Oracle

Exam 1z0-808

Java SE 8 Programmer I

Version: 6.0

[Total Questions: 236]

Question No : 1

Given the code fragment:

```
public class Test {

    static int count = 0;
    int i = 0;

    public void changeCount() {
        while (i < 5) {
            i++;
            count++;
        }
    }

    public static void main(String[] args) {
        Test check1 = new Test();
        Test check2 = new Test();
        check1.changeCount();
        check2.changeCount();
        System.out.print(check1.count + " : " + check2.count);
    }
}
```

What is the result?

A. 10 : 10
B. 5 : 5
C. 5 : 10
D. Compilation fails

Answer: A

Question No : 2

Given:

```
public class Case {
    public static void main(String[] args) {
        String product = "Pen";
        product.toLowerCase();
        product.concat(" BOX".toLowerCase());
        System.out.print(product.substring(4,6));
    }
}
```

What is the result?

A. box
B. nbo
C. bo
D. nb
E. An exception is thrown at runtime

Answer: E

Question No : 3

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 created

Answer: A,C,E

Question No : 4

Given the code fragment:

```
public class Person {
    String name;
    int age = 25;

    public Person(String name) {
        this();                                    //line n1
        setName(name);
    }

    public Person(String name, int age) {
        Person(name);                              //line n2
        setAge(age);
    }

    //setter and getter methods go here

    public String show() {
        return name + " " + age + " " + number ;
    }
    public static void main(String[] args) {
        Person p1 = new Person("Jesse");
        Person p2 = new Person("Walter",52);
        System.out.println(p1.show());
        System.out.println(p2.show());
    }
}
```

What is the result?

A. Jesse 25
Walter 52
B. Compilation fails only at line n1
C. Compilation fails only at line n2
D. Compilation fails at both line n1 and line n2

Answer: D

Question No : 5

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);

Answer: A,D

Explanation:

Incorrect:

Not B: circular definition of n3.

Not C: Compilation error. line Calc c = new Mid();

required: Calc

found: Mid

Not E: Compilation error. line int n3 = Calc.findMid(n1, n2);

non-static method findMid(int,int) cannot be referenced from a static context

Question No : 6

Given the code fragment:

```
LocalDate date1 = LocalDate.now();
LocalDate date2 = LocalDate.of(2014, 6, 20);
LocalDate date3 = LocalDate.parse("2014-06-20", DateTimeFormatter.ISO_DATE);
System.out.println("date1 = " + date1);
System.out.println("date2 = " + date2);
System.out.println("date3 = " + date3);
```

Assume that the system date is June 20, 2014. What is the result?

A)
```
date1 = 2014-06-20
date2 = 2014-06-20
date3 = 2014-06-20
```

B)
```
date1 = 06/20/2014
date2 = 2014-06-20
date3 = Jun 20, 2014
```

C) Compilation fails.

D) A `DateParseExcpetion` is thrown at runtime.

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

Answer: A

Question No : 7

Given the code fragment:

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

What is the result?

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

Answer: A

Question No : 8

Given the code fragment:

```
public static void main(String[] args) {
    String date = LocalDate
                  .parse("2014-05-04")
                  .format(DateTimeFormatter.ISO_DATE_TIME);
    System.out.println(date);
}
```

What is the result?

A. May 04, 2014T00:00:00.000
B. 2014-05-04T00:00: 00. 000
C. 5/4/14T00:00:00.000
D. An exception is thrown at runtime.

Answer: D

Explanation:

java.time.temporal.UnsupportedTemporalTypeException: Unsupported field: HourOfDay

Question No : 9

Given the code fragment:

```
public static void main(String[] args) {
    double discount = 0;
    int qty = Integer.parseInt(args[0]);
    //line n1;
}
```

And given the requirements:

- If the value of the qty variable is greater than or equal to 90, discount = 0.5
- If the value of the qty variable is between 80 and 90, discount = 0.2

Which two code fragments can be independently placed at line n1 to meet the requirements?

☐ A)
```
if (qty >= 90) { discount = 0.5; }
if (qty > 80 && qty < 90) { discount = 0.2; }
```

☐ B)
```
discount = (qty >= 90) ? 0.5 : 0;
discount = (qty > 80) ? 0.2 : 0;
```

☐ C)
```
discount = (qty >= 90) ? 0.5 : (qty > 80)? 0.2 : 0;
```

☐ D)
```
if (qty > 80 && qty < 90) {
    discount = 0.2;
} else {
    discount = 0;
}
if (qty >= 90) {
    discount = 0.5;
} else {
    discount = 0;
}
```

☐ E)
```
discount = (qty > 80) ? 0.2 : (qty >= 90) ? 0.5 : 0;
```

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

Answer: A,C

Question No : 10

Given:

```
public class Circle {
    double radius;
    public double area;
    public Circle(double r) { radius = r; }
    public double getRadius() { return radius; }
    public void setRadius(double r) { radius = r; }
    public double getArea() { return /* ??? */; }
}

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

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 Match.PI * radius * radius; }
C. Add the following method:
public double getArea ( ) {area = Match.PI * radius * radius; }
D. Change the cacess modifier of the SerRadius ( ) method to be protected.

Answer: B,D

Question No : 11

Given the code fragment:

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

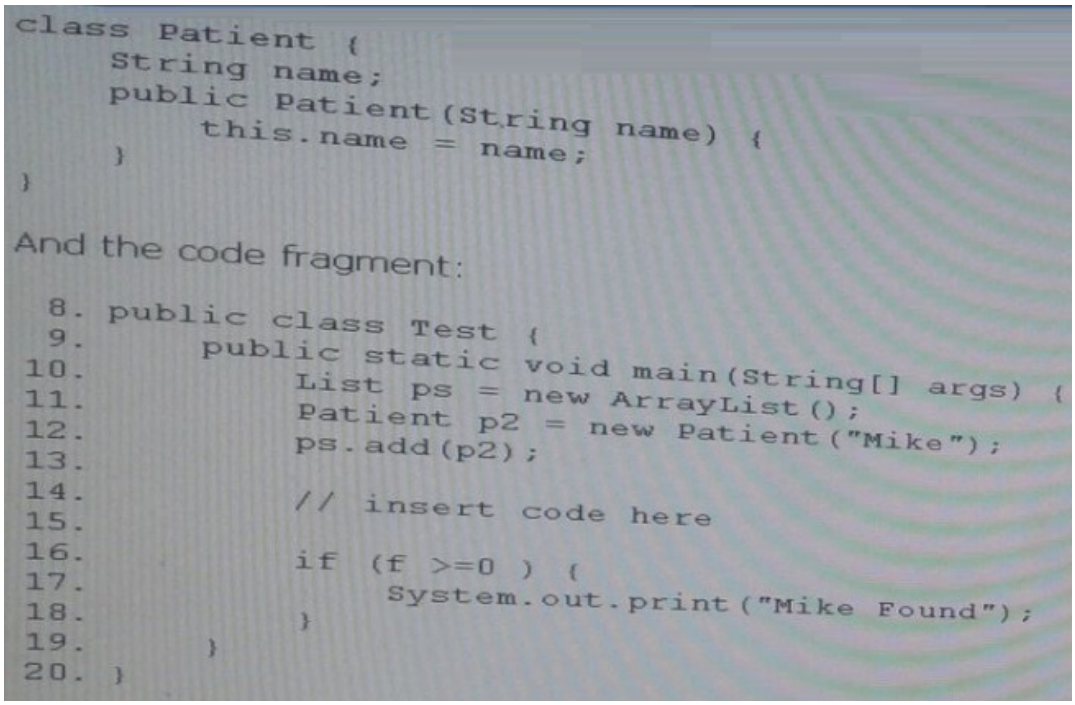
What is the result?

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

Answer: B

Question No : 12

Given:



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

And the code fragment:

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

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

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

indexOf表示某个对象出现的位置，new一个新的对象是找不到的。

Answer: C

Question No : 13

Given:

```
class CD {
    int r;
    CD(int r){
        this.r=r;
    }
}

class DVD extends CD {
    int c;
    DVD(int r, int c) {
        // line n1
    }
}
```

And given the code fragment:

```
DVD dvd = new DVD(10,20);
```

Which code fragment should you use at line n1 to instantiate the dvd object successfully?

A)
```
super.r = r;
this.c = c;
```

B)
```
super(r);
this(c);
```

C)
```
super(r);
this.c = c;
```

D)
```
this.c = r;
super(c);
```

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

Answer: C

Question No : 14

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. The expression expr1 is optional. it initializes the loop and is evaluated once, as the loop begin.
C. When expr2 evaluates to false, the loop terminates. It is evaluated only after each iteration through the loop.
D. The expression expr3 must be present. It is evaluated after each iteration through the loop.

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

Question No : 15

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

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

Answer: A,B,C

Explanation: A: Private constructors prevent a class from being explicitly instantiated by its callers.
If the programmer does not provide a constructor for a class, then the system will always provide a default, public no-argument constructor. To disable this default constructor, simply add a private no-argument constructor to the class. This private constructor may be empty.
B: The following works fine:

```
int cake() {
int cake=0;
return (1);
}
```

C: We can overload static method in Java. In terms of method overloading static method are just like normal methods and in order to overload static method you need to provide another static method with same name but different method signature.

Incorrect:
Not D: Only a public class in an application need to have a main method.
Not E:
Example:

```
class A
{
public string something;
public int a;
```

}
Q: What do you call classes without methods?
Most of the time: An anti pattern.

Why? Because it faciliates procedural programming with "Operator" classes and data structures. You separate data and behaviour which isn't exactly good OOP.

Often times: A DTO (Data Transfer Object)

Read only datastructures meant to exchange data, derived from a business/domain object.

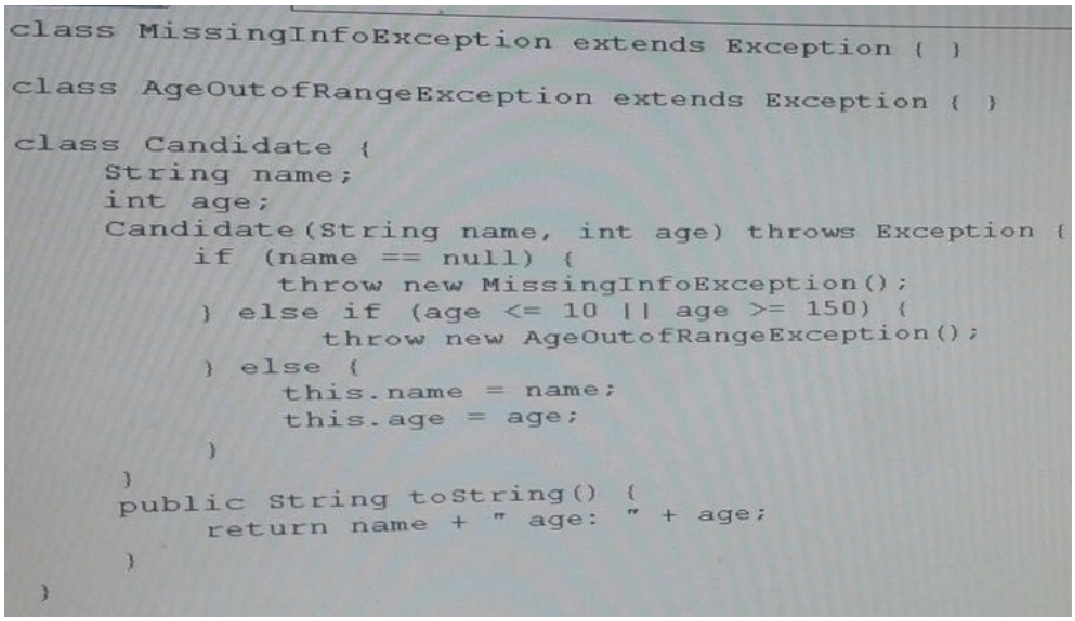
Sometimes: Just data structure.

Well sometimes, you just gotta have those structures to hold data that is just plain and simple and has no operations on it.

Not F: Fields need to be initialtized. If not the code will not compile.
Example:
Uncompilable source code - variable x might not have been initialized

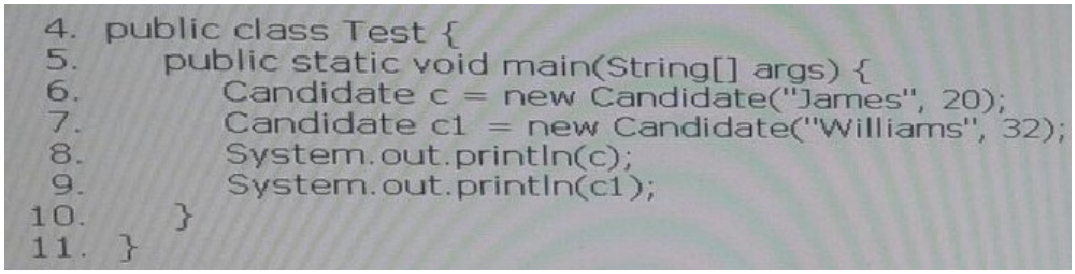
Question No : 16

View the exhibit.



```
class MissingInfoException extends Exception { }

class AgeOutofRangeException extends Exception { }

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

Given the code fragment:



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

Which change enables the code to print the following?

James age: 20

Williams age: 32

A. Replacing line 5 with public static void main (String [] args) throws MissingInfoException, AgeOutofRangeException {
B. Replacing line 5 with public static void main (String [] args) throws.Exception {
C. Enclosing line 6 and line 7 within a try block and adding:
catch(Exception e1) { //code goes here}
catch (missingInfoException e2) { //code goes here}
catch (AgeOutofRangeException e3) {//code goes here}
D. Enclosing line 6 and line 7 within a try block and adding:
catch (missingInfoException e2) { //code goes here}
catch (AgeOutofRangeException e3) {//code goes here}

Answer: C

Question No : 17

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

Answer: B

Question No : 18

Given:

```
public class Test {

public static void main(String[] args) {

int day = 1;

switch (day) {

case "7": System.out.print("Uranus");

case "6": System.out.print("Saturn");

case "1": System.out.print("Mercury");

case "2": System.out.print("Venus");

case "3": System.out.print("Earth");

case "4": System.out.print("Mars");

case "5": System.out.print("Jupiter");

}

}

}
```

Which two modifications, made independently, enable the code to compile and run?

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

Answer: A,C

Explanation: The following will work fine:

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

Question No : 19

Given:

```
public class Product {
     int id;
     String name;
     public Product(int id, String name) {
        this.id = id;
        this.name = name;
     }
}
```

And given the code fragment:

```
4. Product p1 = new Product(101, "Pen");
5. Product p2 = new Product(101, "Pen");
6. Product p3 = p1;
7. boolean ans1 = p1 == p2;
8. boolean ans2 = p1.name.equals(p2.name);
9. System.out.print(ans1 + ":" + ans2);
```

What is the result?

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

Answer: C

Question No : 20

Given:

```
class Alpha {
    int ns;
    static int s;
    Alpha(int ns) {
        if (s < ns) {
            s = ns;
            this.ns = ns;
        }
    }
    void doPrint() {
        System.out.println("ns = " + ns + " s = " + s);
    }
}
```

And,

```
public class TestA {
    public static void main(String[] args) {
        Alpha ref1 = new Alpha(50);
        Alpha ref2 = new Alpha(125);
        Alpha ref3 = new Alpha(100);
        ref1.doPrint();
        ref2.doPrint();
        ref3.doPrint();
    }
}
```

What is the result?

- A)
  ```
  ns = 50 s = 125
  ns = 125 s = 125
  ns = 100 s = 125
  ```
- B)
  ```
  ns = 50 s = 125
  ns = 125 s = 125
  ns = 0 s = 125
  ```
- C)
  ```
  ns = 50 s = 50
  ns = 125 s = 125
  ns = 100 s = 100
  ```
- D)
  ```
  ns = 50 s =  50
  ns = 125 s = 125
  ns = 0 s = 125
  ```

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

Answer: B

Question No : 21

Given:

```
1. import java.io.Error;
2.      public class TestApp {
3.      public static void main(String[] args) {
4.          TestApp t = new TestApp();
5.          try {
6.              t.doPrint();
7.              t.doList();
8.
9.          } catch (Exception e2) {
10.             System.out.println("Caught " + e2);
11.         }
12.     }
13.     public void doList() throws Exception {
14.         throw new Error("Error");
15.     }
16.     public void doPrint() throws Exception {
17.         throw new RuntimeException("Exception");
18.     }
19. }
```

What is the result?

```
A) Caught java.lang.RuntimeException: Exception
   Exception in thread "main" java.lang.Error: Error
       at TestApp.doList(TestApp.java: 14)
       at TestApp.main(TestApp.java: 6)

B) Exception in thread "main" java.lang.Error: Error
       at TestApp.doList(TestApp.java: 14)
       at TestApp.main(TestApp.java: 6)

C) Caught java.lang.RuntimeException: Exception
   Caught java.lang.Error: Error

D) Caught java.lang.RuntimeException: Exception
```

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

Answer: C

Question No : 22

Given:

```
int x = 10;
if (x > 10) {
        System.out.println(">");
} else if (x < 10) {
        System.out.println("<");
} else {
        System.out.println("=");
}
```

Which of the following is equivalent to the above code fragment?

A. System.out.printLn(x>10?">,': "<":,'=");
B. System.out.println(x>10? ">"?"<":"=");
C. System.out.println(x>10?">":x<10?"<":"=");
D. System.out.printLn(x>10?">"?,'<"?"=");
E. None of the above

Answer: B

Explanation:

Option A is incorrect as we can't use abstract with non abstract method, (here method has method body.)

Option C is incorrect as when overriding method we can't use more restrictive access

modifier, so trying to use private to override default access Level method causes a compile time error.
Option D is incorrect as default methods (not methods with default access level) are allowed only in interfaces.
Option E is incorrect as method all ready has void as return type, so we can't add int there.
Option B is correct as we can use final there, since the method is non abstract
https://docs.oracle.com/javase/tutorial/java/landl/polymorphism.html

Question No : 23

Given:

```
public class Test {
public static void main(String[] args) {
try {
String[] arr =new String[4];
arr[1] = "Unix";
arr[2] = "Linux";
arr[3] = "Solarios";
for (String var : arr) {
System.out.print(var + " ");
}
} catch(Exception e) {
System.out.print (e.getClass());
}
}
}
```

What is the result?

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

Answer: B

Explanation: null Unix Linux Solarios

The first element, arr[0], has not been defined.

Question No : 24

Given:

Given:

```
public class SuperTest {
public static void main(String[] args) {
statement1
statement2
statement3
}
}
class Shape {
public Shape() {
System.out.println("Shape: constructor");
}
public void foo() {
System.out.println("Shape: foo");
```

```
}

}

class Square extends Shape {

public Square() {

super();

}

public Square(String label) {

System.out.println("Square: constructor");

}

public void foo() {

super.foo();

}

public void foo(String label) {

System.out.println("Square: foo");

}

}

}

}
```

What should statement1, statement2, and statement3, be respectively, in order to produce the result?

Shape: constructor

Square: foo

Shape: foo

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

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

Answer: F

Question No : 25

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 > 0 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 ;

Answer: A

Explanation:

Note: Code in B (code2 is --index;). also works fine.

Question No : 26

Given:

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

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

What is the result?

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

D. Compilation fails
E. The loop executes infinite times

Answer: E

Question No : 27

Which two are Java Exception classes?

A. SercurityException
B. DuplicatePathException
C. IllegalArgumentException
D. TooManyArgumentsException

Answer: A,C

Question No : 28

Given:

```
public static void main(String[] args) {
    String ta = "A ";
    ta = ta.concat("B ");
    String tb = "C ";
    ta = ta.concat(tb);
    ta.replace('C', 'D');
    ta = ta.concat(tb);
    System.out.println(ta);
}
```

What is the result?

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

Answer: C

Question No : 29

Given the code fragment:

```
1. class X {
2.     public void printFileContent() {
3.         /* code goes here */
4.         throw new IOException();
5.     }
6. }
7. public class Test {
8.     public static void main(String[] args) {
9.         X xobj = new X();
10.        xobj.printFileContent();
11.    }
12. }
```

Which two modifications should you make so that the code compiles successfully?

☐ A) Replace line 8 with `public static void main(String[] args) throws Exception {`

☐ B) Replace line 10 with:

```
try {
    xobj.printFileContent();
}
catch(Exception e) { }
catch(IOException e) { }
```

☐ C) Replace line 2 with `public void printFileContent() throws IOException {`

☐ D) Replace line 4 with `throw IOException("Exception raised");`

☐ E) At line 11, insert `throw new IOException();`

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

Answer: A,C

Explanation:

Add throws clause in both printFileContent and main.

Question No : 30

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

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

Answer: A

Question No : 31

Which two actions will improve the encapsulation of a class?

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

Answer: A,D
Reference:

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

Question No : 32

Given:

public class ComputeSum {

```
public int x;

public int y;

public int sum;

public ComputeSum (int nx, int ny) {

x = nx; y =ny;

updateSum();

}

public void setX(int nx) { x = nx; updateSum();}

public void setY(int ny) { x = ny; updateSum();}

void updateSum() { sum = x + y;}

}
```

This class needs to protect an invariant on the sum field.

