClassCastException is thrown at runtime

Answer: B

NEW QUESTION 19
Given the code fragment:
interface SampleClosable {
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 implements SampleCloseable {
       public void close() throws java.io.FileNotFoundException
           // do something
       }
   }

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

E) public class Test implements SampleCloseable {
       public void close() {
           // do something
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E

Answer: ACE

Explanation: A: Throwing the same exception is fine.
C: Using a subclass of java.io.IOException (here java.io.FileNotFoundException) is fine E: Not using a throw clause is fine.

NEW QUESTION 22
Given:

```
1.  public class Speak {
2.      public static void main(String[] args){
3.          Speak speakIt = new Tell();
4.          Tell tellIt = new Tell();
5.          speakIt.tellItLikeItIs();
6.          (Truth)speakIt.tellItLikeItIs();
7.          ((Truth)speakIt).tellItLikeItIs();
8.          tellIt.tellItLikeItIs();
9.          (Truth)tellIt.tellItLikeItIs();
10.         ((Truth)tellIt).tellItLikeItIs();
11.     }
12. }
13. class Tell extends Speak implements Truth {
14.     public void tellItLikeItIs() {
15.         System.out.println("Right on!");
16.     }
17. }
18. interface Truth { public void tellItLikeItIs(); }
```

Which three lines will compile and output “right on!”?

A. Line 5
B. Line 6
C. Line 7
D. Line 8
E. Line 9
F. Line 10

Answer: CDF

NEW QUESTION 27
Given:
class Cake { int model; String flavor; Cake() { model = 0;
flavor = "Unknown";
}
}
public class Test {
public static void main(String[] args) { Cake c = new Cake();
bake1(c);
System.out.println(c.model + " " + c.flavor); bake2(c);
System.out.println(c.model + " " + c.flavor);
}
public static Cake bake1(Cake c) { c.flavor = "Strawberry";

A. c.model = 1200; return c;}public static void bake2(Cake c) {c.flavor = "Chocolate"; c.model = 1230; return;}}What is the result?
B. 0 unknown0 unknown
C. 1200 Strawberry1200 Strawberry
D. 1200 Strawberry1230 Chocolate
E. Compilation fails

Answer: C

Explanation: 1200 Strawberry
1230 Chocolate

NEW QUESTION 31
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. Java.util
D. Javax.net
E. Java.*
F. The package with no name

Answer: AF

Explanation: For convenience, the Java compiler automatically imports three entire packages for each source file: (1) the package with no name, (2) the java.lang package, and (3) the current package (the package for the current file).
Note:Packages in the Java language itself begin withjava.orjavax.

NEW QUESTION 32
The catch clause argument is always of type _____.

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

Answer: C

Explanation: Because all exceptions in Java are the sub-class ofjava.lang.Exceptionclass, you can have a singlecatch blockthat catches an exception of typeExceptiononly. Hence the compiler is fooled into thinking that this block canhandle any exception.
See the following example:
try
{
// ...
}
catch(Exception ex)
{
// Exception handling code for ANY exception
}
You can also use the java.lang.Throwable class here, since Throwable is the parent class for the application-specificException classes. However, this is discouraged in Java programming circles. This is because Throwable happens to also be the parent class for the non-application specific Error classes which are not meant to be handled explicitly as they are catered forby the JVM itself.
Note: The Throwable class is the superclass of all errors and exceptions in the Java language. Only objects that are instances of this class (or one of its subclasses) are thrown by the Java Virtual Machine or can be thrown by the Java throw statement.
A throwable contains a snapshot of the execution stack of its thread at the time it was created. It can also contain a message string that gives more information about the error.

NEW QUESTION 34
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 is Z; object type is Z.
B. Reference type is Y; object type is Y.
C. Reference type is Z; object type is Y.
D. Reference type is X; object type is Z.

Answer: C

NEW QUESTION 35
Which two statements are true for a two-dimensional array?

A. It is implemented as an array of the specified element type.
B. Using a row by column convention, each row of a two-dimensional array must be of the same size.
C. At declarationtime, the number of elements of the array in each dimension must be specified.
D. All methods of the class Object may be invoked on the two-dimensional array.

Answer: AD

NEW QUESTION 38
Which two statements are true for a two-dimensional array of primitive data type?

A. It cannot contain elements of different types.
B. The length of each dimension must be the same.
C. At the declaration time, the number of elements of the array in each dimension must be specified.
D. All methods of the class object may be invoked on the two-dimensional array.

Answer: CD

Explanation: http://stackoverflow.com/questions/12806739/is-an-array-a-primitive-type-or-an-object-or-something-else-entirely

NEW QUESTION 39
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 ofx,y,zwill produce the following result?
1 2 3 4
1 2 3 4
1 2 3 4
---
1 2 3 4

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

Answer: E

Explanation: Z is for the innermost loop. Should print 1 2 3 4. So Z must be 4.
Y is for the middle loop. Should print three lines of 1 2 3 4. So Ymust be set 3. X is for the outmost loop. Should print 2 lines of. So X should be 2.

NEW QUESTION 41
Given:

```
1. public class TestLoop {
2.     public static void main(String[] args) {
3.         float myarray[] = {10.20f, 20.30f, 30.40f, 50.60f};
4.         int index = 0;
5.         boolean isFound = false;
6.         float key = 30.40f;
7.        // insert code here
8.         System.out.println(isFound);
9.     }
10. }
```

Which code fragment, when inserted at line 7, enables the code print true?

