Question := 1

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

Question := 2

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.

Question := 3

Given the code fragment:

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

What is the result?

A. 4
null

B. null

C. An IllegalArgumentException is thrown at run time

D. 4
An ArrayIndexOutOfBoundException is thrown at run time

Question := 4

Given:

```
class X {
       String str = "default";
       X (String s)  {str =s;}
       void print () {System.out.println (str);  }
        public static void main (String[] args) {
                 new X ("hello").print();
        }
}
```

What is the result?

A. hello

B. Default

C. Compilation fails

D. The program prints nothing

E. An exception is thrown at run time

Question := 5

Which code fragment is illegal?

```
A)    class Base1 {
        abstract class Abs1  { }
 }
B)    abstract class Abs1  {
         void doit () { }
}
C)    class Base1  { }
abstract class Abs1 extends Base1 { }

D)    abstract int var1 =89;
```

A. Option A

B. Option B

C. Option C

D. Option D

Question := 6

Given a java source file:

```
class X {
          X () { }
          private void one () { }
}
 class Y extends X {
    Y()  { }
     private void two () {one() ; }
     public static void main (String[] args)  {
        new Y().two();
     }
}
```

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

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

Question := 7

Given:

```
1.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

Question := 8

Given the code fragment:

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

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

What is the result?

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

Question := 9

Given the code fragment:

```
String color = “Red”;

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

What is the result?

A. Found Red
B. Found Red
Found Blue

C. Found Red
Found Blue
Found White
D. Found Red
Found Blue
Found White
Found Default

Question := 10
Given:

```
5.    // insert code here
6.        public abstract void bark();
7.    }
8.
9.    // insert code here
10.   public void bark()   {
11.    System.out.println (“woof”);
12.     }
13.  }
```

What code should be inserted?

A) 5. class Dog {
     9. class Poodle extends Dog {

B) 5. abstract Dog {
     9. class Poodle extends Dog {

C) 5. abstract class Dog {
     9. class Poodle extends Dog {

D) 5.class Dog {
     9. class Poodle implements Dog {

E)   5. abstract Dog {
      9. class Poodle implements Dog {

F   5. abstract class Dog {
     9.   class Poodle implements Dog {

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

Question := 11

Given the code fragment:

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

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

```
1 2 3 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

Question := 12

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

A. Compilation fails.

B. The third argument is given the value null.

C. The third argument is given the value void.

D. The third argument is given the value zero.

E. The third argument is given the appropriate false value for its declared type.

F. An exception occurs when the method attempts to access the third argument.

Question := 13

Which three are advantages of the Java exception mechanism?

A. Improves the program structure because the error handling code is separated from the normal program function

B. Provides a set of standard exceptions that covers all the possible errors

C. Improves the program structure because the programmer can choose where to handle exceptions

D. Improves the program structure because exceptions must be handled in the method in which they occurred

E. allows the creation of new exceptions that are tailored to the particular program being

Question := 14

Given:

Which line causes a compilation error?

```
class ScopeTest1  {
    public static void main(String[] args)  {
        doStuff();                    // line x1
        int x1 = x2;                  // line x2
        int x2 = j;                   // line x3
    }
    static void doStuff()   {
        System.out.println(j);     // line x4
    }
    static int j;
}
```

A. line x1

B. line x2

C. line x3

D. line x4

Question := 15

Given:

```
class Basic  {
     private static int letter;
     public static int getLetter();
```

```
        public static int 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

Question := 16
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:
Public int 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.price = 4.0;
E. Change line 5 to the following:
price.price = 4;
F. Change line 5 to the following:
price = (float) 4:
G. Change line 5 to the following:
Price = (Simple) 4;
H. The code compiles and runs properly; no changes are necessary

Question := 17
Identify two benefits of using ArrayList over array in software development.

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

Question := 18

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")

Question := 19

Given:

```
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.

Question := 20

Given:

```
class SampleClass  {
          public static void main (String [ ]args) {
                 SampleClass sc, scA, scB;
                 sc = new SampleClass ();
                 scA = new SampleClassA ();
                 scB = new SampleClassB ();
                 System.out.println ("Harsh is : "+
              sc.getHarsh() + ", " + scA.getHarsh()+ ", " +scB.getHarsh() );
          }
          public  int getHarsh ()   {
                return 111111;
          }
}
class SampleClassA extends SampleClass  {
      public long getHarsh()  {
        return 44444444;
      }
}
class SampleClassB extends SampleClass  {
      public long getHarsh()  {
                return 999999999;
      }
}
```

What is the result?

A. Compilation fails

B. An exception is thrown at runtime

C. There is no result because this is not correct way to determine the hash code

D. Hash is: 111111, 44444444, 999999999

Question := 21

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

Question := 22

The catch clause argument is always of type___________.

A. Exception

B. Exception but NOT including RuntimeException

C. Throwable

D. RuntimeException

E. CheckedException

F. Error

Question := 23

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

```
    public static void doStuff() {
     if (Math.random() > 0.5) throw new RuntimeException();
     doMoreStuff();
     System.out.println("3 ");
    }
    public static void doMoreStuff() {
        System.out.println("4");
    }
}
```

Which two are possible outputs?

A. 2

B. 4

C. 1

D. 1

Question := 24

Given:

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

What is the result?

A. aa
   cc

```
B.  aa
    bb
    cc
C. cc
   dd
D. cc
```

E. Compilation fails.

Question := 25

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 ();
      bob.name = "Bob";
      bob.age = 18;
      bob.year = 1982;
   }
}
```

What is the result?

A. year is set to 1982.

B. bob.year is set to 1982

C. A runtime error is generated.

D. A compile time error is generated.

Question := 26

View the exhibit:

```
public class Student {

public String name = "";

public int age = 0;

public String major = "Undeclared";

public boolean fulltime = true;

public void display() {

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

public boolean isFullTime() {

return fulltime;

}

}
```

Which line of code initializes a student instance?

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

Question := 27

```
boolean log3 = ( 5.0 != 6.0) && ( 4 != 5);

boolean log4 = (4 != 4) || (4 == 4);

System.out.println("log3:"+ log3 + \nlog4" + log4);
```

What is the result?

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

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

Question := 28

Which two are valid array declaration?

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

Question := 29

Given:

```
public class ScopeTest {

int j, int k;

public static void main(String[] args) {

new ScopeTest().doStuff(); }

void doStuff() {

int x = 5;

doStuff2();
System.out.println("x");

}

void doStuff2() {

int y = 7;

System.out.println("y");

for (int z = 0; z < 5; z++) {

System.out.println("z");

}
```

Which two items are fields?

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

Question := 30

Given:

```
public class SampleClass {

public static void main(String[] args) {

AnotherSampleClass asc = new AnotherSampleClass(); SampleClass sc = new SampleClass();

sc = asc;

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

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

}}

class AnotherSampleClass extends SampleClass {

}
```

What is the result?

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

Question := 31

Which two are Java Exception classes?

A. SercurityException

B. DuplicatePathException

C. IllegalArgumentException

D. TooManyArgumentsException

Question := 32

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 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 := 33

Given:

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

Question := 34

Given:

```
class A { }

class B { }

Interface X { }

Interface Y { }
```

Which two definitions of class C are valid?

A. class C extends A implements X { }
B. class C implements Y extends B { }
C. class C extends A, B { }
D. class C implements X, Y extends B { }
E. class C extends B implements X, Y { }

Question := 35

Given:

```
class Best   {
  static void dispResult (int[] num)    {
    try {
      System.out.println(num[1] / (num[1] - num[2]));
    }catch (ArithmeticException e){
        System.err.println("First exception");
    }
    System.err.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. 0

Done

B. First Exception

Done

C. Second Exception

D. Done

Third Exception

E. Third Exception

Question := 36

Given the code fragment:

```
public class Q108 {
    public static void main(String[] args) {
        String color = "teal";
        switch  (color)   {
            case "Red":
                    System.out.println("Found Red");
             case "Blue":
                    System.out.println("Found Blue");
            case "Teal":
                    System.out.println("Found Teal");
                    break;
            default:
                    System.out.println("Found Default");
        }
    }
}
```

What is the result?

A. Found Red

Found Default

B. Found Teal

C. Found Red

Found Blue

Found Teal

D. Found Red

Found Blue

Found Teal

Found Default
E. Found Default

Question := 37

Given:

```
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

Question := 38

Given:

```
class testry   {
        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. hEllO jAvA

B. Hello java!

C. out   of limits

hEllO jAvA

D. out   of limits

Question := 39

Given:

```
public class Q117 {
    public static void main(String[] args) {
        String theString   =  "hello world" ;
        System.out.println   (theString.charAt(11)) ;
    }
}
```

What is the result?

A. The program prints nothing

B. d

C. A StringIndexOutOfBoundsException is thrown at runtime.

D. AnArrayIndexOutOfBoundsException is thrown at runtime.

E. A NullPointerException is thrown at runtime.

Question := 40

Given:

```
public class Q120 {
    public static void main(String[] args) {
        int []  xx = null ;
        for  (int ii  :  xx )    {
               System . out . println (ii) ;
           }
   }
}
```

What is the result?

A. Null

B. Compilation fails

C. An exception is thrown at runtime

D. 0

Question := 41

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,

```
List <Alpha1 >   Strs  =  new ArrayList <Alpha1 >  ( ) ;
         Strs.add (new Alpha1() ) ;
         Strs.add (new Beta( )  ) ;
         Strs.add (new  Gamma( )   )  ;
         for(Alpha1  t   :  Strs)     {
            System.out.println(t.doStuff("java ")  )  ;
         }
```

What is the result?

A. Java
Java
Java
B. Java
Java
va
C. java
jeve
va
D. Compilation fails

Question := 42

Given the fragments:

```
class Q126  extends Root   {
   public static void main ( String  [] args ){
     Root  r  =  new  Q126 ( )  ;
     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

Question := 43

Given:

```
class Star   {
   public  void  doStuff  ( )     {
      System.out.println(" Twinkling star " )  ;
   }
 }
 interface Universe     {
   public  void  doStuff();
 }
class Sun extends Star  implements Universe  {
   public void doStuff  ( )  {
       System.out.println("shining sun ")  ;
   }
}
```

```
class Bob   {
   public static void main(String[] args) {
      Sun obj2  =  new Sun ( ) ;
      Star obj3  =  obj2 ;
      ((Sun )obj3).doStuff ( ) ;
      ((Star )obj2).doStuff ( ) ;
      ((Universe )obj2).doStuff ( ) ;
   }
}
```

What is the result?

A. Shining Sun
Shining Sun
Shining Sun
B. Shining Sun
Twinkling Star
Shining Sun
C. Compilation fails
D. A ClassCastException is thrown at runtime

Question := 44

Given:

```
abstract  class  XX   {
   public  abstract void methodX ( )  ;
}
 interface   YY   {
   public void  methodY   ( ) ;
}
```

Which two code fragments are valid?

```
A)    class ZZ  extends  XX  implements YY  {
            public void method  z ( ) { }
      }
B)  abstract class  ZZ  extends XX implements YY {
     public void methodZ  ( )  { }
   }
C)    class  ZZ  extends  XX   implements  YY {
```

```
                public void method X  ( )   { }
        }
D)    abstract   class ZZ  extends XX implements  YY  {
 }

E)    class ZZ   extends   XX   implements  YY{
                public void method Y  ( )   { }
        }
```

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

Question := 45

Given:

```
class Circle1  {
              double radius;
              public double area ;
              public Circle1( double r ) {  radius = r  ;    }
              public double getRadius ()   {  return radius ;   }
              public void setRadius (double  r)   {  radius= r ;   }
              public  double getArea  ( ) {return  / * ???  */;    }
  }
  class  App1   {
              public static  void main (String [ ] args)     {
                Circle1 c1 = new Circle1 (17.4);
c1.area = Math.PI*c1.getRadius()*c1.getRadius();
          }
    }
```

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

Which two modifications are necessary to ensure that the class is being properly encapsulated?

A. Remove the area field.
B. Change the getArea( ) method as follows:
public double getArea ( ) { return Math.PI * radius * radius; }
C. Add the following method:
public double getArea ( ) {area = Math.PI * radius * radius; }
D. Change the access modifier of the setRadius ( ) method to be protected.

Question := 46

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.

Question := 47

Given the code fragment:

```
System.out.println  (  28  +  5  <=  4  +  29  )  ;
System.out.println   (   (  28  +  5  )  <=  (4+29)     ) ;
```

What is the result?

A. 28false29
    true

B. 285 < 429
    true

C. true
    true

D. compilation fails

Question := 48

Given:

```
class SpecialException  extends Exception  {
    public SpecialException (String  message ){
            super(message);
            System.out.println  (message ) ;
     }
 }
 class ExceptionTest     {
      public static void main( String []  args  )  {
         try  {
            dosomething() ;
         }
         catch  ( SpecialException   e )  {
            System.out.println (e) ;
         }
   }
   static void dosomething()throws SpecialException   {
      int   [ ]  ages   =  new int  [ 4 ]  ;
      ages  [ 4 ]   =  17 ;
      dosomethingelse  ()  ;
   }
   static void dosomethingelse ()  throws SpecialException   {
      throw  new  SpecialException("thrown at and of dosomething()method");
   }
}
```

What will be the output?
(A)   SpecialException : thrown at end of dosomothing () method
(B)   Error in thrown   “main “   java .lang.ArrayIndexOutOf Bounds error
(C)   Exception In thread “ main “ java.lang .ArrayIndexOutOf BoundsException :   4
At Exceptiontest . dosomething (ExceptionTest . java : 18)
At Exceptiontest .main (ExceptionTest . java : 10)
(D) SpecialException: Thrown at end of dosomothing () method
at ExceptionTest . dosomethingelse (ExceptionTest . java : 16)
at ExceptionTest . dosomething (ExceptTontest . java : 13)
at ExceptionTest. Main (ExceptionTest . java : 4)

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

Question := 49
Given:

```
class  palindrome    {
   public static  int  main  (String  [ ] args )  {
      System.out.print(args [ 1 ] );
         return  0;
   }
}
```

And the commands :

```
Javac palindrome . java
Java  palindrome wow mom
```

What is the result?

A. Compilation fails
B. The code compiles, but does not execute.
C. Paildrome
D. Wow
E. Mom

Question := 50

Given:

```
class Nest{
        public static void main (String []  args ){
                int numbers [ ] ;
                numbers   =  new  int  [ 2 ]  ;
                numbers  [ 0 ]   =  10 ;
                numbers  [ 1 ]   =  20 ;
                numbers  =  new int [ 4 ] ;
                numbers  [ 2 ]    =    30 ;
                numbers  [ 3 ]    =    40 ;
                for  (int   x   :  numbers ) {
                    System.out.print(" "+x);
                }
        }
```

What is the result?

A. 10 20 30 40
B. 0 0 30 40
C. Compilation fails
D. An exception is thrown at runtime

Question := 51

Give:

```
class Alpha{
    public String [] main = new String[2];
    Alpha(String main[]){
        for(int ii=0;ii<main.length;ii++){
            this.main[ii]=main[ii]+5;
        }
    }
    public  void  main(){
        System.out.print(main[0]);
    }
}
class Test{
    public static void main(String[] args) {
        Alpha main = new Alpha(args);
        main.main();
    }
}
```

And the Commands:

```
javac Test.java
```

```
java Test 1 2
```

What is the result?

A. 15
B. 13
C. Compilation fails
D. An exception is thrown at runtime
E. The program fails to execute due to runtime error

Question := 52

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 y1;
   Z(){
      x1=3;
      x2=20;
      y1=100;
   }
}
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

Question := 53

Given:

```
public class Series  {
   private boolean flag;
   private void displaySeries(){
   int num =2;
   while(flag){
      if(num%7==0)
         flag=false;
      System.out.print(num);
      num +=2;
   }
   }
   public static void main(String [] args){
       new Series().displaySeries();
   }
}
```

What is the result?

A. 2 4 6 8 10 12

B. 2 4 6 8 10 12 14

C. Compilation fails

D. The program prints multiple of 2 infinite times

E. The program prints nothing

Question := 54

Given the code fragment:

```
int row =10;
for(;row > 0;){
  int col=row;
  while (col > 0){
    System.out.print(col + " ");
    col-=2;
  }
  row=row/col;
}
```

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

Question := 55

Given:

```
class Mid {

public int findMid(int n1, int n2) {

return (n1 + n2) / 2;

}

}
```

```
public class Calc extends Mid {

public static void main(String[] args) {

int n1 = 22, n2 = 2;

// insert code here

System.out.print(n3);


}
```

Which two code fragments, when inserted at // insert code here, enable the code to compile and print 12?

```
A. Calc c = new Calc();
int n3 = c.findMid(n1,n2);
B. int n3 = super.findMid(n1,n3);
C. Calc c = new Mid();
int n3 = c.findMid(n1, n2);
D. Mid m1 = new Calc();
int n3 = m1.findMid(n1, n2);
E. int n3 = Calc.findMid(n1, n2);
```

Question := 56

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;

// line n2
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;import org.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;

Question := 57
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";
c.model = 1200;
return c;
}
public static void bake2(Cake c) {
c.flavor = "Chocolate";
c.model = 1230;
return;
}
}.
```

What is the result?

A. 0 unknown

0 unknown

B. 1200 Strawberry

1200 Strawberry

C. 1200 Strawberry

1230 Chocolate

D. Compilation fails

Question := 58

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

1

C. Invalid Index
D. Compilation fails

Question := 59
Given:

```
public class TestLoop1 {
public static void main(String[] args) {
int a = 0, z=10;
while (a < z) {
a++;
--z;
}

System.out.print(a + " : " + z);

}

}
```

What is the result?

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

Question := 60

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

Question := 61

Given the code fragment:

```
StringBuilder sb = new StringBuilder ( ) ;

Sb.append ("world");
```

Which code fragment prints Hello World?

A. sb.insert(0,"Hello ");

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

Question := 62

Given the code fragment:

```
public class Test {

static String[][] arr =new String[3][];

private static void doPrint() {
//insert code here

}

public static void main(String[] args) {

String[] class1 = {"A","B","C"};
String[] class2 = {"L","M","N","O"};

String[] class3 = {"I","J"};

arr[0] = class1;
arr[1] = class2;
arr[2] = class3;
Test.doPrint();
}
}
```

Which code fragment, when inserted at line //insert code here, enables the code to print COJ?

```
A. int i = 0;
   for (String[] sub: arr) {
       int j = sub.length -1;
       for (String str: sub) {
              System.out.println(str[j]);
              i++;
       }
   }
```

```
B. for (int i = 0;i < arr.length;i++) {
        int j = arr[i].length-1;
        System.out.print(arr[i][j]);
    }

C. int i = 0;
    for (String[] sub: arr[][]) {
         int j = sub.length;
        System.out.print(arr[i][j]);
        i++;
    }
D. for (int i = 0;i < arr.length-1;i++) {
         int j = arr[i].length-1;
        System.out.print(arr[i][j]);
        i++;
   }
```

Question := 63

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

Question := 64

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

Question := 65

Given:

```
public class X {
```

```
static int i;

int j;

public static void main(String[] args) {

X x1 = new X();

X x2 = new X();

x1.i = 3;

x1.j = 4;

x2.i = 5;

x2.j = 6;

System.out.println(

x1.i + " "+

x1.j + " "+

x2.i + " "+

x2.j);

}

}
```

What is the result?

A. 3 4 5 6
B. 3 4 3 6
C. 5 4 5 6
D. 3 6 4 6

Question := 66
Given:

```
public class TestField {
```

```
int x;

int y;

public void doStuff(int x, int y) {

this.x = x;

y =this.y;

}
public void display() {

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

}

public static void main(String[] args) {

TestField m1 = new TestField();

m1.x = 100;

m1.y = 200;

TestField m2 = new TestField();

m2.doStuff(m1.x, m1.y);

m1.display();

m2.display();

}

}
```

What is the result?

A. 100 200 : 100 200
B. 100 0 : 100 0 :
C. 100 200 : 100 0 :

D. 100 0 : 100 200 :

Question := 67

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 is the result?

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

Question := 68

Given:

```
interface Pet { }

class Dog implements Pet { }

public class Beagle extends Dog{ }
```

Which three are valid?

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

Question := 69

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. 011223
B. 012321
C. 123456
D. 000000

Question := 70

Given:

```
abstract class A1 {

public abstract void m1();
public void m2() { System.out.println("Green"); }

}

abstract class A2 extends A1 {

public abstract void m3();

public void m1() { System.out.println("Cyan"); }

public void m2() { System.out.println("Blue"); }

}

public class A3 extends A2 {

public void m1() { System.out.println("Yellow"); }

public void m2() { System.out.println("Pink"); }

public void m3() { System.out.println("Red"); }

public static void main(String[] args) {

A2 tp = new A3();

tp.m1();

tp.m2();

tp.m3();
}

}
```

What is the result?
A. Yellow
Pink
Red
B. Cyan
Blue
Red
C. Cyan

Green
Red
D. Compilation Fails

Question := 71
Given:

```
public class App {

// Insert code here

System.out.print("Welcome to the world of Java");

}

}
```

Which two code fragments, when inserted independently at line // Insert code here, enable the program to

execute and print the welcome message on the screen?

A. static public void main (String [] args) {
B. static void main (String [] args) {
C. public static void Main (String [] args) {
D. public static void main (String [] args) {
E. public void main (String [] args) {

Question := 72
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