Which three members must have the private access modifier to ensure that this invariant is maintained?

A. The x field
B. The y field
C. The sum field
D. The ComputerSum ( ) constructor
E. The setX ( ) method
F. The setY ( ) method

Answer: C,E,F

Explanation: The sum field and the two methods (setX and SetY) that updates the sum field.

Question No : 33

Given the following array:

```
int[] intArr = {8, 16, 32, 64, 128};
```

Which two code fragments, independently, print each element in this array?

☐ A)
```
for (int i : intArr) {
    System.out.print(intArr[i] +" ");
}
```

☐ B)
```
for (int i : intArr) {
    System.out.print(i +" ");
}
```

☐ C)
```
for (int i=0 : intArr) {
    System.out.print(intArr[i] +" ");
    i++;
}
```

☐ D)
```
for (int i=0; i < intArr.length; i++) {
    System.out.print(i +" ");
}
```

☐ E)
```
for (int i=0; i < intArr.length; i++) {
    System.out.print(intArr[i] +" ");
}
```

☐ F)
```
for (int i; i < intArr.length; i++) {
    System.out.print(intArr[i] +" ");
}
```

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

Answer: B,E

Explanation: All the remaining options have syntax errors

Question No : 34

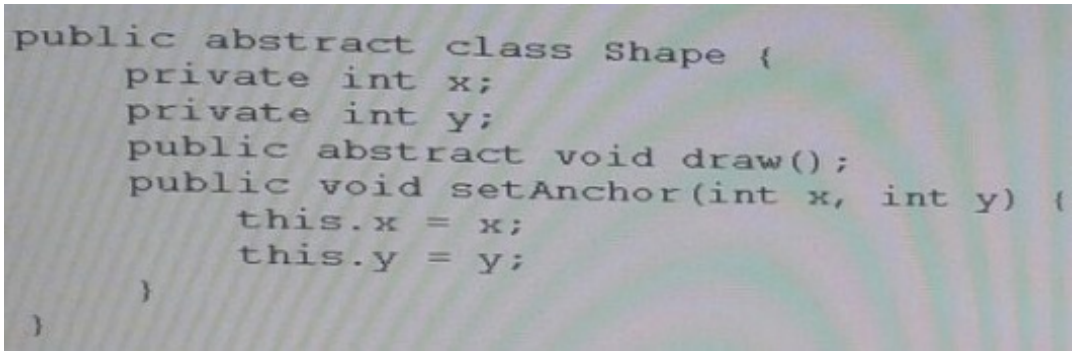
Which statement best describes encapsulation?

A. Encapsulation ensures that classes can be designed so that only certain fields and methods of an object are accessible from other objects.
B. Encapsulation ensures that classes can be designed so that their methods are inheritable.
C. Encapsulation ensures that classes can be designed with some fields and methods declared as abstract.
D. Encapsulation ensures that classes can be designed so that if a method has an argument MyType x, any subclass of MyType can be passed to that method.

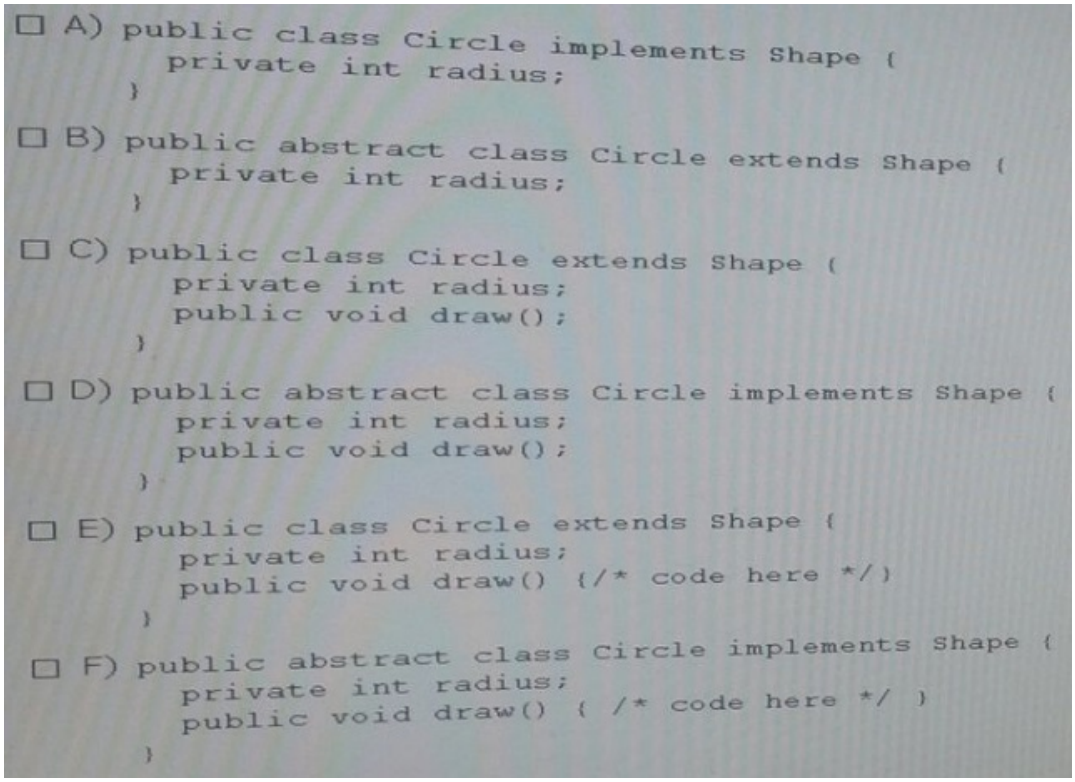
Answer: A

Question No : 35

Given:



Which two classes use the shape class correctly?



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

Answer: B,E

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

Question No : 36

Given:

```
public class Test {

    public static void main(String[] args) {

        String[][] chs = new String[2][];
        chs[0] = new String[2];
        chs[1] = new String[5];
        int i = 97;

        for (int a = 0; a < chs.length; a++) {
            for (int b = 0; b < chs.length; b++) {
                chs[a][b] = "" + i;
                i++;
            }
        }

        for (String[] ca : chs) {
            for (String c : ca) {
                System.out.print(c + " ");
            }
            System.out.println();
        }
    }
}
```

What is the result?

A. 97 98
99 100 null null null
B. 91 98
99 100 101 102 103
C. Compilation rails.
D. A NullPointerException is thrown at runtime.
E. An ArraylndexOutOfBoundsException is thrown at runtime.

Answer: A

Question No : 37

Given the code fragment:

```
int nums1[] = new int[3];
int nums2[] = {1, 2, 3, 4, 5};
nums1 = nums2;
for (int x : nums1){
    System.out.print(x + ":");
}
```

What is the result?

A. 1:2:3:4:5:
B. 1:2:3:
C. Compilation fails.
D. An ArrayoutofBoundsException is thrown at runtime.

Answer: A

Question No : 38

Given:

public class Test1 {

static void doubling (Integer ref, int pv) {

ref =20;

pv = 20;

}

public static void main(String[] args) {

Integer iObj = new Integer(10);

int iVar = 10;

doubling(iObj++, iVar++);

System.out.println(iObj+ ", "+iVar);

What is the result?

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

Answer: A

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

Question No : 39

Given:

```
public class FieldInit {

char c;

boolean b;

float f;

void printAll() {

System.out.println("c = " + c);

System.out.println("c = " + b);

System.out.println("c = " + f);

}


public static void main(String[] args) {

FieldInit f = new FieldInit();

f.printAll();

}

}
```

What is the result?

A. c = null
b = false
f = 0.0F
B. c = 0
b = false
f = 0.0f
C. c = null
b = true
f = 0.0
D. c =
b = false
f = 0.0

Answer: D

Question No : 40

Given the code fragment:

```
String shirts[][] = new String[2][2];
shirts[0][0] = "red";
shirts[0][1] = "blue";
shirts[1][0] = "small";
shirts[1][1] = "medium";
```

Which code fragment prints red: blue: small: medium?

A)
```
for (int index = 1; index < 2; index++) {
    for (int idx = 1; idx < 2; idx++) {
        System.out.print(shirts[index][idx] + ":");
    }
}
```

B)
```
for (int index = 0; index < 2; ++index) {
    for (int idx = 0; idx < index; ++idx) {
        System.out.print(shirts[index][idx] + ":");
    }
}
```

C)
```
for (String c : colors) {
    for (String s : sizes) {
        System.out.println(s + ":");
    }
}
```

D)
```
for (int index = 0; index < 2;) {
    for (int idx = 0; idx < 2;) {
        System.out.print(shirts[index][idx] + ":");
        idx++;
    }
    index++;
}
```

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

Answer: D

Question No : 41

Given:

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

What is the result?

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

Answer: A

Question No : 42

Given the class definitions:

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

And the code fragment of the main() method,

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

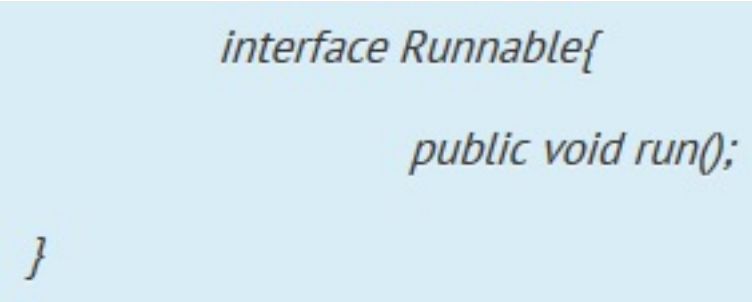
What is the result?

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

Answer: D

Question No : 43

Consider following interface.



Which of the following will create instance of Runnable type?

A. Runnable run = 0 -> {System.out.println("Run");}
B. Runnable run = 0 -> System.outprintlnfRun");
C. Runnable run = 0 > System.outprintlnfRun");
D. Runnable run = > System.ouLprintlnfRun"};
E. None of the above.

Answer: A

Explanation:

Option A is the correct answer.

To create we have used following method with LocalDate class;

public static LocalDate of(intyear, int month, intdayOfMonth)

Here we need to remember that month is not zero based so if you pass 1 for month, then month will be January.

Then we have used period object of 1 day and add to date object which makes current date to next day, so final output is 2015-03-27. Hence option A is correct.

Question No : 44

Which of the following can fill in the blank in this code to make it compile? (Select 2 options.)

```
1.  public void method() ____ Exception {
2.      _____ Exception();
3.  }
```

A. On line 1, fill in throws
B. On line 1, fill in throws new
C. On line 2, fill in throw new
D. On line 2, fill in throws
E. On line 2, fill in throws new

Answer: A,C

Explanation:

Option A and C are the correct answer.