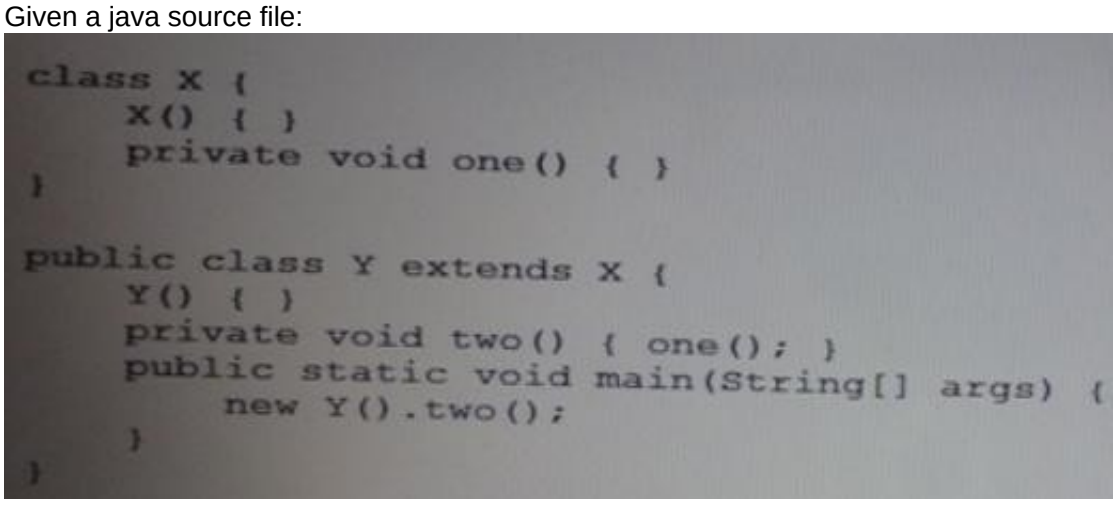
```
A) while (key == myarray[index++]) {
       isFound = true;
   }

B) while (index <= 4) {
       if (key == myarray[index]) {
           index++;
           isFound = true;
           break;
       }
   }

C) while (index++ < 5) {
       if (key == myarray[index]) {
           isFound = true;
       }
   }

D) while (index < 5) {
       if (key == myarray[index]) {
           isFound = true;
           break;
       }
       index++;
   }
```

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

Answer: A

NEW QUESTION 43

Given a java source file:



What changes will make this code compile? (Select Two)

A. Adding the public modifier to the declaration of class 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

Answer: CE

Explanation: Using the private protected, instead of the private modifier, for the declaration of the one() method, would enable the two() method to access the one() method.

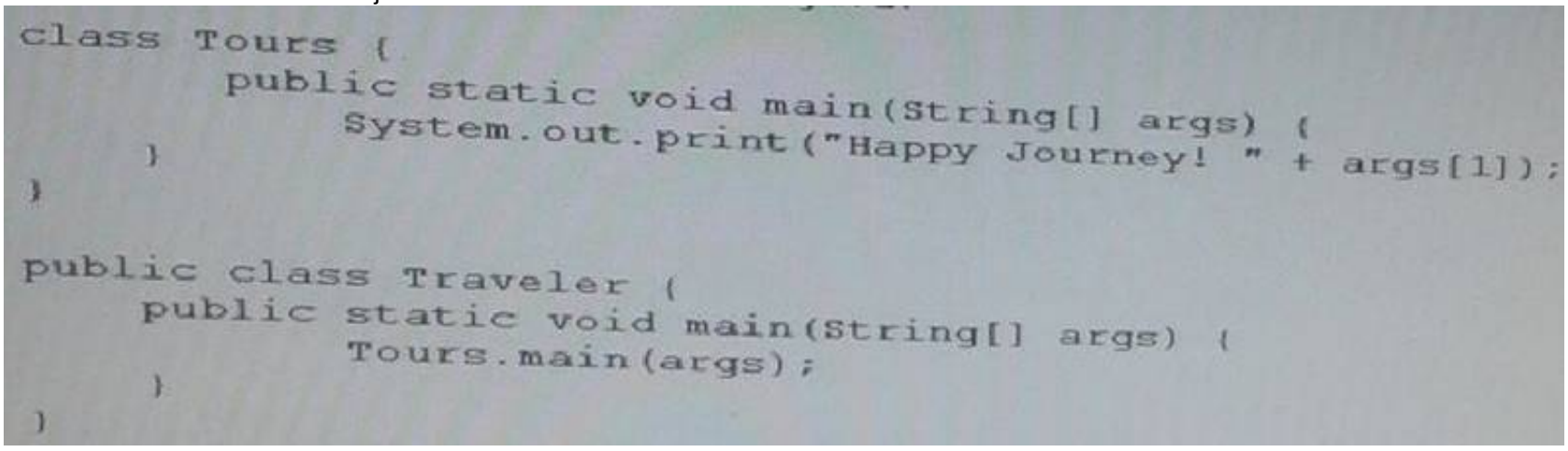
NEW QUESTION 46
Given the for loop construct:
for (expr1 ; expr2 ; expr3) { statement;
}
Which two statements aretrue?

A. This is not the only valid for loop construct; there exits another form of for loop constructor.
B. The expression expr1 is optiona
C. it initializes the loop and is evaluated once, as the loop begin.
D. When expr2 evaluates to false, the loop terminate
E. It is evaluated only after each iteration through the loop.
F. The expression expr3 must be presen
G. It is evaluated after each iteration through the loop.

Answer: BC

Explanation: The for statement have this forms: for (init-stmt;condition;next-stmt) { body
}
There are three clauses in the for statement.
The init-stmt statement is done before the loop is started, usually to initialize an iteration variable.
The condition expression is tested before each time the loop is done. The loop isn't executed if the boolean expression is false (the same as the while loop).
The next-stmt statement is done after the body is executed. It typically increments an iteration variable.

NEW QUESTION 49
Given the code in a file Traveler.java:



And the commands:
Javac Traveler.java
Java Traveler Java Duke What is the result?

A. Happy Journey! Duke
B. Happy Journey! Java
C. An exception is thrown at runtime
D. The program fails to execute due to a runtime error

Answer: D

NEW QUESTION 51
Which two items can legally be contained within a java class declaration?

A. An import statement
B. A field declaration
C. A package declaration
D. A method declaration

Answer: BD

Explanation: Reference: http://docs.oracle.com/javase/tutorial/java/javaOO/methods.html

NEW QUESTION 52
Given:
class Base {
public static void main(String[] args) { System.out.println("Base " + args[2]);
}
}
public class Sub extends Base{
public static void main(String[] args) { System.out.println("Overriden " + args[1]);
}
}
And the commands: javac Sub.java
java Sub 10 20 30 What is the result?

A. Base 30
B. Overridden 20
C. Overridden 20Base 30
D. Base 30Overridden 20

Answer: B

NEW QUESTION 55
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}}

Answer: D

NEW QUESTION 60
Class StaticField { static int i = 7;
public static void main(String[] args) { StaticFied 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

Answer: A

NEW QUESTION 64
The protected modifier on a Field declaration within a public class means that the field _____.

A. Cannot be modified
B. Can be read but not written from outside the class
C. Can be read and written from this class and its subclasses only within the same package
D. Can be read and written from this class and its subclasses defined in any package

Answer: D

Explanation: Reference:
http://beginnersbook.com/2013/05/java-access-modifiers/

NEW QUESTION 66
Given:

```
7.   StringBuilder sb1 = new StringBuilder("Duke");
8.   String str1 = sb1.toString();
9.   // insert code here
10.    System.out.print(str1 == str2);
```

Which code fragment, when inserted at line 9, enables the code to print true?

