Q1)

**package** com.lti.q1;

**public** **class** A {

**private** **boolean** checkValue (**int** val) {

**return** **true**;

  }

}

**public** **class** B **extends** A {

**public** **int** modifyVal (**int** val) {

**if** (checkValue (val)) {

**return** val;

} **else** {

**return** 0;

  }

}

**public** **static** **void** Main(String[] args) {

  B b = **new** B();

  System.***out***.println(b.modifyVal (10));

  }

}

What is the result?  
A. nothing  
B. It fails to compile. -(correct answer)  
C. 0  
D. A java.lang.IllegalArgumentException is thrown.  
E. 1

Q2)public interface API {   //line 1  
   public void checkValue(Object value)  
     throws IllegalArgumentException; //line 2   
public boolean isValueANumber (Object val) {  
  if (val instanceof Number) {  
       return true;  
}else  {  
   try {  
      Double.parseDouble (val.toString());  
   return true;  
   } catch (NumberFormatException ex) {  
           return false;  
  }  
       }    
     }  
}

 Which two changes need to be made to make this class compile? (Choose two.)  
A. Change Line 1 to an abstract class:public abstract class API {   -(correct answer)  
B. Change Line 2 access modifier to protected:protected void checkValue(Object value)throws IllegalArgumentException;  
C. Change Line 1 to a class:public class API {         
D. Change Line 1 to extend java.lang.AutoCloseable:public interface API extends AutoCloseable {  
E. Change Line 2 to an abstract method:public abstract void checkValue(Object value)throws IllegalArgumentException;  -(correct answer)

Q3.Which two modules include APIs in the Java SE Specification? (Choose two.)  
A. java.logging  -(correct answer)  
B. java.desktop  -(correct answer)  
C. javafx  
D. jdk.httpserver   
E. jdk.jartoo

Q4.

**package** com.lti.q4;

**public** **class** Test{

**private** **int** num = 1;

**private** **int** div = 0;

**public** **void** divide () {

**try** {

   num = num / div;

   System.***out***.print ("Exception");

}

**catch** (ArithmeticException ae) {num = 100; }

**catch** (Exception e) { num = 200; }

**finally** { num = 300; }

System.***out***.print (num);

}

**public** **static** **void** main(String args[])

{

Test test = **new** Test();

test.divide ();

}

}

What is the output?

A. 300

B. Exception

C. 200

D. 100

Q5)Which two statements are true about the modular JDK? (Choose two.)   
A. The foundational APIs of the Java SE Platform are found in the java.base module.   
B. An application must be structured as modules in order to run on the modular JDK.   
C. It is possible but undesirable to configure modules’ exports from the command line.   
D. APIs are deprecated more aggressively because the JDK has been modularized.

Q6.

**package** com.lti.q6;

**import** java.util.Arrays;

**public** **class** Test1 {

**public** **static** **void** main(String[] args) {

**int**[] secA = { 2, 4, 6, 8, 10 };

**int**[] secB = { 2, 4, 8, 6, 10 };

**int** res1 = Arrays.*mismatch* (secA, secB);

**int** res2 = Arrays.*compare* (secA, secB);

    System.***out***.print (res1 + ": " + res2);

  }

}

What is the result?

A. -1 : 2

B. 2 : -1s

C. 2 : 3

D. 3 : 0

Q7.

**package** com.lti.ques1;

**import** java.io.\*;

**public** **class** Tester {

**public** **static** **void** main(String[] args) {

**try** {

*doA*();

*doB*();

    } **catch** (IOException e) {

      System.***out***.println("c");

**return**;

    } **finally** {

      System.***out***.println("d");

    }

    System.***out***.println("f");

  }

**private** **static** **void** doA() {

    System.***out***.println("a");

**if** (**false**) {

**throw** **new** IndexOutOfBoundsException();

    }

  }

**private** **static** **void** doB() **throws** FileNotFoundException {

    System.***out***.println("b");

**if** (**true**) {

**throw** **new** FileNotFoundException();

    }

  }

}

What is the result?  
A. The compilation fails.  
B. abdf  
C. abd  
D. adf  
E. abcd

Q8. Which set of commands is necessary to create and run a custom runtime image from Java source files?

1. java, jdeps  
   B. javac, jlink  
   C. jar, jlink  
   D. javac, jar

Q9.