In a method declaration, the keyword throws is used. So here at line 1 we have to use option A.

To actually throw an exception, the keyword throw is used and a new exception is created, so at line 2 we have to use throw and new keywords, which is option C. Finally it will look like;

public void method() throws Exception {

throw new Exception0;

}

REFERENCE : httpsy/docs.oracle.com/javase/tutorial/essential/io/fileOps.html#exception

The correct answer is: On line 1, fill in throws. On line 2, fill in throw new

Question No : 45

Given the code fragment:

```
public class Test{

    void readCard(int cardNo) throws Exception {
        System.out.println("Reading Card");
    }

    void checkCard(int cardNo) throws RuntimeException { // line n1
        System.out.println("Checking Card");
    }

    public static void main(String[] args) {
        Test ex = new Test();
        int cardNo = 12344;
        ex.checkCard(cardNo);                //line n2
        ex.readCard(cardNo);                 //line n3
    }
}
```

运行时错误不用在调用的时候抛出异常。所以只有 linen3会编译失败

What is the result?

A. Reading Card
Checking Card
B. Compilation fails only at line n1.
C. Compilation fails only at line n2.
D. Compilation fails only at line n3.
E. Compilation fails at both line n2 and line n3.

Answer: D

Question No : 46

Given:

```
public class Test {

    public static void main(String[] args) {
        if (args[0].equals("Hello") ? false : true) {
            System.out.println("Success");
        } else {
            System.out.println("Failure");
        }
    }
}
```

And given the commands:

javac Test.Java

Java Test Hello

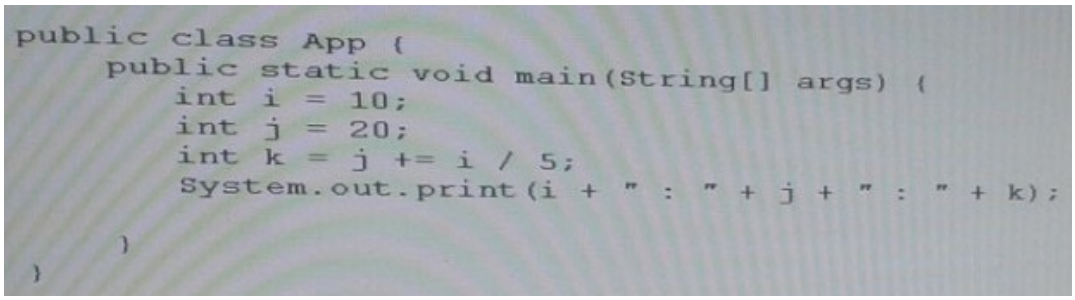
What is the result?

A. Success
B. Failure
C. Compilation fails.
D. An exception is thrown at runtime

Answer: B

Question No : 47

Given:



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

What is the result?

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

Answer: B

Question No : 48

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

Answer: A

Explanation: The java.lang.StringBuilder.insert(int offset, char c) method inserts the string representation of the char argument into this sequence.
The second argument is inserted into the contents of this sequence at the position indicated by offset. The length of this sequence increases by one.The offset argument must be greater than or equal to 0, and less than or equal to the length of this sequence.

Reference: Java.lang.StringBuilder.insert() Method

Question No : 49

Given the code fragment:

```
public static void main(String[] args) {
    String str = " ";
    str.trim();
    System.out.println(str.equals("") + " " + str.isEmpty());
}
```

What is the result?

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

Answer: C

Question No : 50

Given:

```
public class Msg {
   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.print(doMsg(z));
   }
}
```

经过实测这么写是可以通过编译的。

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

~~**Answer: E**~~

Question No : 51

Given:

Base.java:

```
class Base {
    public void test(){
        System.out.println("Base ");
    }
}
```

DerivedA.java:

```
class DerivedA extends Base {
    public void test(){
        System.out.println("DerivedA ");
    }
}
```

DerivedB.java:

```
class DerivedB  extends DerivedA {
    public void test(){
        System.out.println("DerivedB ");
    }
    public static void main(String[] args) {
        Base b1 = new DerivedB();
        Base b2 = new DerivedA();
        Base b3 = new DerivedB();
        b1 = (Base) b3;
        Base b4 = (DerivedA) b3;
        b1.test();
        b4.test();
    }
}
```

What is the result?

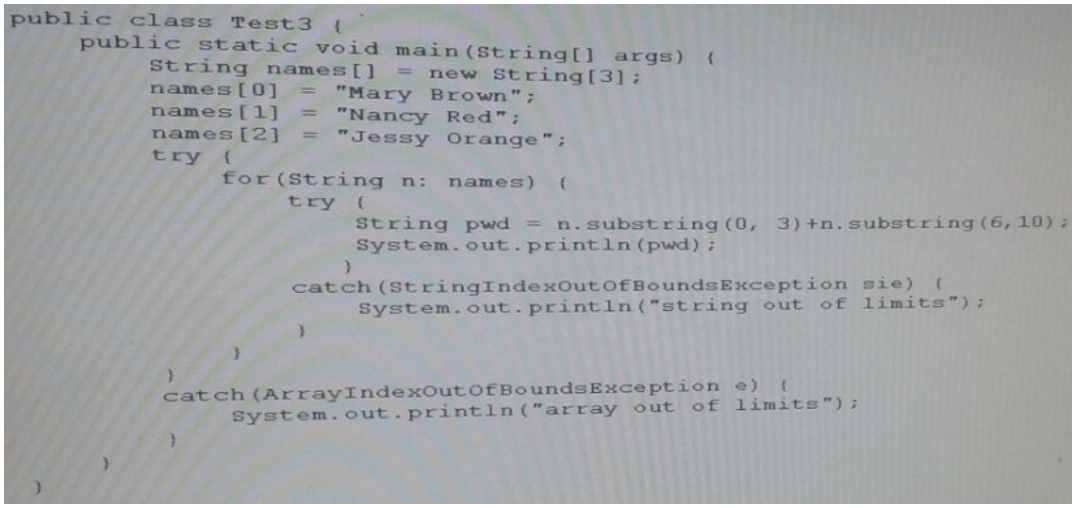
A. Base
DerivedA
B. Base
DerivedB
C. DerivedB
DerivedB

D. DerivedB
DerivedA
E. A classcast Except ion is thrown at runtime.

Answer: C

Question No : 52

Given:



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

What is the result?

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

Answer: A

Question No : 53

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

Answer: C

Explanation: class java.lang.StringBuilder
class java.lang.String
class java.util.ArrayList

Question No : 54

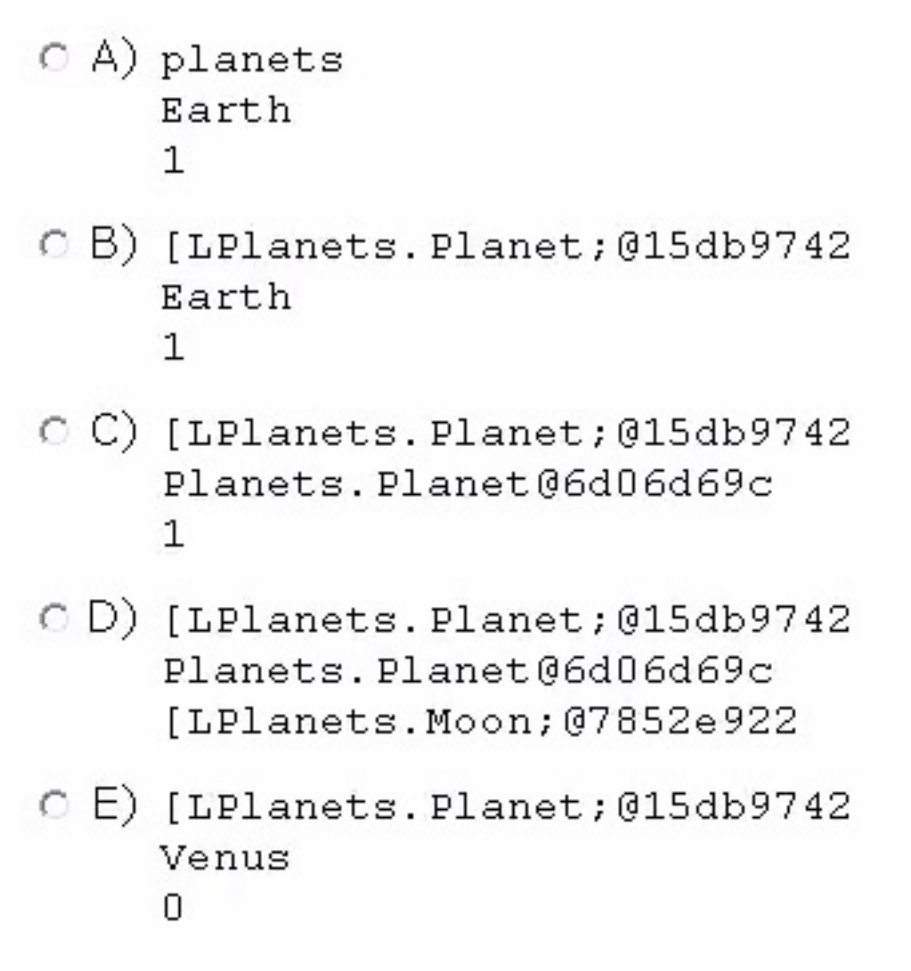
Given the following code for a Planet object:

```
public class Planet {
    public String name;
    public int moons;

    public Planet(String name, int moons) {
        this.name = name;
        this.moons = moons;
    }
}
```

And the following main method:

```
public static void main(String[] args){
    Planet[] planets = {
        new Planet("Mercury", 0),
        new Planet("Venus", 0),
        new Planet("Earth", 1),
        new Planet("Mars", 2)
    };

    System.out.println(planets);
    System.out.println(planets[2]);
    System.out.println(planets[2].moons);
}
```

What is the output?



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

Answer: C

Question No : 55

Given:

public class String1 {

public static void main(String[] args) {

String s = "123";

if (s.length() >2)

s.concat("456");

for(int x = 0; x <3; x++)

s += "x";

System.out.println(s);

}

}

What is the result?

A. 123
B. 123xxx
C. 123456
D. 123456xxx
E. Compilation fails

Answer: B

Explanation: 123xxx

The if clause is not applied.

Note: Syntax of if-statement:
if (Statement) {
}

Question No : 56

Given the code fragment:

```
public static void main(String[] args) {
    int ii = 0;
    int jj = 7;
    for (ii = 0; ii < jj - 1; ii = ii + 2) {
         System.out.print(ii + " ");
     }
}
```

What is the result?