A. String str2 =str1;
B. String str2 = new string (str1);
C. String str2 = sb1.toString();
D. String str2 = “Duke”;

Answer: B

NEW QUESTION 70
Given:

```
public class Test {
   public static void main(String[] args) {
      Test ts = new Test();
      System.out.print(isAvailable + " ");
      isAvailable= ts.doStuff();
      System.out.println(isAvailable);
   }
   public static boolean doStuff() {
      return !isAvailable;
   }
   static boolean isAvailable = false;
}
```

What is the result?

A. true true
B. true false
C. false true
D. false false
E. Compilation fails

Answer: E

NEW QUESTION 74
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.

Answer: D

NEW QUESTION 76
Given the code fragment:

```
int[] lst = {1, 2, 3, 4, 5, 4, 3, 2, 1};
int sum = 0;
for (int frnt = 0, rear = lst.length - 1;
     frnt < 5 && rear >= 5;
     frnt++, rear--) {
   sum = sum + lst[frnt] + lst[rear];
}
System.out.print(sum);
```

What is the result?

A. 20
B. 25
C. 29
D. Compilation fails
E. AnArrayIndexOutOfBoundsException is thrown at runtime

Answer: A

NEW QUESTION 78
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 valuefor its declared type.
F. An exception occurs when the method attempts to access the third argument.

Answer: A

Explanation: The problem is noticed at build/compile time. At build you would receive an error message like:
required: int,int,int found: int,int

NEW QUESTION 81
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 block of try – catch
B. Put the doSomethingElse() method in the try block of a try – catch
C. Put the doSomething() method in the try block of a try – catch
D. Put thedoSomething() method and the doSomethingElse() method in the try block of a try – catch

Answer: A

Explanation: We need to catch the exception in the doSomethingElse() method. Such as:
private static void doSomeThingElse() { try {
throw new Exception();} catch (Exception e)
{}
}
Note: One alternative, but not an option here, is the declare the exception in doSomeThingElse and catch it in the doSomeThing method.

NEW QUESTION 82

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

Answer: D

NEW QUESTION 85
public class Two {
public static void main(String[] args) { try {
doStuff(); system.out.println("1");
}
catch { system.out.println("2");
}}
public static void do Stuff() {
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. 431
C. 1
D. 12

Answer: AB

Explanation: A: Output is 2 if Math.random() is greater than 0.5.
B: If Math.random() returns a value less equal to 0.5, the code won't throw an exception, it will continue with the doMore() method which will println "4" after which the program will continue with the doStuff() method and will println "3", after that we will be back in main() and the program will print "1".

NEW QUESTION 90
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 typ
F. F) An exception occurs when the method attempts to access the third argument.

Answer: A

NEW QUESTION 95
Given the following four Java file definitions:
// Foo.java
package facades; public interface Foo { }
// Boo.java package facades;
public interface Boo extends Foo { }
// Woofy.java package org.domain
// line n1
public class Woofy implements Boo, Foo {}
// Test.java package.org; public class Test {
public static void main(String[] args) { Foo obj=new Woofy();
Which set modifications enable the code to compile and run?

A. At line n1, Insert: import facades;At line n2, insert:import facades;importorg.domain;
B. At line n1, Insert: import facades.*;At line n2, insert:import facades;import org.*;
C. At line n1, Insert: import facades.*;At line n2, insert:import facades.Boo;import org.*;
D. At line n1, Insert: import facades.Foo, Boo;At line n2, insert:import org.domain.Woofy;
E. At line n1, Insert: import facades.*;At line n2, insert:import facades;import org.domain.Woofy;

Answer: E

NEW QUESTION 98
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.
E. An abstract class can be extended by a concrete class.
F. An abstract class CANNOT be extended by an abstract class.

Answer: AE

Explanation: http://docs.oracle.com/javase/tutorial/java/IandI/abstract.html

NEW QUESTION 100
Given:

```
Given:

class Caller {
    private void init() {
        System.out.println("Initialized");
    }

    public void start() {
        init();
        System.out.println("Started");
    }
}


public class TestCall {
    public static void main(String[] args) {
        Caller c = new Caller();
        c.start();
        c.init();
    }
}
```

What is the result?

A. Initialized Started
B. Initialized Started Initialized
C. Compilation fails
D. An exception is thrown at runtime

Answer: B

NEW QUESTION 103
Given the code fragment:

```
12. int row = 10;
13. for ( ; row > 0 ; ) {
14.     int col = row;
15.     while (col >= 0) {
16.         System.out.print(col + " ");
17.         col -= 2;
18.     }
19.     row = row / col;
20. }
```

What is the result?

A. 10 8 6 4 2 0
B. 10 8 6 4 2
C. AnArithmeticException is thrown at runtime
D. The program goes into an infinite loop outputting: 10 8 6 4 2 0. . .
E. Compilation fails

Answer: B

NEW QUESTION 108
Given the fragments:

```
public class TestA extends Root {
   public static void main(String[] args) {
      Root r = new TestA();
      System.out.println(r.method1());      // line n1
      System.out.println(r.method2());      // line n2
   }
}
class Root {
   private static final int MAX = 20000;
   private int method1() {
      int a = 100 + MAX;                    // line n3
      return a;
   }
   protected int method2() {
      int a = 200 + MAX;                    // line n4
      return a;
   }
}
```

Which line causes a compilation error?

A. Line n1
B. Line n2
C. Line n3
D. Line n4

Answer: A

NEW QUESTION 113
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 real match
B. They really match
C. They match
D. Nothing is printed to the screen

Answer: B

NEW QUESTION 118
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: 33
B. i: 15
   j: 33
C. i: 33
   j: 23
D. i: 15
   j: 23

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

Answer: B

NEW QUESTION 120
Given:
public class TestOperator {
public static voidmain(String[] args) { int result = 30 -12 / (2*5)+1; System.out.print("Result = " + result);
}
}
What is the result?

A. Result = 2
B. Result = 3
C. Result = 28
D. Result = 29
E. Result = 30

Answer: E

NEW QUESTION 124
Given the code fragment:
for (int ii = 0; ii < 3;ii++) { int count = 0;
for (int jj = 3; jj > 0; jj--) { if (ii == jj) {
++count; break;
}
}
System.out.print(count); continue;
}
What is the result?