**public** **class** Tester {

**public** **static** **void** main(String[] args) {

    StringBuilder sb = **new** StringBuilder(5);

    sb.append("HOWDY");

    sb.insert(0, ' ');

    sb.replace(3, 5, "LL");

    sb.insert(6, "COW");

    sb.delete(2, 7);

    System.***out***.println(sb.length());

}                                                                                                                                                                               What is the result?  
A. 4   
B. 3   
C. An exception is thrown at runtime.   
D. 5

Q10.

**package** Dumps1\_question10;

**import** java.util.function.BiFunction;

**public** **class** Pair<T> {

**final** BiFunction<T, T, Boolean> validator;

  T left = **null**;

T right = **null**;

**private** Pair() {

    validator = **null**;

  }

Pair(BiFunction<T, T, Boolean> v, T x, T y) {

    validator = v;

    set(x,y);

  }

**void** set(T x, T y)

  {

**if**(!validator.apply(x, y)) **throw** **new** IllegalArgumentException();

    setLeft(x);

    setRight(y);

  }

**void** setLeft(T x) {

    left = x;

  }

**void** setRight(T y) {

    right = y;

  }

**final** **boolean** isValid() {

**return** validator.apply(left , right);

  }

}

It is required that if p instanceof Pair then p.isValid()  returns true. Which is the smallest set of visibility changes to insure this requirement is met?

Options :

A) setLeft and set Right must be protected

B) **left and right must be private**

C) isValid must be public

D) left,right,setLeft,and setRight must be private

**Q11. package** Dumps1\_question11;

**public** **class** Test {

**public** **static** **void** main(String[] args) {

**var** i = 10;

**var** j = 5;

    i += (j\*5 + j) / i - 2;

    System.***out***.println(i);

  }

}

**Q12.package** Dumps1\_question12;

**public** **class** Tester {

**private** **int** x;

**private** **static** **int** *y*;

**public** **static** **void** main(String[] args) {

    Tester t1 = **new** Tester();

    t1.x = 2;

    Tester.*y* = 3;

    Tester t2 = **new** Tester();

    t2.x = 4;

    t2.*y* = 5;

    System.***out***.print(t1.x+","+t1.*y*);

    System.***out***.print(t2.x+","+Tester.*y*);

    System.***out***.print(t2.x+","+t1.*y*);

  }

}

Options :

A) 2,34,34,5

B) 2,34,54,5

C) **2,54,54,5**

D) 2,34,54,3

Q13

 public interface EulerInterface {

  double getEulerValue();  
}

public class EulerLamda {

public static void main(String[] args) {  
    EulerInterface myEulerInterface;  
    myEulerInterface = () -> "2.71828";  
    System.out.println("Value of Euler = " + myEulerInterface.getEulerValue());  
  }  
}

 What is the result?     
A. It throws a runtime exception.  
B. Value of Euler = 2.71828   
C. The code does not compile.  
D. Value of Euler = “2.71828”

Q14

package com.lti.lamda;

 public class Myclass {

public static void main(String[] arg) {  
    System.out.println(arg[1] + "--" + arg[3] + "--" + arg[0]);  
  }  
}

 executed using this command: java Myclass My Car is red

 What is the output of this class?   
A. Car--red--My  
B. My--Car--is  
C. My--is--java  
D. java--Myclass--My  
E. Myclass--Car—red

Q15.

package b;

public class Person {  
  
 protected Person() { //line 1  
  
  }  
}

and

package a;

import b.Person;

public class Main { //line 2

 public static void main(String[] args) {  
    Person person = new Person(); // line 3  
  }  
}

Which two allow a.Main to allocate a new Person? (Choose two.)

A. In Line 1, change the access modifier to privateprivate Person() {

B. In Line 1, change the access modifier to publicpublic Person() {

C. In Line 2, add extends Person to the Main classpublic class Main extends Person {and change Line  
3 to create a new Main objectPerson person = new Main();

D. In Line 2, change the access modifier to protectedprotected class Main {

E. In Line 1, remove the access modifierPerson() {

Q16.