A. 2 4
B. 0 2 4 6
C. 0 2 4
D. Compilation fails

Answer: C

Question No : 57

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

Answer: B

Question No : 58

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. Hello Universe!
C. Hello World!
D. Compilation fails.

Answer: A

Question No : 59

Given:

Given:

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

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

A. Ym
Xm2
B. Ym
Xm1
C. Compilation fails
D. A ClassCastException is thrown at runtime

Answer: B

Question No : 60

public class ForTest {

public static void main(String[] args) {

int[] arrar = {1,2,3};

for (foo) {

}

}

}

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

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

Answer: A,B,C

Question No : 61

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. private static void doPrint() {
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++;
}

Answer: B

Explanation:

Incorrect:

not A: The following line causes a compile error:

System.out.println(str[j]);

Not C: Compile erro line:

for (String[] sub: arr[][])

not D: Output: C

Question No : 62

Given:

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

返回指定索引处的字符。theString的最大索引只有10，所以会报下标越界的错误

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.

Answer: C

Question No : 63

Which statement is true about Java byte code?

A. It can run on any platform.
B. It can run on any platform only if it was compiled for that platform.
C. It can run on any platform that has the Java Runtime Environment.
D. It can run on any platform that has a Java compiler.
E. It can run on any platform only if that platform has both the Java Runtime Environment and a Java compiler.

Answer: C

Question No : 64

Given:

```
1. import java.util.ArrayList;
2. import java.util.List;
3.
4. public class Whizlabs{
5.
6.          public static void main(String[] args){
7.                    List<int> list = new ArrayList<>();
8.                    list.add(21); list.add(13);
9.                    list.add(30); list.add(11);
10.                   list.removeIf(e -> e%2 != 0);
11.                   System.out.println(list);
12.         }
13. }
```

What is the output?

A. [21, 13, 11]
B. [30]
C. []
D. Compilation fails due to error at line 7
E. Compilation tails due to error at line 10

Answer: D

Explanation:

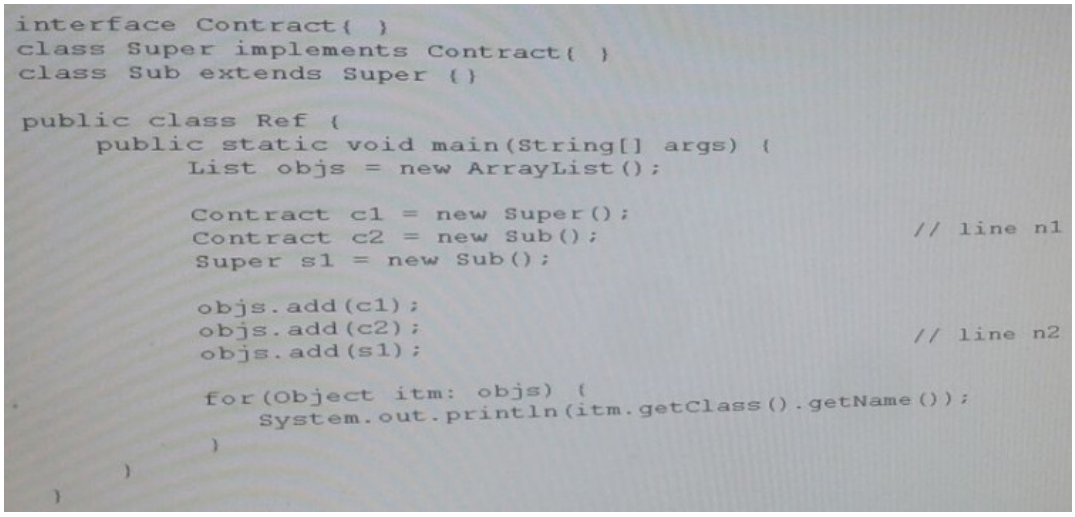
Option D is the correct answer.

Code fails to compile as we can't use primitive for collections type, so in this code trying to use int at line 7, causes a compile error. We should have use wrapper. Integer there. So option D is correct.

https://docs.oracle.eom/javase/8/docs/api/java/util/ArrayList.html

Question No : 65

Given the code fragments:



```
interface Contract{ }
class Super implements Contract{ }
class Sub extends Super {}

public class Ref {
    public static void main(String[] args) {
        List objs = new ArrayList();

        Contract c1 = new Super();
        Contract c2 = new Sub();                    // line n1
        Super s1 = new Sub();

        objs.add(c1);
        objs.add(c2);                               // line n2
        objs.add(s1);

        for(Object itm: objs) {
            System.out.println(itm.getClass().getName());
        }
    }
}
```

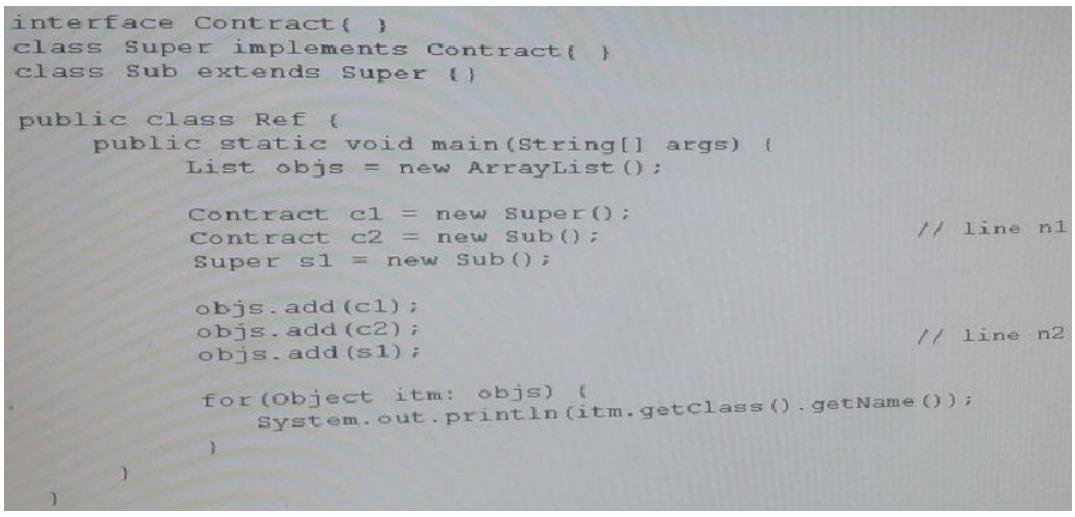
What is the result?

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

Answer: D

Question No : 66

Given the code fragment:



```
interface Contract{ }
class Super implements Contract{ }
class Sub extends Super {}

public class Ref {
    public static void main(String[] args) {
        List objs = new ArrayList();

        Contract c1 = new Super();
        Contract c2 = new Sub();                    // line n1
        Super s1 = new Sub();

        objs.add(c1);
        objs.add(c2);                               // line n2
        objs.add(s1);

        for(Object itm: objs) {
            System.out.println(itm.getclass().getName());
        }
    }
}
```

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

Answer: D

Question No : 67

Given the following two classes:

```
public class Customer {
    ElectricAccount acct = new ElectricAccount();

    public void useElectricity(double kWh){
        acct.addKWh(kWh);
    }
}

public class ElectricAccount {
    private double kWh;
    private double rate = 0.07;
    private double bill;

    //line n1
}
```

How should you write methods in the ElectricAccount class at line n1 so that the member variable bill is always equal to the value of the member variable kwh multiplied by the member variable rate?

Any amount of electricity used by a customer (represented by an instance of the customer class) must contribute to the customer's bill (represented by the member variable bill) through the method useElectricity method. An instance of the customer class should never be able to tamper with or decrease the value of the member variable bill.

A)
```
public void addKWh(double kWh) {
    this.kWh += kWh;
    this.bill = this.kWh*this.rate;
}
```

B)
```
public void addKWh(double kWh) {
    if (kWh > 0){
        this.kWh += kWh;
        this.bill = this.kWh * this.rate;
    }
}
```

C)
```
private void addKWh(double kWh) {
    if (kWh > 0) {
        this.kWh += kWh;
        this.bill = this.kWh*this.rate;
    }
}
```

D)
```
public void addKWh(double kWh) {
    if(kWh > 0) {
        this.kWh += kWh;
        setBill(this.kWh);
    }
}
public void setBill(double kWh) {
    bill = kWh*rate;
}
```

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

Answer: B

Question No : 68

Given:

```
public class TestScope {
    public static void main(String[] args) {
        int var1 = 200;
        System.out.print(doCalc(var1));
        System.out.print(" "+var1);
    }
    static int doCalc(int var1){
        var1 = var1 * 2;
        return var1;
    }
}
```

What is the result?

A. 400 200
B. 200 200
C. 400 400
D. Compilation fails.

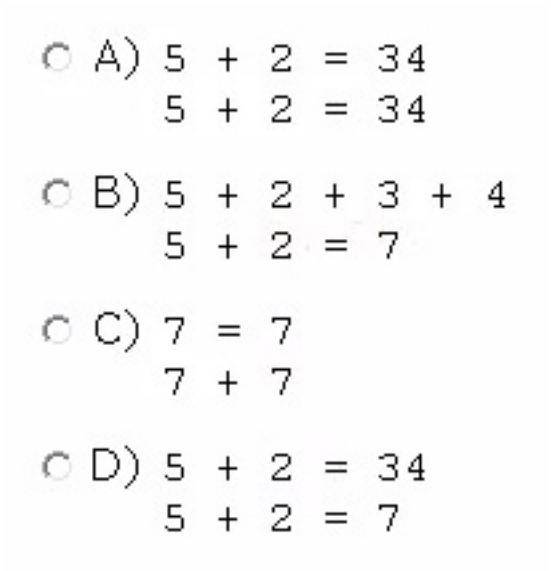
Answer: A

Question No : 69

Given:

```
System.out.println("5 + 2 = " + 3 + 4);
System.out.println("5 + 2 = " + (3 + 4));
```

What is the result?



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

Answer: D

Question No : 70

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

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

Answer: D
Reference:

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

Question No : 71

Given:

```
1.  import java.time.LocalDate;
2.  import java.time.Period;
3.
4.  public class Whizlabs {
5.          public static void main(String[] args) {
6.                  LocalDate date = LocalDate.of(2015, 3, 26);
7.                  Period p = Period.ofDays(1);
8.                  System.out.println(date.plus(p));
9.          }
10. }
```

What is the output?

A. 2015-03-27
B. 2015-04-27
C. 2015-02-27
D. Compilation fails due to error at line 6.
E. Compilation fails due to error at line 8.