A. 011
B. 012
C. 123
D. 000

Answer: A

NEW QUESTION 129
Given:

```
public class CharToStr {
  public static void main(String[] args) {
    String str1 = "Java";
    char str2[] = { 'J', 'a', 'v', 'a' };
    String str3 = null;
      for (char c : str2) {
         str3 = str3 + c;
      }
      if (str1.equals(str3))
        System.out.print("Successful");
      else
        System.out.print("Unsuccessful");
  }
}
```

What is result?

A. Successful
B. Unsuccessful
C. Compilation fails
D. An exception is thrown at runtime

Answer: C

NEW QUESTION 130
Given:
package p1; public class Test {
static double dvalue; static Test ref;
public static void main(String[] args) { System.out.println(ref); System.out.println(dvalue);
}
}
What is the result?

A. p1.Test.class 0.0
B. <the summary address refrenced by ref> 0.000000

C. Null 0.0
D. Compilation fails
E. A NullPointerException is thrown at runtime

Answer: C

NEW QUESTION 131
Given:
public class Test {
public static void main(String[] args) { try {
String[] arr =new String[4]; arr[1] = "Unix";
arr[2] = "Linux"; arr[3] = "Solarios"; for (String var : arr) {
System.out.print(var + " ");
}
} catch(Exception e) { System.out.print (e.getClass());
}
}
}
What is the result?

A. Unix Linux Solaris
B. Null Unix Linux Solaris
C. Class java.lang.Exception
D. Class java.lang.NullPointerException

Answer: B

Explanation: null Unix Linux Solarios
The first element, arr[0], has not been defined.

NEW QUESTION 133
View the exhibit.

```
class MissingInfoException extends Exception { }

class AgeOutofRangeException extends Exception { }

class Candidate {
    String name;
    int age;
    Candidate(String name, int age) throws Exception {
        if (name == null) {
            throw new MissingInfoException();
        } else if (age <= 10 || age >= 150) {
            throw new AgeOutofRangeException();
        } else {
            this.name = name;
            this.age = age;
        }
    }
    public String toString() {
        return name + " age: " + age;
    }
}
```

Given the code fragment:

```
4.  public class Test {
5.      public static void main(String[] args) {
6.          Candidate c = new Candidate("James", 20);
7.          Candidate c1 = new Candidate("Williams", 32);
8.          System.out.println(c);
9.          System.out.println(c1);
10.     }
11. }
```

Which change enables the code to print the following? James age: 20
Williams age: 32

A. Replacing line 5 with public static void main (String [] args) throws MissingInfoException, AgeOutofRangeException {
B. Replacing line 5 with public static void main (String [] args) throws.Exception {
C. Enclosing line 6 and line 7 within a try block and adding: catch(Exception e1) { //code goes here}catch (missingInfoExceptione2) { //code goes here} catch

(AgeOutofRangeException e3) {//code goes here}
D. Enclosing line 6 and line 7 within a try block and adding: catch (missingInfoException e2) { //code goes here}catch (AgeOutofRangeException e3) {//code goes here}

Answer: C

NEW QUESTION 137
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

Answer: B

Explanation: The getLetter() method needs a body public static int getLetter() { }; .

NEW QUESTION 138
Given:

```
public class Test {
public static void main(String[] args) { int arr[] = new int[4];
arr[0] = 1;
arr[1] = 2;
arr[2] = 4;
arr[3] = 5;
int sum = 0; try {
for (int pos = 0; pos <= 4; pos++) { sum = sum +arr[pos];
}
} catch (Exception e) { System.out.println("Invalid index");
}
System.out.println(sum);
}
}
```

What is the result?

A. 12
B. Invalid Index 12
C. Invalid Index
D. Compilation fails

Answer: B

Explanation: The loop (for (int pos = 0; pos <= 4; pos++) {), it should be pos <= 3, causes an exception, which is caught. Then the correct sum is printed.

NEW QUESTION 142
Given:

```
public class TestLoop {
public static void main(String[] args) { int array[] = {0, 1, 2, 3, 4};
int key = 3;
for (int pos = 0; pos < array.length; ++pos) { if (array[pos] == key) {
break;
}
}
System.out.print("Found " + key + "at " + pos);
}
}
```

What is the result?

A. Found 3 at 2
B. Found 3 at 3
C. Compilation fails
D. An exception is thrown at runtime

Answer: C

Explanation: The following linedoes not compile: System.out.print("Found " + key + "at " + pos);
The variable pos is undefined at this line, as its scope is only valid in the for loop. Any variables created inside of a loop are LOCAL TO THE LOOP.

NEW QUESTION 144
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 thepossible 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

Answer: ACE

Explanation: A: The error handling is separated from the normal program logic.
C: You have some choice where to handle the exceptions. E: You can create your own exceptions.

NEW QUESTION 149
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 [];

Answer: AD

Explanation: int[][] array2D; is the standard convention to declare a 2-dimensional integer array. int[] array2D[]; works as well, but it is not recommended.

NEW QUESTION 152
Given the code fragment:

```
String[] colors = {"red", "blue", "green", "yellow","maroon","cyan"};
```

Which code fragment prints blue, cyan, ?

```
A)  for (String c:colors){
        if (c.length() != 4) {
            continue;
        }
    System.out.print(c+", ");
    }

B) for (String c:colors[]) {
        if (c.length() <= 4) {
            continue;
        }
    System.out.print(c+", ");
    }

C) for (String c:String[] colors) {
        if (c.length() >= 3) {
            continue;
        }
    System.out.print(c+", ");
    }

D) for (String c:colors){
        if (c.length() != 4) {
            System.out.print(c+", ");
            continue;
        }
    }
```

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

Answer: A

NEW QUESTION 157
Given:

```
public class TestTry {
    public static void main(String[] args) {
        StringBuilder message = new StringBuilder("hello java!");
        int pos =0;
        try {
            for ( pos = 0; pos < 12; pos++) {
                switch (message.charAt(pos)) {
                    case 'a':
                    case 'e':
                    case 'o':
                     String uc=Character.toString(message.charAt(pos)).toUpperCase();
                     message.replace(pos, pos+1, uc);
                    }
            } catch (Exception e) {
                System.out.println("Out of limits");
            }
            System.out.println(message);
    }
}
```

What is the result?

A. hEllOjAvA!
B. Hello java!
C. Out of limits hEllOjAvA!
D. Out of limits

Answer: C

NEW QUESTION 160
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.