{  
  Iterator iter=List.of(1,2,3).iterator();  
  while(iter.hasNext()){  
    foo(iter.next());  
  }  
  Iterator iter2 = List.of(1, 2, 3).iterator();  
  while(iter.hasNext()){  
    bar(iter2.next());  
  }  
}  
  for(Iterator iter = List.of(1, 2, 3).iterator();iter.hasNext();)  
  {  
    foo(iter.next());  
  }  
  for(Iterator iter2 = List.of(1, 2, 3).iterator();iter.hasNext();)  
  {  
    bar(iter2.next());  
  }

Which loop incurs a compile time error?  
A. the loop starting line 11  
B. the loop starting line 7  
C. the loop starting line 14  
D. the loop starting line 3

Q17. Which two statements are true about Java modules? (Choose two.)  
A. Modular jars loaded from --module-path are automatic modules.  
B. Any named module can directly access all classes in an automatic module.  
C. Classes found in –classpath are part of an unnamed module.  
D. Modular jars loaded from –classpath are automatic modules.  
E. If a package is defined in both the named module and the unnamed module, then the package in   
the unnamed module is ignored.

Q18.

public class DNASynth{  
  int aCount;  
  int tCount;  
  int cCount;  
  int gCount;  
  DNASynth(int a, int tCount, int c, int g) {  
    //line1  
  }  
  int setCCount(int c) {  
    return c;  
  }  
  void setGCount (int gCount) {  
    this.gCount = gCount;  
  }  
}

A. setCCount(c) = cCount;  
B. tCount = tCount;  
C. setGCount(g);  
D. cCount = setCCount(c);  
E. aCount = a;

**Question:19**

**class** Mycar {

}

and

javac C:\workspace4\Mycar.java

What is the expected result of javac?

A. javac fails to compile the class and prints the error message, C:\workspace4\Mycar.java:1:error: package java does not exist

B. javac compiles Mycar.java without errors or warnings.

C. javac fails to compile the class and prints the error message, C:\workspace4\Mycar.java:1:error: expected import java.lang

D. javac fails to compile the class and prints the error message, Error: Could not find or load main class Mycar.class

**Question:20**

**1.public** **interface** Pastry {

2.          **void** getIngredients();

3.}

**4.abstract** **class** Cookie **implements** Pastry {}

**5.**

**6.class** ChocolateCookie **implements** Cookie {

7.          **publicvoid** getIngredients() {}

8.}

**9.class** Coconut ChocolateCookie **extends** ChocolateCookie {

10.         **void** getIngredients (**int** x) {}

11.}

Which is true?

A. The *compilation* fails due to an error in line 6.

B. The compilation succeeds.

C. The compilation fails due to an error in line 4.

D. The compilation fails due to an error in line 10.

E. The compilation fails due to an error in line 7.

F. The compilation fails due to an error in line 9.

G. The compilation fails due to an error in line 2.

**Question :21**

StringBuilder s = **new** StringBuilder("ABCD");

Which would cause s to be AQCD?

A. s.replace(s.indexOf(“A”), s.indexOf(“C”), “Q”);

B. s.replace(s.indexOf(“B”), s.indexOf(“C”), “Q”);

C. s.replace(s.indexOf(“B”), s.indexOf(“B”), “Q”);

D. s.replace(s.indexOf(“A”), s.indexOf(“B”), “Q”);

**Q22**

**package** com.lti.classapp;

**import** java.util.ArrayList;

**class** Employee {

   String office;

}

 and the code fragment:

**5. public** **class** HRApp

6.**var** employee = **new** ArrayList<Employee>();

7.  **public** **void** display() {

8.   **var** employee = **new** Employee();

9.  **var** offices=**new** ArrayList<>();

10.      offices.add("Chicago");

11.      offices.add("Bangalore");

 12.     **for** (**var** office : offices) { // Explicitly declared type

 13.         System.***out***.println("Employee Location: " + office);

 14.    }

 15.    }

 16. }

 Which two lines cause compilation errors? (Choose two.)

A. line 12

B. line 6

C. line 9

D. line 8

E. line 7

Q23.Which describes a characteristic of setting up the Java development environment?

