Question := 1

View the exhibit

```
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:

```
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

Question := 2

Given:

```
class ScopeTest  {
   int z;
   public static void main (String[] args)  {
```

```
        ScopeTest myScope = new ScopeTest();
         int z = 6;
         System.out.println (z);
         myScope.doStuff();
         System.out.println (z);
         System.out.println (myScope.z);
  }
  void doStuff()   {
          int z = 5;
          doStuff2();
          System.out.println (z);
  }
  void doStuff2()   {
         z = 4;
  }
}
```

What is the result?

A. 6 5 6 4
B. 4 5 5 6
C. 6 4 5 4
D. 6 0 7 4

Question := 3

Given the code fragment:

```
interface SampleClosable {

public void close () throws java.io.IOException;

}
```

Which three implementations are valid?

A)

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

B)

```
public class Test implements SampleClosable {
      public void close() throws Exception {
           // do something
```

```
        }
}
C)    public class Test implements SampleClosable {
    public void close() throws java.io.FileNotFoundException {
          // do something
    }
}
D)    public class Test extends SampleClosable {
         public void close() throws java.io.IOException {
               // do something
         }
}
E)    class Test implements SampleClosable {
    public void close(){
          // do something
    }
}
```

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

Question := 4

Given:

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

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

A. Zero

B. Once

C. Twice
D. Thrice
E. Compilation fails.

Question := 5
Given the code fragment:

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

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

What is the result?

A. Result: 10
Result: 30

B. Result: 10
Result: 25

C. Result: 235
Result: 215

D. Result: 215
Result: 215

E. Compilation fails

Question := 6
Given the code fragment:

```
    int a = 0;

    a++;

    System.out.print(a++);

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

What is the result?

A. 12
B. 22

C. 11
D. 12

Question := 7
Given:
String message1 = "Wham bam!";
String message2 =new String ("Wham bam!");

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

What is the result?

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

Question := 8
Given the code fragment:

String h1 = "Bob";

String h2 = new String ("Bob");

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

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

Question := 9
A method doSomething () that has no exception handling code is modified to trail a method that throws a checked excep
independently, will allow the program to compile?
A. Catch the exception in the method doSomething().
B. Declare the exception to be thrown in the doSomething() method signature.

C. Cast the exception to a RunTimeException in the doSomething() method.
D. Catch the exception in the method that calls doSomething().

Question := 10
Which two may precede the word `class' in a class declaration?

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

Question := 11
Given:

```
public static void main(String [ ] args)  {
        int a, b, c = 0;
        int a; b; c ;
        int g, int h, int i = 0;
        int d, e, f ;
        int k=0,l=0,m = 0;
}
```

Which three declarations will compile?

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

Question := 12
Which statement initializes a stringBuilder to a capacity of 128?

A. StringBuilder sb = new String ("128");
B. StringBuilder sb = StringBuilder.setCapacity (128);
C. StringBuilder sb = StringBuilder.getInstance (128);
D. StringBuilder sb = new StringBuilder (128);
E. none of above.

Question := 13
Given the following code fragment:

```
if    (value >= 0)  {
       if    (value  != 0)
             System.out.print (“the “);
       else
            System.out.print (“quick“);
       if    (value  < 10)
             System.out.print (“brown“);
       if    (value  > 30)
             System.out.print (“Fox “);
       else if    (value  < 50)
             System.out.print (“jumps “);
       else if    (value  < 10)
             System.out.print (“over “);
       if    (value  > 10)
             System.out.print (“lazy “);
} else {
      System.out.print (“dog“);
}
System.out.printIn (“… “);
```

What is the result if the integer value is 33?

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

Question := 14
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 the doSomething() method and the doSomethingElse() method in the try block of a try catch

Question := 15

Which declaration initializes a boolean variable?

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

Question := 16

Given:

```
class Circle  {
      double radius;
      public double area;
      public Circle(double r)  {radius = r;}
      public double getRadius() {return radius;  }
      public void setRadius (double r)  {radius = r;  }
      public double getArea ()  {return /* ??? */;  }
}
class App {
    public static void main (String[] args)  {
      Circle c1 = new Circle (17.4);
       c1.area = Math.PI*c1.getRadius() * c1.getRadius ();
    }
}
```

This class is poorly encapsulated. You need to change the circle class to compute and return the area instead.

What three modifications are necessary to ensure that the class is being properly encapsulated?

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

```
return radius;
}
```

Question := 17

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;

Question := 18

Which three are valid types for switch?
A. int
B. float
C. double
D. integer
E. String
F. Float

Question := 19

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

Question := 20

Given:

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

```
        }
}
class AnotherSampleClass extends SampleClass  {
}
```

What is the result?

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

Question := 21

Given:

```
abstract class Wow {
    private int wow;
    public Wow (int wow)  {
       this.wow = wow;
    }
    public void wow()  { }
    private void wowza()  { }
}
```

What is true about the class Wow?

A. It compiles without error.
B. It does not compile because an 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.

Question := 22

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.

Question := 23
View the Exhibit.

```
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

```
class TestHat {

public static void main(String[] args) {

Hat blackCowboyHat = new Hat();

}

}
```

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

Question := 24
Given:

```
class MyFor {
    public static void main(String[] args) {
        for (int ii = 0; ii < 4; ii++) {
```

```
            System.out.println("ii = "+ ii);
            ii = ii +1;
        }
    }
}
```

What is the result?

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

Question := 25

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
}
}

Question := 26

Given the code fragment:

```
String name = "Spot";

int age = 4;
```

String str ="My dog " + name + " is " + age;

System.out.println(str);

And
StringBuilder sb = new StringBuilder();

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

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

Question := 27
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

Question := 28

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

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

Question := 29

Given:

```
public class MainMethod {

void main() {

System.out.println("one");

}

static void main(String args) {

System.out.println("two");

}

public static void main(String[] args) {

System.out.println("three");

}
void mina(Object[] args) {

}

}
```

What is printed out when the program is excuted?

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

Question := 30

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

A. Compilation fails.
B. B. The third argument is given the value null.
C. C. The third argument is given the value void.
D. D. The third argument is given the value zero.
E. E. The third argument is given the appropriate falsy value for its declared type. F) An exception occurs when the met

Question := 31
What is the proper way to defined a method that take two int values and returns their sum as an int value?

A. int sum(int first, int second) { first + second; }
B. int sum(int first, second) { return first + second; }
C. sum(int first, int second) { return first + second; }
D. int sum(int first, int second) { return first + second; }
E. void sum (int first, int second) { return first + second; }

Question := 32
Given the for loop construct:

for (expr1 ; expr2 ; expr3) {

statement;

}

Which two statements are true?

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

Question := 33
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

Question := 34
Given:

```
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

Question := 35
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.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,
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 (MissingInfoException e2) { //code goes here}
catch (AgeOutofRangeException e3) {//code goes here}
D. Enclosing line 6 7 8 and line 9 within a try block and adding:
catch (MissingInfoException e2) { //code goes here}
catch (AgeOutofRangeException e3) {//code goes here}

Question := 36

Given the code format:

```
class DBConfiguration  {
       String user;
      String password;
}
//And:
class DBHandler  {
    DBConfiguration configuration(String uname, String password)   {
```

```
            // insert code here
    }
    public static void main(String[] args)   {
        DBHandler r = new DBHandler();
        DBConfiguration dbconf = r.configuration("manager", "manager");
    }
}
```

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;
E. retutn null;

Question := 37

Given:

```
class Best2  {
    public static void main(String[] args)   {
        int ar1[] = {2, 4, 6, 8};
        int ar2[] = {1, 3, 5, 7, 9};
        ar2 = ar1;
        for (int e2 : ar2)    {
            System.out.print(" " + e2);
        }
    }
}
```

What is the result?

A. 2 4 6 8
B. 2 4 6 8 9
C. 1 3 5 7
D. 1 3 5 7 9

Question := 38

Given:

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