Answer: D

Explanation: Because the Java programming language does not require methods to catch or to specify unchecked exceptions (RuntimeException,Error, and their subclasses), programmers may be tempted to write code that throws only unchecked exceptions or to make all their exceptionsubclasses inherit fromRuntimeException. Both of these shortcuts allow programmers to write code without bothering with compiler errors and without bothering to specify or to catch anyexceptions. Although this may seem convenient to the programmer, it sidesteps theintent of thecatchorspecifyrequirement and can cause problems for others using your classes.

NEW QUESTION 162
Given the code fragment:
System.out.printIn ("Result: " +3+5); System.out.printIn ("Result: " + (3+5));
What is the result?

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

Answer: B

Explanation: In the first statement 3 and 5 are treated as strings andare simply concatenated. In the first statement 3 and 5 are treated as integers and their sum is calculated.

NEW QUESTION 167
Given:
public class String1 {
public static void main(String[] args) { String s = "123";
if (s.length() >2)

A. s.concat("456");for(int x = 0; x <3; x++) s += "x";System.out.println(s);}}What is the result?
B. 123
C. 123xxx
D. 123456
E. 123456xxx
F. Compilation fails

Answer: B

Explanation: 123xxx
The if clause is not applied. Note: Syntax of if-statement:
if (Statement) {
}

NEW QUESTION 172
Given the code fragment:

```
public class Test {
    public static List data = new ArrayList();

    // insert code here
    {
        for (String x : strs) {
            data.add(x);
        }
        return data;
    }

    public static void main(String[] args) {
        String[] d = {"a", "b", "c"};
        update(d);
        for (String s : d) {
            System.out.print(s + "  ");
        }
    }
}
```

Which code fragment, when inserted at // insert code here, enables the code to compile and and print a b c?

A. List update (String[] strs)
B. Static ArrayListupdate(String [] strs)
C. Static List update (String [] strs)
D. Static void update (String[] strs)
E. ArrayList static update(String [] strs)

Answer: E

NEW QUESTION 177
Given:

```
public class Test3 {
    public static void main(String[] args) {
        String names[] = new String[3];
        names[0] = "Mary Brown";
        names[1] = "Nancy Red";
        names[2] = "Jessy Orange";
        try {
            for(String n: names) {
                try {
                    String pwd = n.substring(0, 3)+n.substring(6,10);
                    System.out.println(pwd);
                }
                catch(StringIndexOutOfBoundsException sie) {
                    System.out.println("string out of limits");
                }
            }
        }
        catch(ArrayIndexOutOfBoundsException e) {
            System.out.println("array out of limits");
        }
    }
}
```

What is the result?

A. Marrown String out of limits JesOran
B. Marrown String out of limits Array out of limits
C. Marrown String out of limits
D. Marrown NanRed JesOran

Answer: A

NEW QUESTION 181
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"); }
}
What is the output?

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

Answer: C

Explanation: The following line causes a runtime exception (as the index is out of bounds): ages[4] = 17;
A runtime exception is thrown as anArrayIndexOutOfBoundsException.
Note: The third kind of exception (compared to checked exceptions and errors) is the runtime exception. These are exceptional conditions that are internal to the application, and
that the application usually cannot anticipate or recover from. These usually indicate programming bugs, such as logic errorsor improper use of an API.
Runtime exceptionsare not subjectto the Catch or Specify Requirement. Runtime exceptions are those indicated byRuntimeExceptionand its subclasses.

NEW QUESTION 184
Given:

Given:

```
class Dog {
  Dog() {
    try {
      throw new Exception();
    } catch (Exception e) { }
  }
}

class Test {
  public static void main(String[] args ) {
    Dog d1 = new Dog();
    Dog d2 = new Dog();
    Dog d3 = d2;
    // do complex stuff
  }
}
```

How many objects have been created when the line / / do complex stuff is reached?

A. Two
B. Three
C. Four
D. Six

Answer: C

NEW QUESTION 187
Given:

```
public class MyFor1 {
    public static void main(String[] args) {
        int[] x = {6, 7, 8};
        for (int i : x) {
            System.out.print(i + " ");
            i++;
        }
    }
}
```

What is the result?

A. 6 7 8
B. 7 8 9
C. 0 1 2
D. 6 8 10
E. Compilation fails

Answer: A

NEW QUESTION 189
Given the code fragment?
public class Test {
public static void main(String[]args) { Test t = new Test();
int[] arr = new int[10]; arr = t.subArray(arr,0,2);
}
// insert code here
}
Which method can be inserted at line // insert code here to enable the code to compile?

A. public int[] subArray(int[] src, int start, int end) { return src;}
B. public int subArray(int src, int start, int end) { return src;}
C. public int[] subArray(int src, int start, int end) { return src;}
D. public int subArray(int[] src, int start, int end) { return src;}

Answer: A

NEW QUESTION 194
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 replacekkin line x to cause jj = 5 to be output?

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

Answer: E

Explanation: We need to get jj to 5. It is initially set to 0. So we need to go through the for loop 5 times. The for loops ends when ii > 6 and ii decreases for every loop. So we need to initially set ii to 11. We set kk to 11.

NEW QUESTION 197
Which two may precede the word ‘class’ in a class declaration?

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

Answer: BC

Explanation: B: A class can be declared as public or private.
C: You can declare two kinds of classes: top-level classes and inner classes.
You define an inner class within a top-level class.Depending on how it is defined, an inner
class can be one of the following four types:Anonymous, Local, Member and Nested top- level.
A nested top-level class is a member classes with astaticmodifier. A nested top-level class is just like any other top-level class except that it is declared within another class or interface. Nested top-level classes are typically used as a convenient way to group related classes without creating a new package.
The following is an example:
public class Main { staticclass Killer {

NEW QUESTION 198
Given:

```
class Patient {
    String name;
    public Patient(String name) {
        this.name = name;
    }
}
```

And the code fragment:

```
 8. public class Test {
 9.     public static void main(String[] args) {
10.         List ps = new ArrayList();
11.         Patient p2 = new Patient("Mike");
12.         ps.add(p2);
13.
14.         // insert code here
15.
16.         if (f >=0 ) {
17.             System.out.print("Mike Found");
18.         }
19.     }
20. }
```

Which code fragment, when inserted at line 14, enables the code to print Mike Found?

A. int f = ps.indexOf {new patient (“Mike”)};
B. int f = ps.indexOf (patient(“Mike”));
C. patient p = new Patient (“Mike”); int f = pas.indexOf(P)
D. int f = ps.indexOf(p2);

Answer: C

NEW QUESTION 202
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. line4
C. line 5
D. line 6
E. line 10
F. line 11
G. line 12
H. line 13

Answer: CDH

NEW QUESTION 206
Given:

```
abstract class X {
    public abstract void methodX();
}
interface Y{
    public void methodY();
}
```

Which two code fragments are valid?