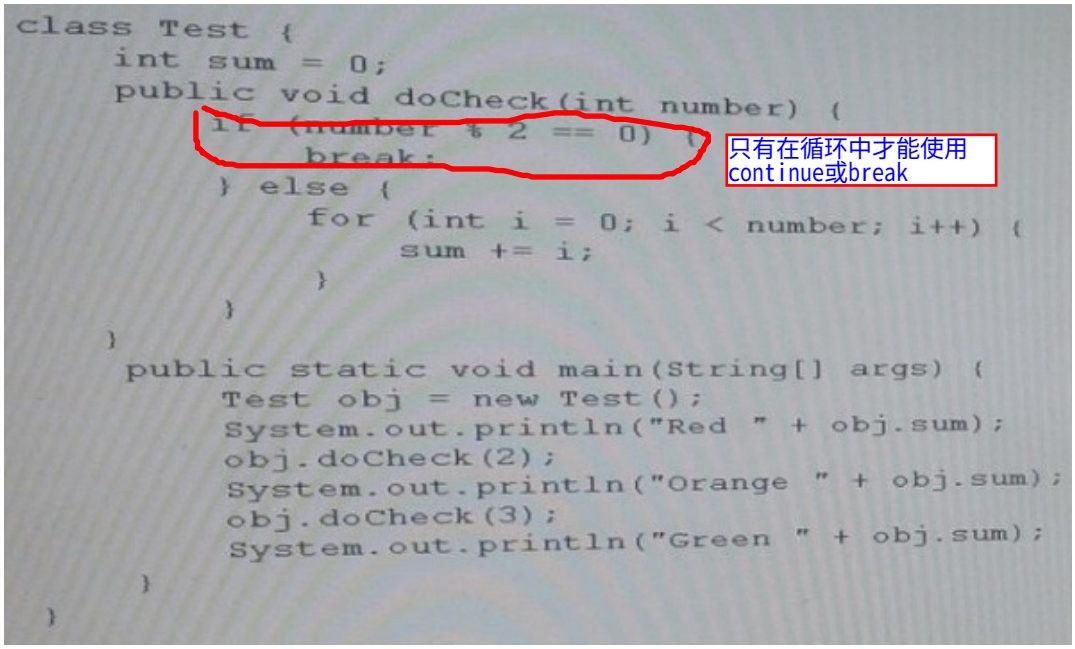
Answer: A

Explanation:

To create we have used following method with LocalDate class;
public static LocalDate of(intyear, int month, int dayOfMonth)
Here we need to remember that month is not zero based so if you pass 1 for month, then month will be January.
Then we have used period object of 1 day and add to date object which makes current date to next day, so final output is 2015-03-27. Hence option A is correct.
https://docs.oracle.com/javase/tutorial/datetime/iso/datetime.html

Question No : 72

Given:



What is the result?

A. Red 0
Orange 0
Green 3
B. Red 0
Orange 0
Green 6
C. Red 0
Orange 1
D. Green 4
E. Compilation fails

Answer: E

Question No : 73

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.

Answer: D

Explanation: In both Java and C#, a "default constructor" refers to a nullary constructor that is automatically generated by the compiler if no constructors have been defined for the class. The default constructor is also empty, meaning that it does nothing. A programmer-defined constructor that takes no parameters is also called a default constructor.

Question No : 74

Given:

```
class Vehicle {
    int x;
    Vehicle(){
        this(10);  // line n1
    }
    Vehicle(int x) {
        this.x = x;
    }
}

class Car extends Vehicle {
    int y;
    Car() {
        super();
        this(20);   // line n2
    }
    Car(int y) {
       this.y = y;
    }
    public String toString() {
        return super.x + ":" + this.y;
    }
}
```

And given the code fragment:

And given the code fragment:

```
Vehicle y = new Car();
System.out.println(y);
```

What is the result?

A. 10:20
B. 0:20
C. Compilation fails at line n1
D. Compilation fails at line n2

Answer: D

Question No : 75

Given the code fragment:

```
public class Employee {
    String name;
    boolean contract;
    double salary;
    Employee() {
        // line n1
    }
    public String toString(){
        return name + ":" + contract + ":" + salary;
    }
    public static void main(String[] args) {
        Employee e = new Employee();
        // line n2
        System.out.print(e);
    }
}
```

Which two modifications, when made independently, enable the code to print joe:true: 100.0?

☐ A) Replace line n2 with:

```
e.name = "Joe";
e.contract = true;
e.salary = 100;
```

☐ B) Replace line n2 with:

```
this.name = "Joe";
this.contract = true;
this.salary = 100;
```

☐ C) Replace line n1 with:

```
this.name = new String("Joe");
this.contract = new Boolean(true);
this.salary = new Double(100);
```

☐ D) Replace line n1 with:

```
name = "Joe";
contract = TRUE;
salary = 100.0f;
```

☐ E) Replace line n1 with:

```
this("Joe", true, 100);
```

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

Answer: A,C

Question No : 76

Given:

```
public class Triangle {
    static double area;
    int b = 2, h = 3;
    public static void main(String[] args) {
        double p, b, h;        //line n1
        if (area == 0) {
            b = 3;
            h = 4;
            p = 0.5;
        }
        area = p * b * h;      //line n2
        System.out.println("Area is " + area);
    }
}
```

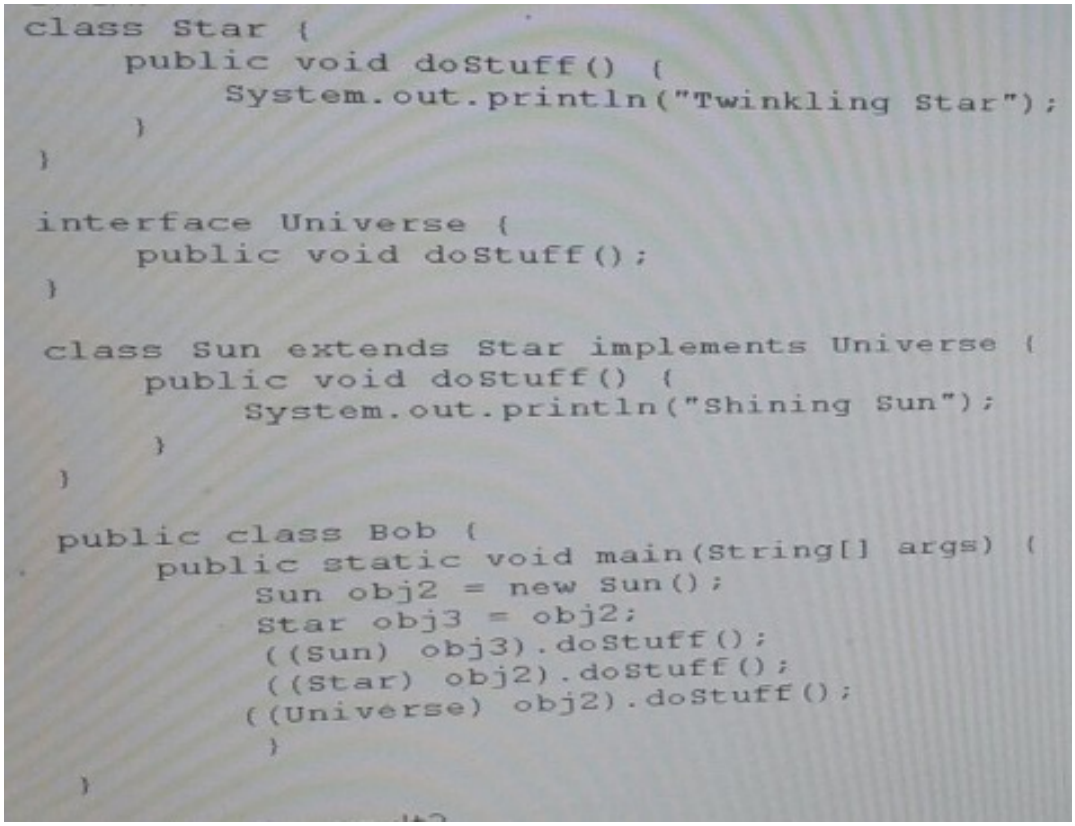
What is the result?

A. Area is 6.0
B. Area is 3.0
C. Compilation fails at line n1
D. Compilation fails at line n2.

Answer: D

Question No : 77

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");
    }
}

public 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

Answer: D

Question No : 78

Given the code fragment?

public class Test {

public static void main(String[] args) {

Test t = new Test();

```
int[] arr = new int[10];

arr = t.subArray(arr,0,2);

}

// insert code here

}
```

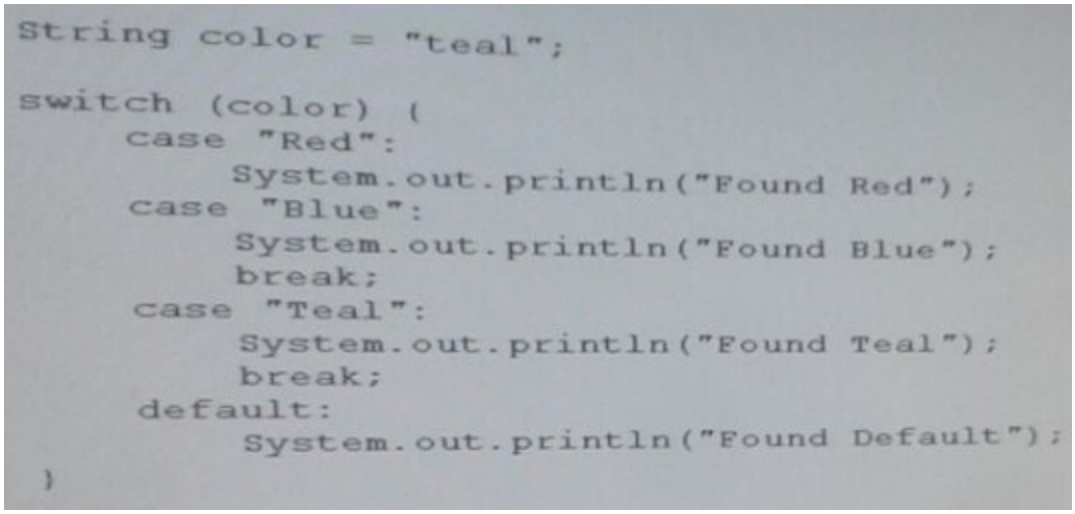
Which method can be inserted at line // insert code here to enable the code to compile?