```
      int  num = 5;
      int  sum ;
      do {
         sum  +=  num ;
      }while(num   >  1);
      System.out.println ("the sun is " + sum +  "," ) ;
   }
}
```

What is the result?

A. The sum is 2
B. The sum is 14
C. The sum is 15
D. The loop executes infinite times
E. Compilation fails

Question := 39

Given the code fragment:

```
public class Q116 {
  public static void main(String[] args) {
    int  [ ]  fst   = {1,  2,  3,  4,  5,  4,  3, 2,  1}  ;
    int sum   =  0;
    for(int frnt=0,rear=fst.length-1;frnt<5 && rear>=5;frnt ++,rear --){
       sum = sum+fst[frnt]+fst[rear]  ;
    }
    System.out.print(sum);
  }
}
```

What is the result?

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

Question := 40

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.

Question := 41

Given:

```
abstract class Shape   {
    private   int   x;
    private   int   y;
    public abstract void draw() ;
    public void  setAnchor  (int   x, int   y )   {
     this.x   =  x;
     this.y   =  y;
   }
}
```

Which two classes use the shape class correctly?

A)
```
public class Circle implements Shape    {
    private int radius ;
  }
```
B)
```
public abstract class Circle extends Shape  {
   private   int radius ;
  }
```
C)
```
public class Circle extends  Shape   {
   private int radius    ;
   public void draw ( ) ;
}
```
D)
```
public abstract class circle implements  Shape   {
   private int radius    ;
   public void draw ( ) ;
}
```
E)
```
public class Circle extends  Shape   {
   private int radius    ;
   public void draw ( ) ; (/* code here * /)
}
```
F)
```
public abstract class circle implements  Shape   {
   private int radius    ;
   public void draw ( ) ;  (/* code here * /)
}
```

A. Option A

B. Option B

C. Option C

D. Option D
E. Option E
F. Option F

Question := 42
Given:

```
public class Q124 {
  public  static  String  doMsg(char  x )  {
     return   "Good Day !  "  ;
  }
  public  static  String doMsg  (int   y)  {
     return  "Good Luck !  " ;
  }
  public static void main(String[] args) {
     char  x  =  8  ;
     int  z   =   '8' ;
     System . out . println  (doMsg(x));
     System . out . println  (doMsg(z));
   }
}
```

What is the result?

A. Good Day!
   Good Luck!
B. Good Day!
   Good Day!
C. Good Luck!
   Good Day!
D. Good Luck!
   Good Luck!
E. Compilation fails

Question := 43
Given the code fragments:

```
import java.util.*;

interface Contract   {  }
class Super implements Contract  {  }

public class Sub extends Super {
   public static void main(String[] args) {
     List objs  =  new ArrayList  ( ) ;
```

```
    Contract c1 = new Super ( ) ;    // line n1
    Contract c2 = new Sub ( ) ;
    Super s1    =  new  Sub  ( ) ;
    objs.add(c1) ;
    objs.add(c2) ;
    objs.add(s1) ;                   // line n2
    for (Object itm : objs)     {
     System.out.println(itm.getClass().getName()) ;
    }
  }
}
```

What is the result?

A. Super
Sub
Sub
B. Contract
Contract
Super
C. Compilation fails at line n1
D. Compilation fails at line n2

Question := 44
Given the code fragment:

```
import java.util.*;
interface Contract  {  }
class Super  implements Contract  {  }
class Sub  extends  Super  { }

public class Q130 {
      public static void main(String[] args) {
         List objs  =  new ArrayList  ( )  ;
         Contract  c1  =  new Super  ( )  ;
         Contract  c2  = new Sub  ( )  ;
         objs.add (c1) ;
         objs.add (c2) ;
         objs.add (c1) ;
         for  (Object Itm : objs )      {
             System. out . println (Itm.getClass().getName( )) ;
```

```
      }
   }
}
```

A. Super
Sub
Sub
B. Super
Sub
Super
C. Contract
Contract
Super
D. Compilation fails at line n1
E. Compilation fails at line n2

Question := 45
Given:

```
class Vowel {
   private  char  var ;
   public static void main(String[] args) {
      char var1  =  'a' ;
      char  var2  =  var1;
      var2  =  'e' ;
      Vowel  obj1  =  new Vowel () ;
      Vowel  obj2  =  obj1;
      obj1.var   =   'I' ;
      obj2.var   =  'o' ;
      System.out.println(var1 +  ",  "   + var2);
      System.out.print(obj1.var  + ", "  +  obj2.var ) ;
   }
}
```

A. a, e
   i, o
B. a, e
   o, o
C. e, e
   I, o
D. e, e
   o, o

Question := 46
Given:

```
class  TestApp  {
   public  static  void  main  (String [ ]  args )  {
     TestApp  t  =  new  TestApp  ( ) ;
     try  {
           t.doList  ( ) ;
           t.doPrint ( ) ;

     }catch(Exception  e2 )  {
      System.out.println( " caught   "   +   e2 ) ;
     }
   }
   public  void  doList  ( )  throws  Exception  {
     throw new Error  ( " error ") ;
   }
   public void  doPrint()  throws  Exception  {
     throw  new  RuntimeException ( " exception ") ;
  }
}
```

What is the result?

(A) Caught java. lang. runtime exception : exception
Exception in thread "main" java . lang. error : error
at test app . do list (test app . java : 14)
at test app . main (test app . java ; 6)

(B) Exception in thread Exception in thread "main" java.lang.Error: error
at TestApp.doList(TestApp.java:13)
at TestApp.main(TestApp.java:5)

(C) Caught java. Lang. runtime exception : exception
Caught java. Lang. error : error

(D) Caught java. Lang. runtime exception : exception

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

Question := 47

Given the code fragment:

```
import java.util.*;
class  Q140  {
 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 ArrayList update(String [] strs)
C. static List update (String [] strs)
D. static void update (String[] strs)
E. ArrayList static update(String [] strs)

Question := 48

Given:

```
class Access {
  private int x = 0 ;
  private int y = 0 ;
  public static void main (String [ ] args ) {
    Access accapp = new Access ( ) ;
    accapp.printThis (1, 2 ) ;
    accapp.printThat (3, 4) ;
   }
  public void printThis (int x, int y ) {
    x = x ;
    y = y ;
    System.out.println("x: " + this.x + " y: " + this.y);
  }
   public void printThat(int x, int y ) {
    this.x = x ;
    this.y = y ;
    System.out.println("x: " + this.x + " y: " + this.y);
  }
```

```
}
```

What is the result?

A. x: 1 y: 2
   x. 3 y: 4
B. x: 0 y: 0
   x: 3 y: 4
C. x: 1 y: 2
   x: 0 y: 0
D. x: 0 y: 0
   x: 0 y: 0

Question := 49

Given:

```
class test2  {
     public static  void doChange  (int[] arr )  {
          for  ( int  pos  =  0 ;  pos  <  arr . length ; pos++) {
             arr [pos]=  arr [pos] +  1 ;
          }
     }
     public static void main  ( String  []  args )   {
       int  [] arr   = { 10 ,  20 ,  30 , };
       doChange(arr ) ;
       for(int  x :  arr )   {
          System.out.print  (x  +  ",  " ) ;
       }
       doChange(arr[0],arr[1],arr[2]) ;
       System.out.print(arr[0]+","+arr[1]+","+arr[2]) ;
         }
}
```

What is the result?

A. 11, 21, 31, 11, 21, 31
B. 11, 21, 31, 12, 22, 32
C. 12, 22, 32, 12, 22, 32
D. 10, 20, 30, 10, 20, 30
E. Compilation fails

Question := 50

Given:

```
class Jump {
   static String args  [ ]  =  {  "lazy",   "lion", "is",  "always" } ;
```

```
   public static void main(String[] args) {
      System.out.println(args[1]+ " " + args[2]+ " "+args[3]+" jumping") ;
   }
}
```

And the commands:

Javac Jump.java

Java Jump crazy elephant is always

What is the result?