☐ A)
```
class Z extends X implements Y{
    public void methodZ(){}
}
```

☐ B)
```
abstract class Z extends X implements Y{
    public void methodZ(){}
}
```

☐ C)
```
class Z extends X implements Y{
    public void methodX(){}
}
```

☐ D)
```
abstract class Z extends X implements Y{
}
```

☐ E)
```
class Z extends X implements Y{
    public void methodY(){}
}
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E

Answer: BC

Explanation: When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class (C). However, if it does not, thenthe subclass must also be declared abstract (B).
Note: An abstract class is a class that is declared abstract—it may or may not include abstract methods. Abstract classes cannot be instantiated, but they can be subclassed.

NEW QUESTION 211
Given:

```
1. public abstract class Wow {
2.     private int wow;
3.     public Wow(int wow) {
4.         this.wow = wow;
5.     }
6.     public void wow() { }
7.     private void wowza() { }
8. }
```

What is true about the class Wow?

A. It compiles without error.
B. It does not compile because an abstract class cannot have private methods.
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.

Answer: A

NEW QUESTION 215
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

Answer: A

NEW QUESTION 216
Given:
public class Test { static boolean bVar;
public static void main(String[] args) { boolean bVar1 = true;
int count =8; do {
System.out.println("Hello Java! " +count); if (count >= 7) {
bVar1 = false;
}
} while (bVar != bVar1 && count > 4); count -= 2;
}
}
What isthe result?

A. Hello Java! 8 Hello Java! 6Hello Java! 4
B. Hello Java! 8 Hello Java! 6
C. Hello Java! 8
D. Compilation fails

Answer: C

Explanation: Hello Java! 8

NEW QUESTION 218
Given the code fragment:
System.out.printIn("Result: " + 2 + 3 + 5);
System.out.printIn("Result: " + 2 + 3 * 5); What is theresult?

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

Answer: C

Explanation: First line:
System.out.println("Result: " + 2 + 3 + 5); String concatenation is produced.
Second line:
System.out.println("Result: " + 2 + 3 * 5);
3*5 is calculated to 15 and is appended to string 2. Result 215.
The output is: Result: 235
Result: 215
Note #1:
To produce an arithmetic result, the following code would have to be used: System.out.println("Result: " + (2 + 3 + 5));
System.out.println("Result: " + (2 + 1 * 5)); run:
Result: 10
Result: 7
Note #2:
If the code was as follows:
System.out.println("Result: " + 2 + 3 + 5");
System.out.println("Result: " + 2 + 1 * 5");
The compilation would fail. There is an unclosed string literal, 5", on each line.

NEW QUESTION 222
Given the code fragment:
Int [] [] array = {{0}, {0, 1}, {0, 2, 4}, {0, 3, 6, 9}, {0, 4, 8, 12, 16}};
Systemout.printIn(array [4] [1]);
System.out.printIn (array) [1] [4]); What is the result?

A. 4Null

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

Answer: D

Explanation: The first println statement, System.out.println(array [4][1]);, works fine. It selects the element/array with index 4, {0, 4, 8, 12, 16},and from this array it selects the element with index 1, 4. Output: 4
The second println statement, System.out.println(array) [1][4]);, fails. It selects the array/element with index 1, {0, 1}, and from this array it try to select the element with index
4. This causes an exception.
Output: 4
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 4

NEW QUESTION 224
Given:
public class MyClass {
public static void main(String[] args) { String s = " Java Duke ";
int len = s.trim().length(); System.out.print(len);
}
}
What is the result?

A. 8
B. 9
C. 11
D. 10
E. Compilation fails

Answer: B

Explanation: Java -String trim() Method
This method returns a copy of the string, with leading and trailing whitespace omitted.

NEW QUESTION 225
Given the code fragment int var1 = -5;
int var2 = var1--;
int var3 = 0; if (var2 < 0) {
var3 = var2++;
} else {
var3 = --var2;
}
System.out.println(var3);
What is the result?

A. – 6
B. – 4
C. – 5
D. 5
E. 4
F. Compilation fails

Answer: C

NEW QUESTION 227
Given:
public class Test {
public static void main(String[] args) { int day = 1;
switch (day) {
case "7": System.out.print("Uranus");
case "6": System.out.print("Saturn");
case "1": System.out.print("Mercury");
case "2": System.out.print("Venus");
case "3": System.out.print("Earth");
case "4": System.out.print("Mars");
case "5": System.out.print("Jupiter");
}
}
}
Which two modifications, made independently, enable the code to compile and run?

A. Adding a break statement after each print statement
B. Adding a default section within the switch code-block
C. Changing the string literals in each case label to integer
D. Changing the type of the variable day to String
E. Arranging the case labels in ascending order

Answer: AC

Explanation: The following will work fine:
public class Test {
public static void main(String[] args) { int day = 1;
switch (day) {
case 7: System.out.print("Uranus"); break; case 6:System.out.print("Saturn"); break; case 1: System.out.print("Mercury"); break; case 2: System.out.print("Venus"); break; case 3: System.out.print("Earth"); break; case 4: System.out.print("Mars"); break; case 5: System.out.print("Jupiter"); break;
}
}
}

NEW QUESTION 230
A method doSomething () that has no exception handling code is modified to trail a method that throws achecked 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 aRunTimeException in the doSomething() method.
D. Catch the exception in the method that calls doSomething().

Answer: AB

Explanation: Valid Java programming language code must honor the Catch or Specify Requirement. This means that code that might throw certain exceptions must be enclosed by either of the following:
* A try statement that catches the exception. The try must provide a handler for the exception, as described in Catching and Handling Exceptions.
* A method that specifies that it can throw the exception. The method must provide a throws clause that lists the exception, as described in Specifying the Exceptions Thrown by a Method.
Code that fails to honor the Catch or Specify Requirement will not compile.

NEW QUESTION 235
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

Answer: C

Explanation: In this scenario the overloading method is called with a double/float value, 4.0. This makes the third overload method to run.
Note:
The Java programming language supportsoverloadingmethods, and Java can distinguish between methods with differentmethod signatures. This means that methods within a class can have the same name if they have different parameter lists. Overloaded methods are differentiated by the number and the type of the arguments passed into the method.

NEW QUESTION 240
Given:

```
public class App {
    public static void main(String[] args) {
        int i = 10;
        int j = 20;
        int k = j += i / 5;
        System.out.print(i + " : " + j + " : " + k);

    }
}
```

What is the result?

A. 10 : 22 : 20
B. 10 : 22 : 22
C. 10 : 22 : 6
D. 10 : 30 : 6

Answer: B

NEW QUESTION 243
Given the code fragment:
int a = 0; a++;
System.out.println(a++); System.out.println(a);
What is the result?

A. 12
B. 01
C. 11
D. 22

Answer: A

Explanation: The first println prints variable a with value 1 and then increases the variable to 2.

NEW QUESTION 244
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;

Answer: AD

Explanation: Any method declared void doesn't return a value. It does not need to contain a return statement, but it may do so. In such a case, a return statement can be used to branch out of a control flow block and exit the method and is simply used like this:
return;

NEW QUESTION 245
Given:

```
class X {
    int x1, x2, x3;
}
class Y extends X {
    int y1;
    Y() {
        x1 = 1;
        x2 = 2;
        y1 = 10;
    }
}

class Z extends Y {
    int z1;
    Z() {
        x1 = 3;
        y1 = 20;
        z1 = 100;
    }
}
```

And,

```
public class Test3 {
    public static void main(String[] args) {
        Z obj = new Z();
        System.out.println(obj.x3 + ", " + obj.y1 + ", " + obj.z1);
    }
}
```

Which constructor initializes the variable x3?

A. Only the default constructor of class X
B. Only the no-argument constructor of class Y
C. Only the no-argument constructor of class Z
D. Only the default constructor of object class

Answer: C

NEW QUESTION 248
Given the code fragment:
float x = 22.00f % 3.00f; int y = 22 % 3;
System.out.print(x + ", "+ y); What is the result?

A. 1.0, 1
B. 1.0f, 1
C. 7.33, 7
D. Compilation fails
E. An exception is thrown at runtime

Answer: A

NEW QUESTION 253
Which two actions will improve the encapsulation of a class?

A. Changing the access modifier of a field from public to private
B. Removing the public modifier from a class declaration
C. Changing the return type of a method to void
D. Returning a copy of the contents of an array or ArrayList instead of a direct reference

Answer: AD

Explanation: Reference: http://www.tutorialspoint.com/java/java_access_modifiers.htm

NEW QUESTION 254
Which three statements are true about the structure of a Java class?

A. A class can have only one private constructor.
B. A method can have the same name as a field.
C. A class can have overloaded static methods.
D. A public class must have a main method.
E. The methods are mandatory components of a class.
F. The fields need not be initialized before use.

Answer: ABC

Explanation: A: Private constructors prevent a class from being explicitly instantiatedby its callers.

If the programmer does not provide a constructor for a class, then the system will always provide a default, public no-argument constructor. To disable this default constructor, simply add a private no-argument constructor to the class. This private constructor may be empty.
B: The following works fine: int cake() {
int cake=0; return (1);
}
C: We can overload static method in Java. In terms of method overloading static method are just like normal methods and in order to overload static method you need to provide another static method with same name but different method signature.
Incorrect:
Not D: Only a public class in an application need to have a main method. Not E:
Example:
class A
{
public string something; public int a;
}
Q: What do you call classes without methods? Most of the time: An anti pattern.
Why? Because it faciliates procedural programming with "Operator" classes and data structures. You separate data and behaviour which isn't exactly good OOP.
Often times: A DTO (Data Transfer Object)
Read only datastructures meant to exchange data, derived from a business/domain object. Sometimes: Just data structure.
Well sometimes, you just gotta have those structures to hold data that is just plain andsimple and has no operations on it.
Not F: Fields need to be initialtized. If not the code will not compile. Example:
Uncompilable source code - variable x might not have been initialized

NEW QUESTION 257
Given:

```
public class Test {

    static void dispResult(int[] num) {
        try {
            System.out.println(num[1] / (num[1] - num[2]));
        } catch(ArithmeticException e) {
            System.err.println("first exception");
        }
        System.out.println("Done");
    }

    public static void main(String[] args) {
        try {
            int[] arr = {100, 100};
            dispResult(arr);
        } catch(IllegalArgumentException e) {
            System.err.println("second exception");
        } catch(Exception e) {
            System.err.println("third exception");
        }
    }
}
```

What is the result?

A. 0Done
B. First Exception Done
C. Second Exception
D. DoneThird Exception
E. Third Exception

Answer: B

NEW QUESTION 262
Given the following code:

```
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. }
```

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: Float price = new simple ();

D. Change line 5 to the following: Price = 4f;
E. Change line 5 to the following: price.price = 4;
F. Change line 5to 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

Answer: E

Explanation: price.price =4; is correct, not price=4;
The attribute price of the instance must be set, not the instance itself.

NEW QUESTION 264
Given the classes:
* AssertionError
* ArithmeticException
* ArrayIndexOutofBoundsException
* FileNotFoundException
* IllegalArgumentException
* IOError
* IOException
* NumberFormatException
* SQLException
Which option lists only those classes that belong to the unchecked exception category?
A. AssertionError, ArrayIndexOutOfBoundsException, ArithmeticException
B. AssertionError, IOError, IOException
C. ArithmeticException, FileNotFoundException, NumberFormatException
D. FileNotFoundException, IOException, SQLException
E. ArrayIndexOutOfBoundException, IllegalArgumentException, FileNotFoundException

Answer: A

Explanation: Not B: IOError and IOException are both checked errors. Not C, not D, not E: FileNotFoundException is a checked error.
Note:
Checked exceptions:
* represent invalid conditions in areas outside the immediate control of the program (invalid user input, database problems, network outages, absent files)
* are subclasses of Exception
* a method is obliged to establish a policy for all checked exceptions thrown by its implementation (either pass the checked exception further up the stack, or handle it somehow)
Note:
Unchecked exceptions:
* represent defects in the program (bugs) - often invalid arguments passed to a non-private method. To quote from The Java Programming Language, by Gosling, Arnold, and Holmes: "Unchecked runtime exceptions represent conditions that, generally speaking,reflect errors in your program's logic and cannot be reasonably recovered from at run time."
* are subclasses of RuntimeException, and are usually implemented using IllegalArgumentException, NullPointerException, or IllegalStateException
* method is not obliged to establish a policy for the unchecked exceptions thrown by its implementation (and they almost always do not do so)

NEW QUESTION 269
Given the code fragment:
Boolean b1 = true; Boolean b2 = false; int i = 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")

Answer: D

Explanation: Equals works fine on strings equals produces a Boolean value.

NEW QUESTION 273
Given:
```
package p1;
public interface DoInterface { void method1(int n1); // line n1
}
package p3;
import p1.DoInterface;
public class DoClass implements DoInterface { public DoClass(int p1) { }
public void method1(int p1) { } // line n2 private void method2(int p1) { } // line n3
}
public class Test {
public static void main(String[] args) { DoInterface doi= new DoClass(100); // line n4 doi.method1(100);
doi.method2(100);
}
}
```

Which change will enable the code to compile?

A. Adding the public modifier to the declarationof method1 at line n1
B. Removing the public modifier from the definition of method1 at line n2
C. Changing the private modifier on the declaration of method 2 public at line n3
D. Changing the line n4 DoClass doi = new DoClass ( );

Answer: C

Explanation: Private members (both fields and methods) are only accessible inside the class they are declared or inside inner classes. private keyword is one of four access modifier provided by Java and its a most restrictive among all four e.g. public, default(package), protected and private.
Read more: http://javarevisited.blogspot.com/2012/03/private-in-java-why-should-you- always.html#ixzz3Sh3mOc4D

NEW QUESTION 277
Given the code fragment:
class Student {
int rollnumber; String name;
List cources = new ArrayList();
// insert code here public String toString() {
return rollnumber + " : " + name + " : " + cources;
}
}
And,
public class Test {
public static void main(String[] args) { List cs = newArrayList(); cs.add("Java");
cs.add("C");
Student s = new Student(123,"Fred", cs); System.out.println(s);
}
}
Which code fragment, when inserted at line // insert code here, enables class Test to print 123 : Fred : [Java, C]?

A. private Student(int i, String name, List cs) {/* initialization code goes here */}
B. public void Student(int i, String name, List cs) {/* initialization code goes here */}
C. Student(int i, String name, List cs) {/* initialization code goes here */}
D. Student(int i, String name, ArrayList cs) {/* initialization code goes here */}

Answer: C

Explanation: Incorrect:
Not A: Student has private access line: Student s = new Student(123,"Fred", cs);
Not D: Cannot be applied to given types.Line: Student s = new Student(123,"Fred", cs);

NEW QUESTION 278
Which statement will empty the contents of aStringBuilder variable named sb?

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

Answer: C

NEW QUESTION 282
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

Answer: CE

NEW QUESTION 286
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

Answer: D

NEW QUESTION 287
Given the code fragment:

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

What is the result?

A. 22
B. 12
C. 32
D. 33

Answer: A

Explanation: Variable b is set to 4.
Variableb is decreased to 3.
Variable b is decreased to 2 and then printed. Output: 2 Variable b is printed. Output: 2

NEW QUESTION 290

Given:
public class Equal {
public static void main(String[] args) { String str1 = "Java";
String[] str2 = {"J","a","v","a"}; String str3 = "";
for (String str : str2) { str3 = str3+str;
}
boolean b1 = (str1 == str3); boolean b2 = (str1.equals(str3)); System.out.print(b1+", "+b2);
}
What is the result?

A. true, false
B. false, true
C. true, true
D. false, false

Answer: B

Explanation: == strict equality. equals compare state, not identity.

NEW QUESTION 291
Given the class definitions:

```
class Alpha {
    public String doStuff(String msg) {
        return msg;
    }
}
class Beta extends Alpha {
    public String doStuff(String msg) {
        return msg.replace('a', 'e');
    }
}
class Gamma extends Beta {
    public String doStuff(String msg) {
        return msg.substring(2);
    }
}
```

And the code fragment of the main() method,

```
12. List<Alpha> strs = new ArrayList<Alpha>();
13. strs.add(new Alpha());
14. strs.add(new Beta());
15. strs.add(new Gamma());
16. for (Alpha t : strs) {
17.     System.out.println(t.doStuff("Java"));
18. }
```

What is the result?

A. Java Java Java
B. Java Jeve va
C. Java Jeve ve
D. Compilation fails

Answer: D

NEW QUESTION 294
Given the code fragment:
Stringh1 = "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))

Answer: B

Explanation: The equals method compares values for equality.

NEW QUESTION 298
Given the code fragment: List colors = new ArrayList(); colors.add("green"); colors.add("red");
colors.add("blue"); colors.add("yellow"); colors.remove(2); colors.add(3,"cyan"); System.out.print(colors);
What is the result?

A. [green, red, yellow, cyan]
B. [green, blue, yellow, cyan]
C. [green, red, cyan, yellow]
D. Am IndexOutOfBoundsException is thrown at runtime

Answer: A

Explanation: First the list [green, red, blue, yellow] is build. The blue element is removed:
[green, red, yellow]
Finally the element cyan is added at then end of the list (index 3). [green, red, yellow, cyan]

NEW QUESTION 302
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;
}
}
Given
public 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");

Answer: D

NEW QUESTION 306
Given:
public class MyClass {
public static void main(String[] args) {
while (int ii = 0; ii < 2) {
ii++;
System.out.println("ii = " + ii);
}
}
}
What is the result?

A. ii = 1 ii = 2
B. Compilation fails
C. The program prints nothing
D. The program goes into an infinite loop with no output
E. The program goes to an infinite loop outputting: ii =1ii = 1

Answer: B

Explanation: The while statement is incorrect. It has the syntax of a for statement.
The while statement continually executes a block of statements while a particular condition is true. Its syntax can be expressed as:
while (expression) { statement(s)
}
The while statement evaluates expression, which must return a boolean value. If the expression evaluates to true, the while statement executes the statement(s) in the while block. The while statement continues testing the expression and executing its block until the expression evaluates to false.
Reference: The while and do-while Statements

NEW QUESTION 307
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 xand 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. 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

Answer: D

NEW QUESTION 309
Given:

```
public class X implements Z {
    public String toString() {
        return "X ";
    }
    public static void main(String[] args) {
        Y myY = new Y();
        X myX = myY;
        Z myZ = myX;
       System.out.print(myX);
       System.out.print((Y)myX);
       System.out.print(myZ);
    }
}

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

A. X XX
B. X Y X
C. Y Y X
D. Y YY

Answer: D

NEW QUESTION 312
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

Answer: C

Explanation: In segment 'if (valid)' valid must be of type boolean, but it is a string. This makes the compilation fail.

NEW QUESTION 313
.....

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 1Z0-803 Exam with Our Prep Materials Via below:

https://www.certleader.com/1Z0-803-dumps.html