A. Setting up the Java development environment requires that you also install the JRE.

B. The Java development environment is set up for all operating systems by default.

C. You set up the Java development environment for a specific operating system when you install the JDK.

D. Setting up the Java development environment occurs when you install an IDE before the JDK.

**Question: 24**

Given:

**package** test.t1;

**public** **class** A {

**public** **int** x = 42;

**protected** A() {}          // line 1

}

and

**package** test.t2;

**import** test.t1.\*;

**public** **class** B **extends** A {

**int** x = 17;               // line 2

**public** B() { **super**(); }   // line 3

}

and

**package** test;

**import** test.t1.\*;

**import** test.t2.\*;

**public** **class** Tester {

**public** **staticvoid** main(String[] args) {

       A obj = **new** B();                         // line 4

       System.***out***.println(obj.x);              // line 5

   }

}

What is the result?

A. 42

B. The compilation fails due to an error in line 4.

C. 17

D. The compilation fails due to an error in line 3.

E. The compilation fails due to an error in line 2.

F. The compilation fails due to an error in line 1.

G. The compilation fails due to an error in line 5.

Q25)Given:

public class Foo {

             public <T> Collection<T> foo(Collection<T> arg) { ... }

}

and

public class Bar extends Foo { ... }

Which two statements are true if the method is added to Bar? (Choose two.)

A. public Collection<String> foo(Collection<String> arg) { ... } overrides Foo.foo.

B. public <T> Collection<T> foo(Stream<T> arg) { ... } overloads Foo.foo.

C. public <T> List<T> foo(Collection<T> arg) { ... } overrides Foo.foo.

D. public <T> Collection<T> foo(Collection<T> arg) { ... } overloads Foo.foo.

E. public <T> Collection<T> bar(Collection<T> arg) { ... } overloads Foo.foo.

F. public <T> Iterable<T> foo(Collection<T> arg) { ... } overrides Foo.foo.

26)What is the result?

A. ab cd ef

B. An ArrayIndexOutOfBoundsException is thrown at runtime.

C. The compilation fails.

D. abc def

E. ad be cf

Q27)executed using command:

java Hello “Hello World” Hello World

What is the output2

A. An exception is thrown at runtime.

B. Hello WorldHello World

C. Hello World Hello World

D. Hello WorldHelloWorld

E. HelloHello WorldHelloWorld

Question 28:

Given

public class Test {

private String[] strings;

}

Which two constructors will compile and set the class field strings? (Choose two.)

A.

public Test (List<String> strings) {

this.strings = strings;

}

B.

public Test (String... strings) {

strings = strings;

}

C.

public Test (String... strings) {

this.strings = strings;

}

D.

public Test (String strings) {

strings = strings;

}

E.

public Test (String[] strings) {

this.strings = strings;

}

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

Question 29:

Given the code fragment:

String s1 = new String("ORACLE");

String s2 = "ORACLE";

String s3 = s1.intern();

System.out.print ((s1==s2) +" ");

System.out.print((s2==s3) + " ");

System.out.println(s1==s3);

What is the result?

A. false true true

B. true false false

C. false false true

D. false true false

Qu30:

import java.util.ArrayList;

import java.util.Arrays;

public class NewMain {

public static void main(String[] args) {

String[] fruitNames = { "apple", "orange",

"grape", "lemon", "apricot", "watermelon" };

var fruits = new ArrayList<> (Arrays.asList (fruitNames));

fruits.sort((var a, var b) -> -a.compareTo (b));

fruits.forEach (System.out::println);

}

}

What is the result?

A. watermelonorangelemongrapeapricotapple

B. nothing

C. appleapricotgrapelemonorangewatermelon

D. appleorangegrapelemonapricotwatermelon

Question 29:

Given the code fragment:

String s1 = new String("ORACLE");

String s2 = "ORACLE";

String s3 = s1.intern();

System.out.print ((s1==s2) +" ");

System.out.print((s2==s3) + " ");

System.out.println(s1==s3);

What is the result?

A. false true true

B. true false false

C. false false true

D. false true false

**Question:31**

Given the code fragment:

**int x = 0;**

**while(x<10) {**

**System.out.print(x++);**

**}**

Which “for” loop produces the same output?

1. **int b=0;**

**For( ;b<10; ){**

**System.out.print(++b);**

**}**

1. **for(a;a<10;a++){**

**System.out.print(a);**

**}**

1. **for(int d = 0; d<10; ){**

**System.out.print(d);**

**++d;**

**}**

1. **for(int c = 0; ; c++){**

**System.out.print(c);**

**If(c==10){**

**Break;**

**}**

**}**

**A. Option A**

**B. Option B**

**C. Option C**

**D. Option D**

**Question:32**

**public interface InterfaceOne{**

**void printOne();**

**}**

**Which three classes successfully override printOne()? (Choose three.)**

1. **public abstract class TestClass implements InterfaceOne{**

**public abstract void printOne();**

**}**

1. **public class TestClass implements InterfaceOne{**

**private void printOne(){**

**System.out.println("one");**

**}**

**}**

1. **public class TestClass implements InterfaceOne{**

**public void printOne(){**

**System.out.println("one");**

**}**

**}**

1. **public abstract class TestClass implements InterfaceOne{**

**public void printOne(){**

**System.out.println("one");**

**}**

**}**

1. **public abstract class TestClass implements InterfaceOne{**

**public String printOne(){**

**return "one";**

**}**

**}**

1. **public class TestClass {**

**public void printOne(){**

**System.out.println("one");**

**}**

**}**

**A. Option A**

**B. Option B**

**C. Option C**

**D. Option D**

**E. Option E**

**F. Option F**

**Question:33**

**public class Test {**

**public static void main(String[] args) {**

**AnotherClass ac = new AnotherClass();**

**SomeClass sc = new AnotherClass();**

**ac = sc;**

**sc.methodA ();**

**ac.methodA ();**

**}**

**}**

**class SomeClass {**

**public void methodA () {**

**System.out.println("SomeClass #methodA ()");**

**}**

**}**

**class AnotherClass extends SomeClass {**

**public void methodA () {**

**System.out.println("AnotherClass #methodA ()");**

**}**

**}**

**What is the result?**

**A. A ClassCastException is thrown at runtime.**

**B. AnotherClass#methodA()AnotherClass#methodA()**

**C. The compilation fails.**

**D. SomeClass#methodA()AnotherClass#methodA()**

**E. AnotherClass#methodA()SomeClass#methodA()**

**F. SomeClass#methodA()SomeClass#methodA()**

**Question no: 34**

Which two statements are correct about try blocks? (Choose two.)

A. A try block can have more than one catch block.

B. A finally block in a try-with-resources statement executes before the resources declared are closed.

C. A finally block must be immediately placed after the try or catch blocks.

D. A try block must have a catch block and a finally block.

E. catch blocks must be ordered from generic to specific exception types.

**Question no: 35**

public class Over {

public void analyze(Object[] o) {

System.***out***.println("I am an object array");

}

public void analyze (long[] l) {

System.***out***.println("I am an array");

}

public void analyze(Object o) {

System.***out***.println("I am an object");

}

public static void main(String[] args) {

int[] nums = new int[10];

new Over().analyze(nums); // line 1

}

}

What is the output?

A. I am an object array

B. The compilation fails due to an error in line 1.

C. I am an array

D. I am an object

**Question no: 36**

public class Price {

private final double value;

public Price(String value) {

this(Double.parseDouble(value));

}

public Price(double value) {

this.value = value;

}

public Price() {

}

public double getValue() {

return value;

}

public static void main(String[] args) {

Price pl = new Price("1.99");

Price p2 = new Price(2.99);

Price p3 = new Price();

System.out.println(pl.getValue() + ", " + p2.getValue() + ", " +p3.getValue());

}

}

**What is the result?**

A. The compilation fails.

B. 1.99,2.99,0

C. 1.99,2.99,0.0

D. 1.99,2.99

**Question 37:**

class ConSuper{

protected ConSuper() {

this(2);

System.***out***.println("1");

}

protected ConSuper(int a) {

System.***out***.println(a);

}

}

public class ConSub extends ConSuper{

ConSub(){

this(4);

System.***out***.println("3");

}

ConSub(int a){

System.***out***.println(a);

}

public static void main(String[] args) {

new ConSub(4);

}

}

What is the result?

A. 2134

B. 2143

C. 214

D. 234

**Question 38:**

public class Tester {

public static void main(String[] args) {

String s = "this is it";

int x = s.indexOf("is");

s.substring(x+3);

x = s.indexOf("is");

System.***out***.println(s+" "+x);

}

}

What is the result?

A. is it 1

B. An IndexOutOfBoundsException is thrown at runtime.

C. is it 0

D. this is it 2

E. this is it 3

**Question 39:**

Which two commands are used to identify class and module dependencies? (Choose two.)

A. jmod describe

B. java Hello.java

C. jdeps --list-deps

D. jar --show-module-resolution

E. java --show-module-resolution

**Question 40:**

public class Main {

  public static void main(String[] args) {

              String[][] arr = { { "Red", "White" }, { "Black" }, { "Blue", "Yellow", "Green", "Violet" } };

    for (int row = 0; row < arr.length; row++) {

      int column = 0;

      for (; column < arr[row].length; column++) {

        System.out.println("[" + row + "," + column + "] = " + arr[row][column]);

      }

    }

    }

}

**What is the result?**

A. [0,0] = Red[0,1] = White[1,0] = Black[1,1] = Blue[2,0] = Yellow[2,1] = Green[3,0] = Violet

B. [0,0] = Red[1,0] = Black[2,0] = Blue

C. java.lang.ArrayIndexOutOfBoundsException thrown

D. [0,0] = Red[0,1] = White[1,0] = Black[2,0] = Blue[2,1] = Yellow[2,2] = Green[2,3] = Violet

**Question 41:**

import java.time.LocalDate;

import static java.time.DayOfWeek.\*;

public class Main {

  public static void main(String[] args) {

    var today = LocalDate.*now*().with(*TUESDAY*).getDayOfWeek();

    switch (today) {

    case *SUNDAY*:

    case *SATURDAY*:

      System.*out*.println("Weekend");

      break;

    case *MONDAY*:

      FRIDAY: System.*out*.println("Working");

    default:

      System.*out*.println("Unknown");

    }

  }

}

**What is the result?**

A. WorkingUnknown

B. Unknown

C. TuesdayUnknown

D. The compilation fails.

E. Tuesday

F. Working

**Question 42:**  
  
public interface A {

abstract void x();  
}

and

public abstract class B /\* position 1 \*/ {

/\* position 2 \*/  
 public void x() { }

public abstract void z();  
}

and

public class C extends B implements A {  
/\* position 3 \*/  
}

**Which code, when inserted at one or more marked positions, would allow classes B and C to compile?**

A. @Override // position 3void x () {} // position 3@Override // position 3public void z() { } // position 3

B. @Override // position 2public void z() { } // position 3

C. implements A // position 1@Override // position 2

D. public void z() { } // position 3

**Question 44:**

**Which two statements are correct about modules in Java? (Choose two.)**

A. java.base exports all of the Java platforms core packages.  
B. module-info.java can be placed in any folder inside module-path.  
C. A module must be declared in module-info.java file.  
D. module-info.java cannot be empty.  
E. By default, modules can access each other as long as they run in the same folder

**Question 45:**

**Which two describe reasons to modularize the JDK? (Choose two.)**

A. easier to understand the Java language  
B. improves security and maintainability  
C. easier to expose implementation details  
D. improves application robustness  
E. easier to build a custom runtime linking application modules and JDK modules

**Question 46:**

public class Test {

  public static void main(String[] args) {

    int x;

    int y = 5;

    if (y > 2) {

      x = ++y;

      y = x + 7;

    } else {

      y++;

    }

    System.***out***.print(x + " " + y);

  }

}

**Question 47:**

class DNASynth {

      int aCount;

      int tCount;

      int cCount;

      int gCount;

      void setACount (int cCount) {

        cCount = cCount;

      }

      void setTCount () {

        this.tCount = tCount;

      }

      int setCCount () {

        return cCount;

      }

      int setGCount (int g) {

      gCount = g;

        return gCount;

      }

      void setAllCounts (int x) {

        aCount = tCount = this.cCount = setGCount (x);

      }

}

Which two methods modify field values? (Choose two.)

**A. setAllCounts**

B. setACount

**C. setGCount**

D.setCCount

E. setTCount

**Question 48:**

public class Tester {

  public static void main(String[] args) {

    char letter = 'b';

    int i = 0;

    switch (letter) {

    case 'a':

      i++;

      break;

    case 'b':

      i++;

    case 'c' | 'd': // line 1

      i++;

    case 'e':

      i++;

      break;

    case 'f':

      i++;

      break;

    default:

      System.*out*.print(letter);

    }

    System.*out*.println(i);

  }

}

**What is the result?**  
A. b1  
B. 2  
C. b2  
D. 1  
E. b3  
**F. 3**  
G. The compilation fails due to an error in line 1

**Question 49**

public class Test {

  public static void main(String[] args) {

    // TODO Auto-generated method stub

    int x = 0;

    do {

    x++;

    if (x == 1) {

    continue;

    }

    System.*out*.println(x);

    } while (x < 1);

  }

}

What is the result?

 A. 01

 B. 0

C. 1

**D. The program prints nothing.**

E. It prints 1 in the infinite loop.

**Question 50:**

public class foo

{

  public static void main (String ... args)

   {

    for (var x : args) {

    System.*out*.println(x);

     }

   }

}

**What is the type of the local variable x?**

 A. Character

 B. char

 C. String[ ]

1. String

**Question 51:**

public class Test {

    static String *prefix* = "Global:";

    private String name = "namescope";

    public static String getName () {

     return new Test ().name;

    }

   public static void main (String[] args) {

     Test t = new Test ();

     System.*out*.println (/\* Insert code here \*/ Test.*prefix*+Test.*getName*()

);

    }

}

**Which two options can you insert inside println method to produce Global:namescope?** (Choose two.)

 A. Test.prefix+Test.name

 B. new Test().prefix+new Test().name

 C. Test.prefix+Test.getName()

**Question 52.**  
import java.util .\*;  
public class Foo {  
public List<Number> foo(Set<CharSequence> m) { ... }

}  
and

import java.util .\*;  
public class Bar extends Foo {  
//line 1  
}

**Which two statements can be added at line 1 in Bar to successfully compile it? (Choose two.)**

A. public List<Integer> foo(Set<CharSequence> m) { ... }   
B. public ArrayList<Number> foo(Set<CharSequence> m) { ... }   
C. public List<Integer> foo(TreeSet<String> m) { ... }   
D. public List<Integer> foo(Set<String> m) { ... }   
E. public List<Object> foo(Set<CharSequence> m) { ... }   
F. public ArrayList<Integer> foo(Set<String> m) { ... }

**Question 53:**

public class Foo {  
private void print () {  
System. out.println ("Bonjour le monde!");  
}

public void foo () {  
print ();  
}  
}

public class Bar extends Foo {  
private void print () {  
System.out.println ("Hello world!");  
}

public void bar () {  
print ();  
}  
public static void main (String ... args) {  
Bar b = new Bar ();  
b.foo();  
b.bar ();  
}  
}

**What is the output?**A. Hello world!Bonjour le monde!  
B. Hello world!Hello world!  
C. Bonjour le monde!Hello world!  
D. Bonjour le monde!Bonjour le monde!

**Question 54:**  
public method foo() throws FooException {

...  }

and omitting the throws FooException clause results in a compilation error.  
**Which statement is true about FooException?**  
A. FooException is a subclass of RuntimeError.  
B. FooException is unchecked.  
C. The body of foo can only throw FooException.  
D. The body of foo can throw FooException or one of its subclasses.

**Question: 55**

Which describes an aspect of Java that contributes to high performance?

A. Java prioritizes garbage collection.

B. Java has a library of built-in functions that can be used to enable pipeline burst execution.

C. Java monitors and optimizes code that is frequently executed.

D. Java automatically parallelizes code execution.

**Question: 56**

**Given:**

public class MethodTest {

// line 1

}

**Which two method implementations are correct, when inserted independently in line 1? (Choose two.)**

A.

public boolean methodD(int x) {

return x > 0;

}

B.

public boolean methodB() {

System.out.println("methodB");

}

C.

public char methodE(String msg) {

return msg;

}

D.

public void methodC(int x) {

return x++;

}

E.

public void methodA() {

System.out.println("methodA");

}

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

**Question: 57**

Given the formula to calculate a monthly mortgage payment:

A black text on a white background

AI-generated content may be incorrect.

and these declarations:

double m; //monthly payment

double r = 0.05/12; //monthly interest rate

int p = 100\_000; //principal

int n = 180; //number of payments

How can you code the formula?

A. m = p \* (r \* Math.pow(1 + r, n) / (Math.pow(1 + r, n) - 1));

B. m = p \* ((r \* Math.pow(1 + r, n) / (Math.pow(1 + r, n)) - 1));

C. m = p \* r \* Math.pow(1 + r, n) / Math.pow(1 + r, n) - 1;

D. m = p \* (r \* Math.pow(1 + r, n) / Math.pow(1 + r, n) - 1);