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

Answer: A

Question No : 79

Given the code fragment:



```
String color = "teal";

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

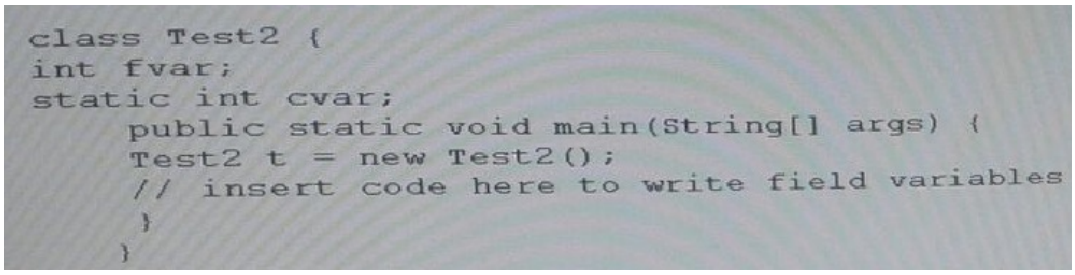
What is the result?

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

Answer: B

Question No : 80

Given the code fragment



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

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

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

Answer: B

Question No : 81

Which of the following will print current time?

A. System.out.print(new LocalTime()-now0);
B. System.out.print(new LocalTime());
C. System.ouLprint(LocalTime.now());
D. System.ouLprint(LocalTime.today());
E. None of the above.

Answer: C

Explanation:

The LocalTime is an interface, so we can't use new keyword with them. So options A and C are incorrect.

To get current time we can call now method on LocalTime interface. So option C is correct.

Option D is incorrect as there is no method called today as in LocalTime interface

https://docs.oracle.com/javase/tutorial/datetime/iso/datetime.html

Question No : 82

Which two statements correctly describe checked exception?

A. These are exceptional conditions that a well-written application should anticipate and recover from.
B. These are exceptional conditions that are external to the application, and that the application usually cannot anticipate or recover from.
C. These are exceptional conditions that are internal to the application, and that the application usually cannot anticipate or recover from.
D. Every class that is a subclass of RuntimeException and Error is categorized as checked exception.
E. Every class that is a subclass of Exception, excluding RuntimeException and its subclasses, is categorized as checked exception.

Answer: B,D

Explanation: Checked exceptions:

* (B) represent invalid conditions in areas outside the immediate control of the program (invalid user input, database problems, network outages, absent files)
* are subclasses of Exception
It's somewhat confusing, but note as well that RuntimeException (unchecked) is itself a subclass of Exception (checked).

* a method is obliged to establish a policy for all checked exceptions thrown by its implementation (either pass the checked exception further up the stack, or handle it somehow)

Reference: Checked versus unchecked exceptions

Question No : 83

Given the code fragment:

```
3. public static void main(String[] args) {
4.     int iVar = 100;
5.     float fVar = 100.100f;
6.     double dVar = 123;
7.     iVar = fVar;
8.     fVar = iVar;
9.     dVar = fVar;
10.    fVar = dVar;
11.    dVar = iVar;
12.    iVar = dVar;
13. }
```

Which three lines fail to compile?

A. Line 7
B. Line 8
C. Line 9
D. Line 10
E. Line 11
F. Line 12

Answer: A,D,F

Question No : 84

Given:

```
public class Test {

public static void main(String[] args) {

int arr[] = new int[4];

arr[0] = 1;

arr[1] = 2;

arr[2] = 4;

arr[3] = 5;

int sum = 0;

try {

for (int pos = 0; pos <= 4; pos++) {

sum = sum + arr[pos];

}

} catch (Exception e) {

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

}

System.out.println(sum);

}

}
```

What is the result?

A. 12
B. Invalid Index
12
C. Invalid Index

D. Compilation fails

Answer: B

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

Question No : 85

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

Answer: B

Question No : 86

Given the code fragment:

```
int a[] = {1, 2, 3, 4, 5};
for(XXX) {
    System.out.print(a[e]);
}
```

Which option can replace xxx to enable the code to print 135?

A. int e = 0; e < = 4; e++
B. int e = 0; e < 5; e + = 2
C. int e = 1; e < = 5; e + = 1
D. int e = 1; e < 5; e+ =2

Answer: B

Question No : 87

Given:

```
public class Series {
    private boolean flag;

    public 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

Answer: B

Question No : 88

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

Answer: A

Question No : 89

Given the code fragment:

```
1. public class Test {
2.     public static void main(String[] args) {
3.         /* insert code here */
4.         array[0]=10;
5.         array[1]=20;
6.         System.out.print(array[0]+":"+array[1]);
7.     }
8. }
```

Which code fragment, when inserted at line 3, enables the code to print 10:20?

A. int[] array n= new int[2];
B. int[] array;
array = int[2];
C. int array = new int[2];
D. int array [2] ;

Answer: A

Question No : 90

Given:

```
public class TestOperator {

public static void main(String[] args) {

int result = 30 -12 / (2*5)+1;

System.out.print("Result = " + result);


}

}
```

What is the result?

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

Answer: E

Question No : 91

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();

Answer: C

Question No : 92

Given:

```
class A {
    public A(){
        System.out.print("A ");
    }
}

class B extends A{
    public B(){                          //line n1
        System.out.print("B ");
    }
}

class C extends B{

    public C(){                          //line n2
        System.out.print("C ");
    }
    public static void main(String[] args) {
        C c = new C();
    }
}
```

What is the result?

A. C B A
B. C
C. A B C
D. Compilation fails at line n1 and line n2

Answer: C

Question No : 93

Given the fragment:

```
String[][] arra = new String[3][];

arra[0] = new String[]{"rose", "lily"};

arra[1] = new String[]{"apple", "berry","cherry","grapes"};

arra[0] = new String[]{"beans", "carrot","potato"};

// insert code fragment here
```

Which code fragment when inserted at line '// insert code fragment here', enables the code to successfully change arra elements to uppercase?

A.
```
String[][] arra = new String[3][];
arra[0] = new String[]{"rose", "lily"};
arra[1] = new String[]{"apple", "berry","cherry","grapes"};
arra[0] = new String[]{"beans", "carrot","potato"};
for (int i = 0; i < arra.length; i++) {
for (int j=0; j < arra[i].length; j++) {
arra[i][j] = arra[i][j].toUpperCase();
}
}
```
B.
```
for (int i = 0; i < 3; i++) {
for (int j=0; j < 4; j++) {
arra[i][j] = arra[i][j].toUpperCase();
}
}
```
C.
```
for (String a[]:arra[][]) {
for (String x:a[]) {
```
D.
```
toUpperCase();
}
```

}
E. for (int i:arra.length) {
for (String x:arra) {
arra[i].toUpperCase();
}
}

Answer: C

Explanation:

Incorrect:

not A: arra.length is 3, but the subarrays have 2, 3 and 4 elements. Index will be out of bound.

not B: The subarrys are of different lengths. Index will be out of bound.

not D: Compile error.

Question No : 94

Given:

public class ScopeTest {

int j, int k;

public static void main(String[] args) {

ew ScopeTest().doStuff(); }

void doStuff() {

nt x = 5;

oStuff2();

System.out.println("x");

}

void doStuff2() {

nt y = 7;

ystem.out.println("y");

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

ystem.out.println("z");

ystem.out.println("y");

}

Which two items are fields?

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

Answer: A,B

Question No : 95

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

Answer: A

Question No : 96

Given the following classes:

```
public class Employee {
    public int salary;
}

public class Manager extends Employee {
    public int budget;
}

public class Director extends Manager {
    public int stockOptions;
}
```

And given the following `main` method:

```
public static void main(String[] args) {
    Employee employee = new Employee();
    Manager  manager  = new Manager();
    Director director = new Director();
    //line n1
}
```

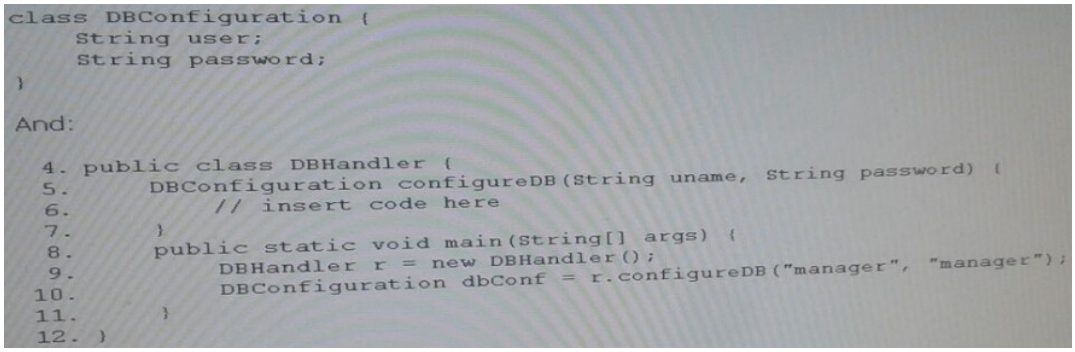
Which two options fail to compile when placed at line n1 of the main method?

A. employee.salary = 50_000;
B. director.salary = 80_000;
C. employee.budget = 200_000;
D. manager.budget = 1_000_000;
E. manager.stockOption = 500;
F. director.stockOptions = 1_000;

Answer: C,E

Question No : 97

Given the code format:



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

And:

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

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

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

Answer: B

Question No : 98

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

Answer: B

Explanation: The while statement is incorrect. It has the syntax of a for statement.

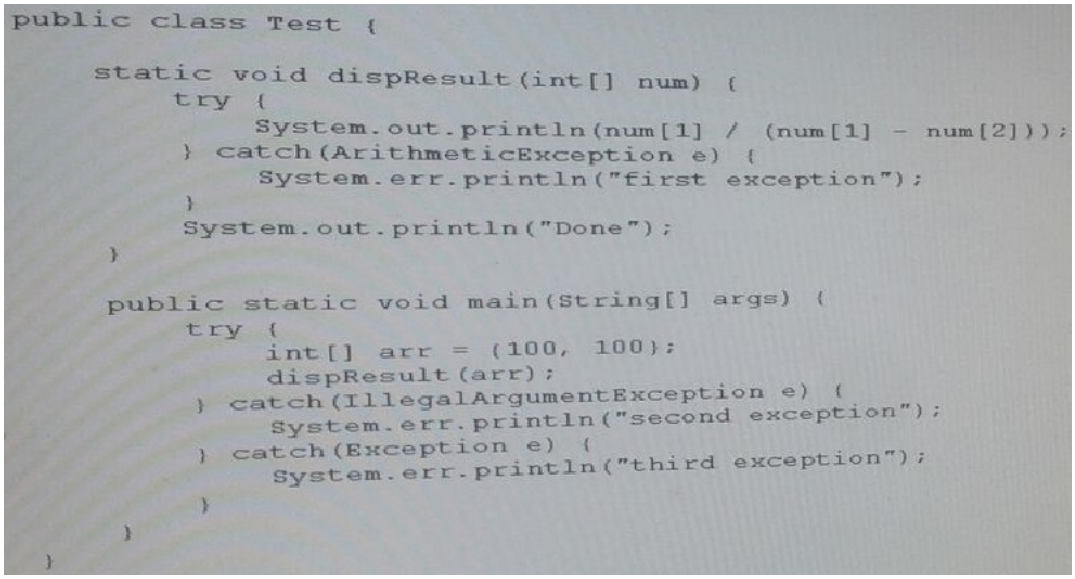
The while statement continually executes a block of statements while a particular condition is true. Its syntax can be expressed as:

while (expression) {
statement(s)
}
The while statement evaluates expression, which must return a boolean value. If the expression evaluates to true, the while statement executes the statement(s) in the while block. The while statement continues testing the expression and executing its block until the expression evaluates to false.