A. Lazy lion is jumping
B. Lion is always jumping
C. Crazy elephant is jumping
D. Elephant is always jumping
E. Compilation fails

Question := 51
Given:

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

And the code fragment :

```
public class Test{
   public static noid main(String arge[]){
      List ps = new ArrayList();
      Patient p2 = new Patient("Mike");
      ps.add(p2);

      //  insert Code here

      if (f >=0 ) {
         System.out.print("Mike Found");
      }
   }
}
```

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);

Question := 52
Given:

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

Question := 53
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.println("Successful");
      else
         System.out.println("Unsuccessful");
   }
}
```

What is result?

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

Question := 54

Given the fragment:

```
24.    float var1 = (12_345.01 > 123_45.00) ? 12_456:124_56.02f;
25.    float var2 = var1+1024;
26.    System.out.print(var2);
```

What is the result?

A. 13480.0
B. 13480.02
C. Compilation fails
D. An exception is thrown at runtime

Question := 55

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

Question := 56
Given:

```
import java.util.*;

public class Ref {

public static void main(String[] args) {

StringBuilder s1 = new StringBuilder("Hello Java!");

String s2 = s1.toString();

List<String> lst = new ArrayList<String>();

lst.add(s2);

System.out.println(s1.getClass());
System.out.println(s2.getClass());

System.out.println(lst.getClass());

}
```

}

What is the result?

A. class java.lang.String
class java.lang.String
class java.util.ArrayList
B. class java.lang.Object
class java.lang. Object
class java.util.Collection
C. class java.lang.StringBuilder
class java.lang.String
class java.util.ArrayList
D. class java.lang.StringBuilder
class java.lang.String
class java.util.List

Question := 57
Given:

public class Marklist {

int num;

public static void graceMarks(Marklist obj4) {

obj4.num += 10;

}

public static void main(String[] args) {
MarkList obj1 = new MarkList();

MarkList obj2 = obj1;

MarkList obj1 = null;

obj2.num = 60;

graceMarks(obj2);

}

```
}
```

How many objects are created in the memory runtime?

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

Question := 58
Given:

```
public class Painting {

private String type;

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public static void main(String[] args) {
Painting obj1 = new Painting();

Painting obj2 = new Painting();

obj1.setType(null);

obj2.setType("Fresco");

System.out.print(obj1.getType() + " : " + obj2.getType());

}

}
```

What is the result?

A. : Fresco
B. null : Fresco
C. Fresco : Fresco
D. A NullPointerException is thrown at runtime

Question := 59
Given the code fragment:

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

int iArray[] = {65, 68, 69};

iArray[2] = iArray[0];

iArray[0] = iArray[1];

iArray[1] = iArray[2];

for (int element : iArray) {

System.out.print(element + " ");

}
```

A. 68, 65, 69
B. 68, 65, 65
C. 65, 68, 65
D. 65, 68, 69
E. Compilation fails

Question := 60
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 = 1
ii = 1

Question := 61
Given:

```
class MarksOutOfBoundsException extends IndexOutOfBoundsException { }

public class GradingProcess {

void verify(int marks) throws IndexOutOfBoundsException {

if (marks > 100) {

throw new MarksOutOfBoundsException();

}

if (marks > 50) {

System.out.print("Pass");

} else {

System.out.print("Fail");

}
```

```
}

public static void main(String[] args) {
int marks = Integer.parseInt(args[2]);

try {

new GradingProcess().verify(marks));

} catch(Exception e) {

System.out.print(e.getClass());

}

}
}
```

And the command line invocation:

Java grading process 89 50 104
What is the result?
A. Pass
B. Fail
C. Class MarketOutOfBoundsException
D. Class IndexOutOfBoundsException
E. Class Exception
Question := 62
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 declaration of 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 ( );

Question := 63
Given the code fragment:

```
String[] cartoons = {"tom","jerry","micky","tom"};

int counter =0;

if ("tom".equals(cartoons[0])) {

counter++;

} else if ("tom".equals(cartoons[1])) {
counter++;
```

```
} else if ("tom".equals(cartoons[2])) {

counter++;

} else if ("tom".equals(cartoons[3])) {

counter++;

}

System.out.print(counter);
```

What is the result?

A. 1
B. 2
C. 4
D. 0

Question := 64

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

Question := 65
Given:

```
class Sports {

int num_players;

String name, ground_condition;

Sports(int np, String sname, String sground){

num_players = np;

name = sname;

ground_condition = sground;

}

}

class Cricket extends Sports {

int num_umpires;

int num_substitutes;
```

Which code fragment can be inserted at line //insert code here to enable the code to compile?

A. Cricket() {
super(11, "Cricket", "Condidtion OK");
num_umpires =3;
num_substitutes=2;
}
B. Cricket() {
super.ground_condition = "Condition OK";
super.name="Cricket";
super.num_players = 11;
num_umpires =3;
num_substitutes=2;
}
C. Cricket() {
this(3,2);
super(11, "Cricket", "Condidtion OK");
}
Cricket(int nu, ns) {
this.num_umpires =nu;
this.num_substitutes=ns;
}
D. Cricket() {
this.num_umpires =3;
this.num_substitutes=2;
super(11, "Cricket", "Condidtion OK");
}

Question := 66

Which statement is true about the default constructor of a top-level class?
A. It can take arguments.
B. It has private access modifier in its declaration.
C. It can be overloaded.
D. The default constructor of a subclass always invokes the no-argument constructor of its superclass.

Question := 67

Given:

public class Natural {

private int i;

void disp() {

while (i <= 5) {

```
for (int i=1; i <=5;) {

System.out.print(i + " ");

i++;

}

i++;

}

}

public static void main(String[] args) {

new Natural().disp();

}

}
```

What is the result?

A. Prints 1 2 3 4 5 once
B. B. Prints 1 3 5 once
C. C. Prints 1 2 3 4 5 five times
D. D. Prints 1 2 3 4 5 six times
E. E. Compilation fails

Question := 68

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

Question := 69
Given:

```
public class Test {

public static void main(String[] args) {

int ax = 10, az = 30;

int aw = 1, ay = 1;

try {

aw = ax % 2;

ay = az / aw;

} catch (ArithmeticException e1) {

System.out.println("Invalid Divisor");

} catch (Exception e2) {

aw = 1;

System.out.println("Divisor Changed");

}

ay = az /aw; // Line 14

System.out.println("Succesful Division " + ay);

}

}
```

What is the result?

A. Invalid Divisor
Divisor Changed
Successful Division 30
B. Invalid Divisor
Successful Division 30
C. Invalid Divisor
Exception in thread "main" java.lang.ArithmeticException: / by zero at test.Teagle.main(Teagle.java:14)
D. Invalid Divisor
Exception in thread "main" java.lang.ArithmeticException: / by zero at test.Teagle.main(Teagle.java:14)
Successful Division 1

Question := 70
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 */
}
```

Question := 71

Given:

```
public class ColorTest {

public static void main(String[] args) {

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

int count = 0;

for (String c : colors) {

if (count >= 4) {

break;

}

else {
```

```
continue;

}

if (c.length() >= 4) {

colors[count] = c.substring(0,3);
}

count++;

}

System.out.println(colors[count]);

}

}
```

What is the result?

A. Yellow
B. Maroon
C. Compilation fails
D. A StringIndexOutOfBoundsException is thrown at runtime.

Question := 72
Given the code fragment:

```
public class Test {
    public static void main(String[] args) {
        boolean isChecked = false;
        int arry[] = {1,3,5,7,8,9};
        int index = arry.length;
        while ( <code1> ) {
            if (arry[index-1] % 2 ==0) {
                isChecked = true;
                <code2>
    }
        }
        System.out.print(arry[index]+", "+isChecked);
    }
```

}

Which set of changes enable the code to print 1, true?

A. Replacing <code1> with index > 0 and replacing <code2> with index--;
B. Replacing <code1> with index > 1 and replacing <code2> with --index;
C. Replacing <code1> with index > 5 and replacing <code2> with --index ;
D. Replacing <code1> with index and replacing <code2> with --index ;