Reference: The while and do-while Statements

Question No : 99

Given:



```
public class Test {

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

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

What is the result?

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

Answer: B

Question No : 100

Given:

```
class Test {
    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

答案a是错的，new过之后值就会被清空

Answer: A

B

Question No : 101

Given the code fragment:

```
4. public static void main(String[] args) {
5.     boolean opt = true;
6.     switch (opt) {
7.         case true:
8.             System.out.print("True");
9.             break;
10.        default:
11.            System.out.print("***");
12.    }
13.    System.out.println("Done");
14. }
```

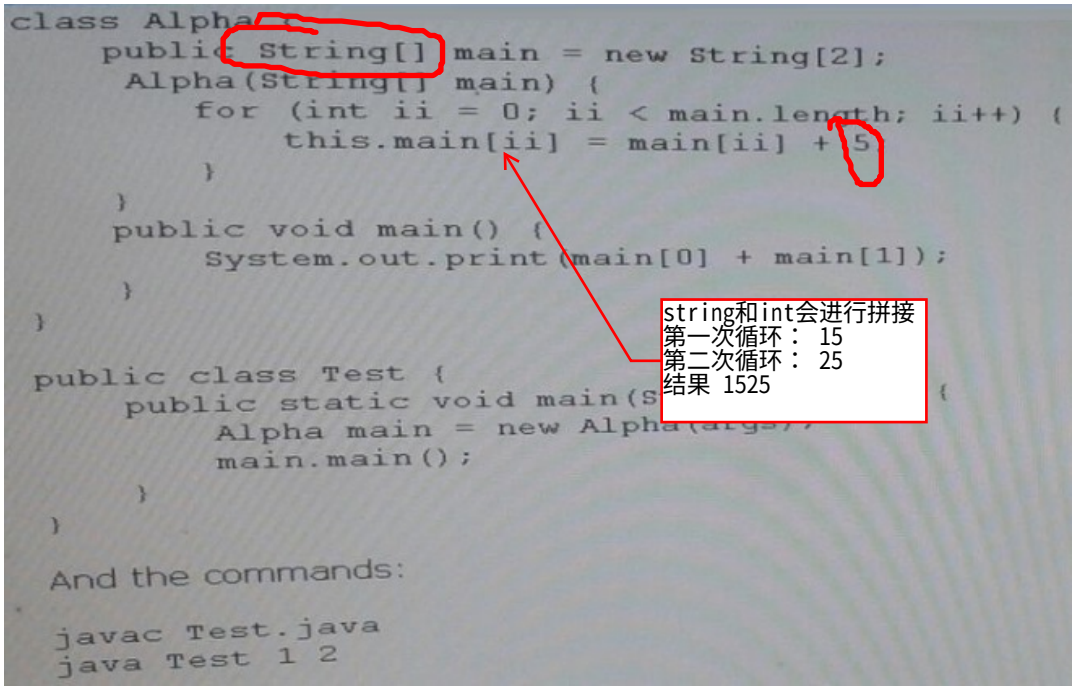
Which modification enables the code fragment to print TrueDone?

A. Replace *line* 5 With String result = "true";
Replace line 7 with case "true":
B. Replace line 5 with boolean opt = l;
Replace line 7 with case 1=
C. At line 9, remove the break statement.
D. Remove the default section.

Answer: A

Question No : 102

Give:



What is the result?

A. 1525
B. 13
C. Compilation fails
D. An exception is thrown at runtime
E. The program fails to execute due to runtime error

Answer: D

Question No : 103

Given the code fragment:

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

What is the result if the integer aVar is 9?

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

Answer: A

Question No : 104

Given the content of three files:

A.java:

```
public class A {
    public void a() {}
    int a;
}
```

B.java:

```
public class B {
    private int doStuff() {
        private int x = 100;
        return x++;
    }
}
```

C.java:

```
import java.io.*;
package p1;
class A {
    public void main(String fileName) throws IOException { }
}
```

Which statement is true?

Which statement is true?

A. Only the A.Java file compiles successfully.
B. Only the B.java file compiles successfully.
C. Only the C.java file compiles successfully.
D. The A.Java and B.java files compile successfully.
E. The B.java and C.java files compile successfully.
F. The A.Java and C.java files compile successfully.

Answer: A

Explanation: In class B.Java doStuff() has access modifier with variable name which is not allowed. C.Java class name is different than file name. Only private classes can have different names than file names

Question No : 105

You are developing a banking module. You have developed a class named ccMask that has a maskcc method.

Given the code fragment:

```
class CCMask {
    public static String maskCC(String creditCard) {
        String x = "XXXX-XXXX-XXXX-";
        //line n1
    }

    public static void main(String[] args) {
        System.out.println(maskCC("1234-5678-9101-1121"));
    }
}
```

You must ensure that the maskcc method returns a string that hides all digits of the credit card number except the four last digits (and the hyphens that separate each group of four digits).

Which two code fragments should you use at line n1, independently, to achieve this

requirement?

☐ A)
```
StringBuilder sb = new StringBuilder(creditCard);
sb.substring(15, 19);
return x + sb;
```

☐ B)
```
return x + creditCard.substring(15, 19);
```

☐ C)
```
StringBuilder sb = new StringBuilder(x);
sb.append(creditCard, 15, 19);
return sb.toString();
```

☐ D)
```
StringBuilder sb = new StringBuilder(creditCard);
StringBuilder s = sb.insert(0, x);
return s.toString();
```

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

Answer: B,C

Question No : 106

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

Answer: C
Explanation: Hello Java! 8

Question No : 107

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

Answer: B

Explanation: obj1 and obj3.

when you do e2 = e1 you're copying object references - you're not making a copy of the object - and so the variables e1 and e2 will both point to the same object.

Question No : 108

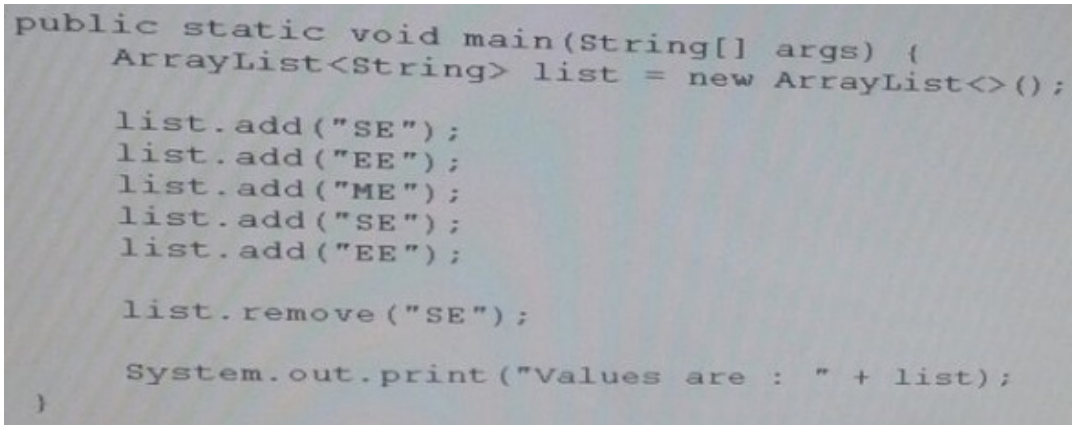
Which three statements describe the object-oriented features of the Java language?

A. Objects cannot be reused.
B. A subclass can inherit from a superclass.
C. Objects can share behaviors with other objects.
D. A package must contain more than one class.
E. Object is the root class of all other objects.
F. A main method must be declared in every class.

Answer: B,C,E

Question No : 109

Given the code fragment:



```
public static void main(String[] args) {
    ArrayList<String> list = new ArrayList<>();

    list.add("SE");
    list.add("EE");
    list.add("ME");
    list.add("SE");
    list.add("EE");

    list.remove("SE");

    System.out.print("Values are : " + list);
}
```

What is the result?

A. Values are : [EE, ME]
B. Values are : [EE, EE, ME]
C. Values are : [EE, ME, EE]
D. Values are : [SE, EE, ME, EE]
E. Values are : [EE, ME, SE, EE]

Answer: E

Question No : 110

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

Answer: A

Question No : 111

Given the code fragment:

```
public class ForTest {

public static void main(String[] args) {

int[] array = {1, 2, 3};

for ( foo ) {

}

}
```

Which three code fragments, when replaced individually for foo, enables the program to compile?

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

Answer: A,B,C

Question No : 112

Given the code fragment from three files:

SalesMan.java:

```
package sales;
public class SalesMan { }
```

Product.java:

```
package sales.products;
public class Product { }
```

Market.java:

```
1. package market;
2. // insert code here
3. public class USMarket {
4.     SalesMan sm;
5.     Product p;
6. }
```

Which code fragment, when inserted at line 2, enables the code to compile?

A)
```
import sales.*;
```

B)
```
import java.sales.products.*;
```

C)
```
import sales;
import sales.products;
```

D)
```
import sales.*;
import products.*;
```

E)
```
import sales.*;
import sales.products.*;
```

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

E. Option E

Answer: E

Question No : 113

Given the code fragment:

```
public static void main(String[] args) {
    ArrayList myList = new ArrayList();
    String[] myArray;
    try {
        while (true) {
            myList.add("My String");
        }
    }
    catch (RuntimeException re) {
        System.out.println("Caught a RuntimeException");
    }
    catch (Exception e) {
        System.out.println("Caught an Exception");
    }
    System.out.println("Ready to use");
}
```

Exception in thread "main" java.lang.OutOfMemoryError: Java heap space

What is the result?

A. Execution terminates in the first catch statement, and caught a RuntimeException is printed to the console.
B. Execution terminates In the second catch statement, and caught an Exception is printed to the console.
C. A runtime error is thrown in the thread "main"
D. Execution completes normally, and Ready to us© is printed to the console.
E. The code fails to compile because a throws keyword is required.

Answer: C

Question No : 114

Given the code fragment: