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?
却少无参数的构造方法，却少 person 方法。

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. **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. AnArrayIndexOutOfBou
ndsException 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. C. patient p = new Patient (“Mike”); int f = pas.indexOf(P)
D. int f = ps.indexOf(p2);

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 纠正为 **BCF**

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.

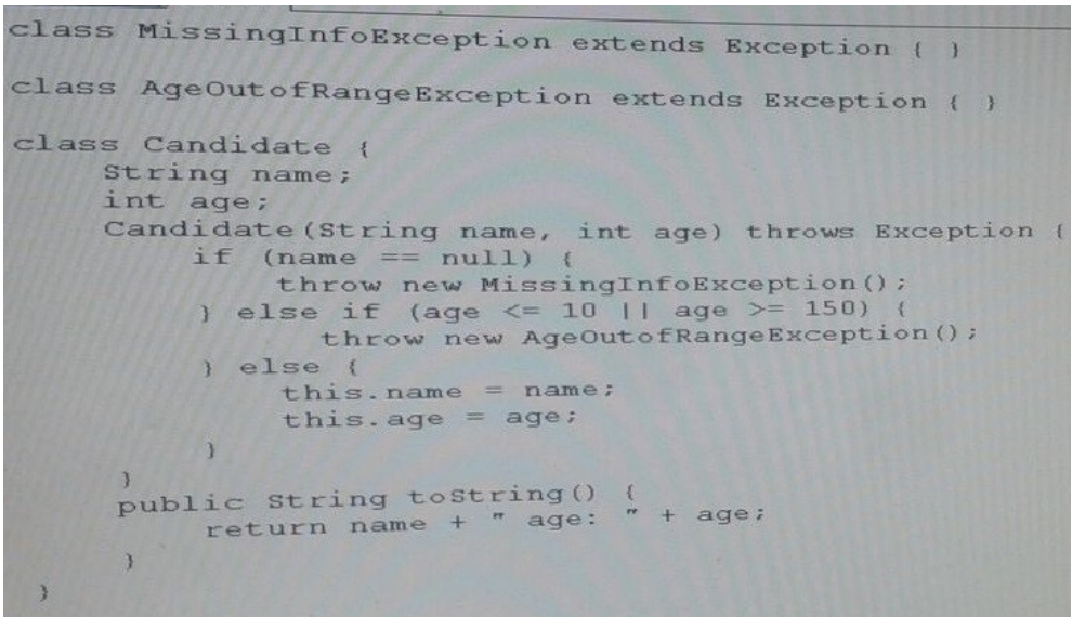
Sometimes: Just data structure.

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.

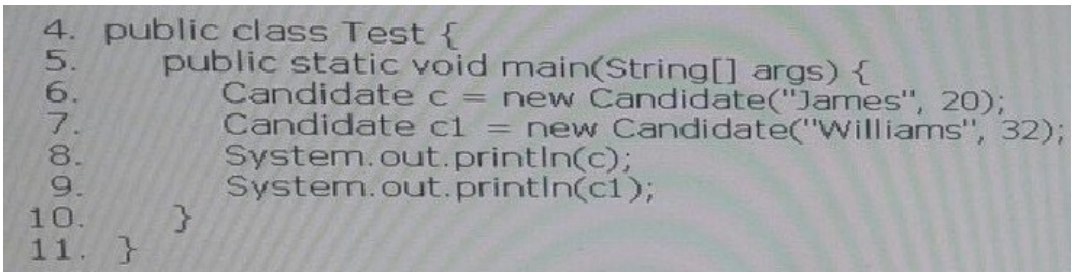


```
class MissingInfoException extends Exception { }

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

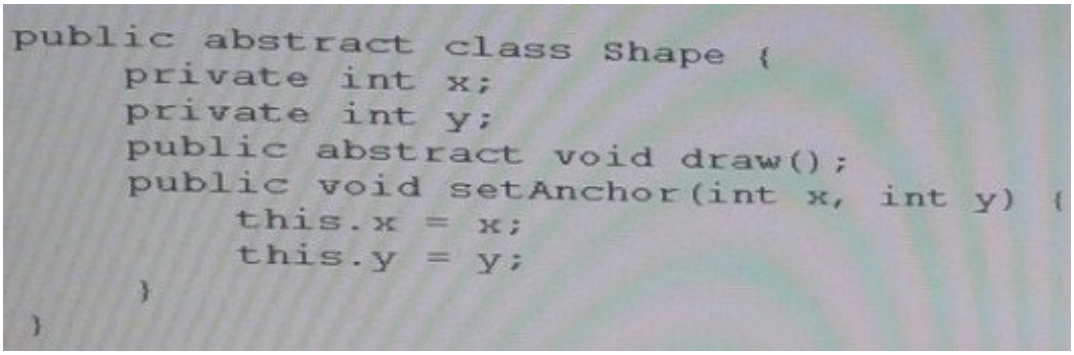
Which statement best describes encapsulation?

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

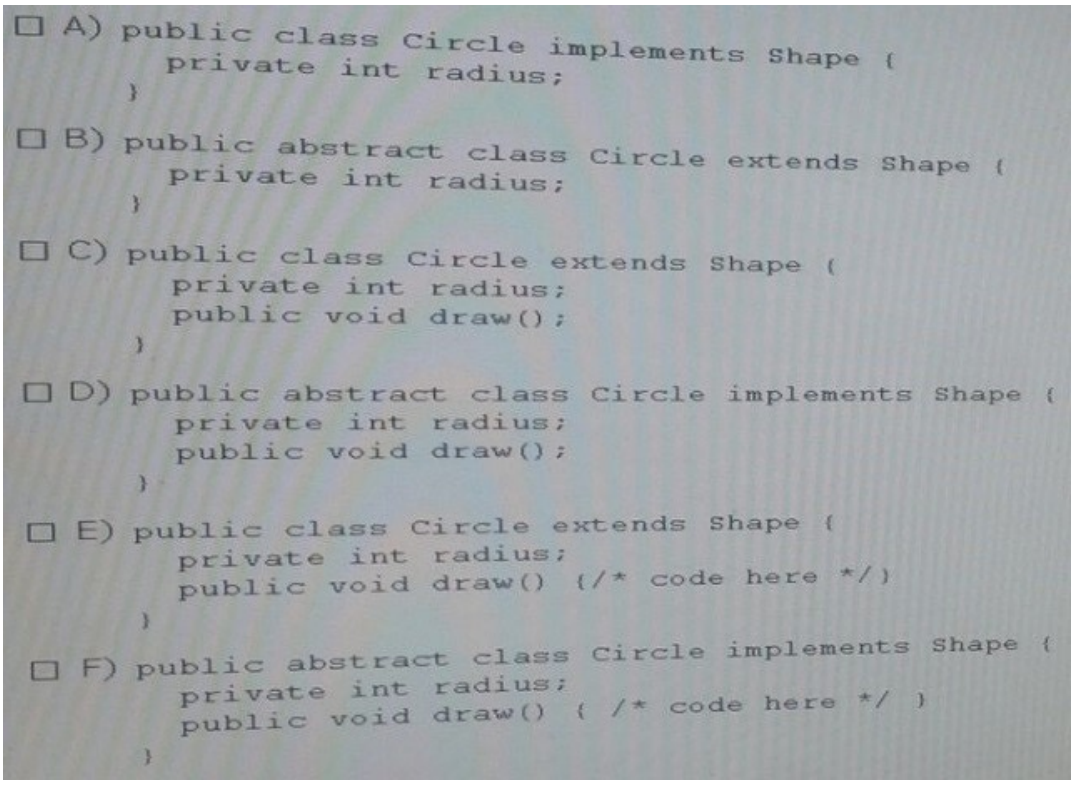
Question No : 35

Given:



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

Which two classes use the shape class correctly?



A. 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 ArrayIndexOutOfBoundsException 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. **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.

```
interface Runnable{
    public void run();
}
```

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
    }
}
```

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));
  }
}
```

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?

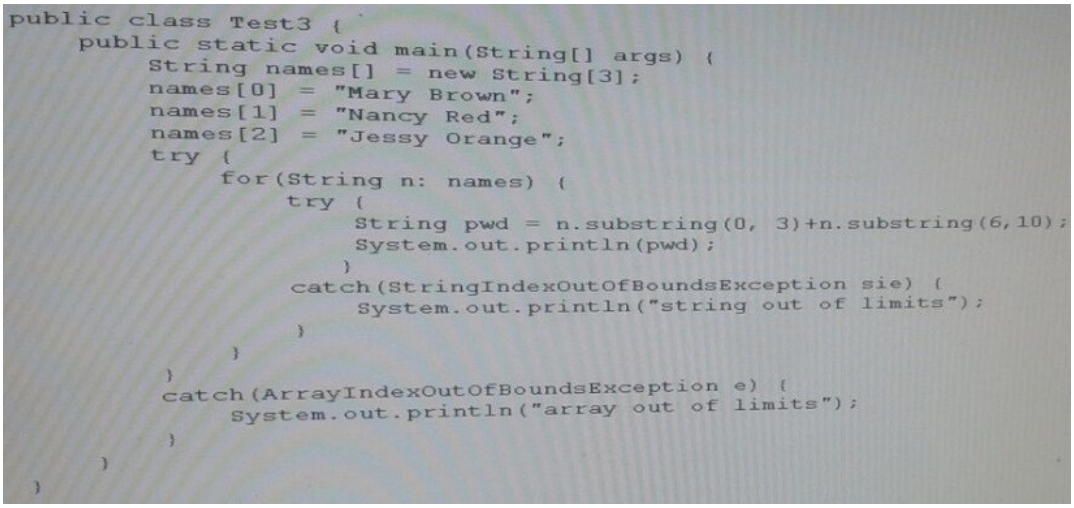
A. Base
DerivedA
B. Base
DerivedB
C. DerivedB
DerivedB
D. DerivedB
DerivedA

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)
```
planets
Earth
1
```

B)
```
[LPlanets.Planet;@15db9742
Earth
1
```

C)
```
[LPlanets.Planet;@15db9742
Planets.Planet@6d06d69c
1
```

D)
```
[LPlanets.Planet;@15db9742
Planets.Planet@6d06d69c
[LPlanets.Moon;@7852e922
```

E)
```
[LPlanets.Planet;@15db9742
Venus
0
```

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!");
}
```

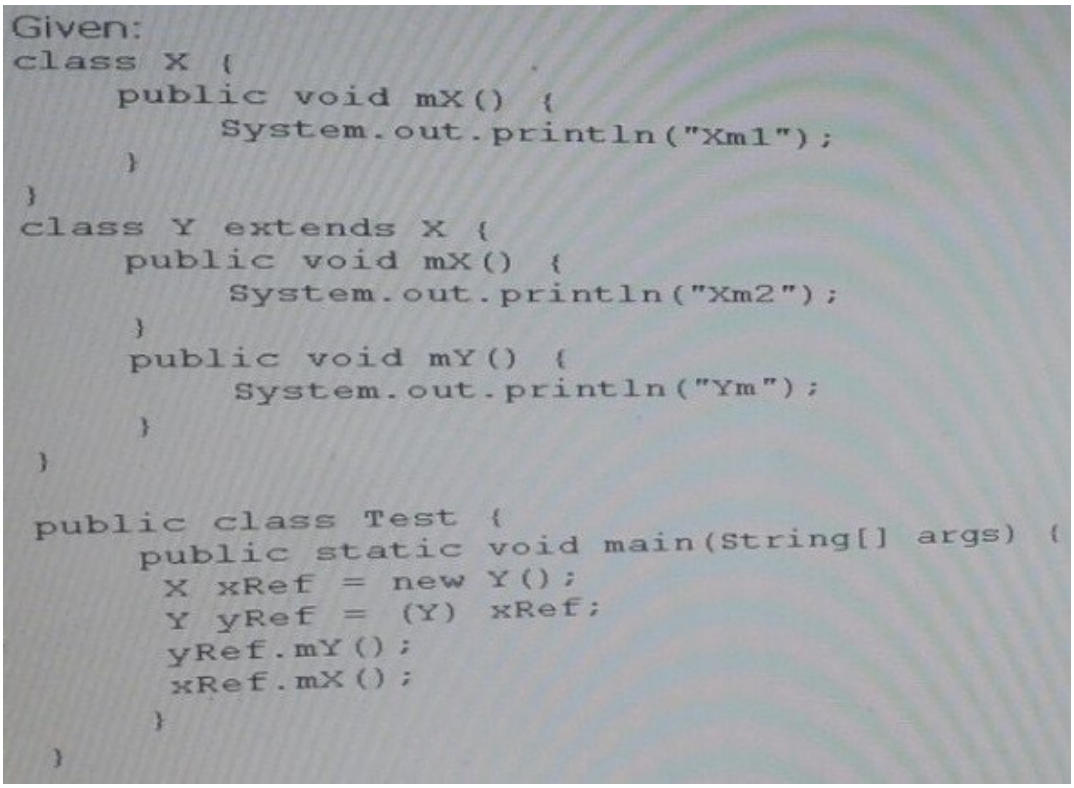
What is the result if the integer aVar is 9?

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));
    }
}
```

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.utiLArrayList;
2. import java.utiLList;
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.removelf(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:

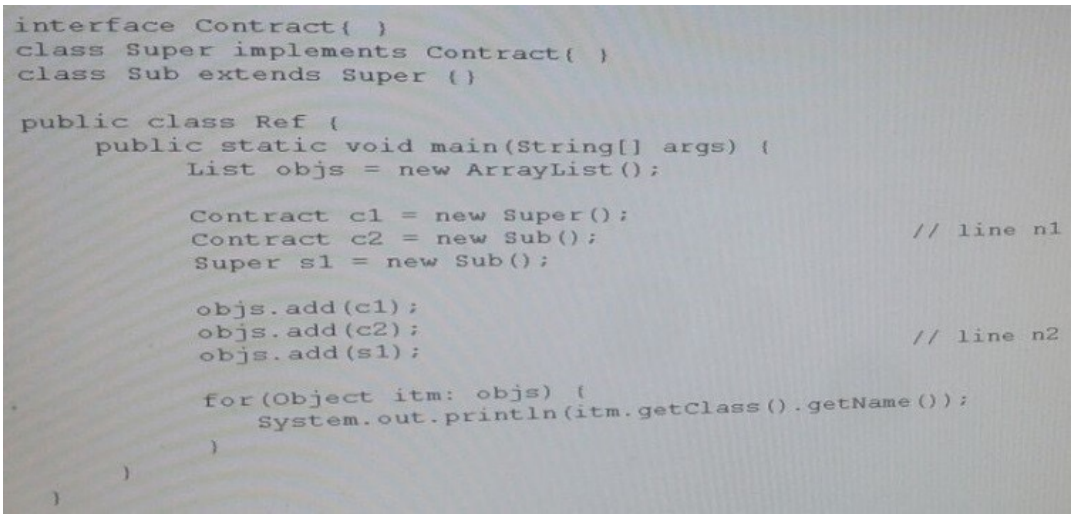
Option D is the correct answer.

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.

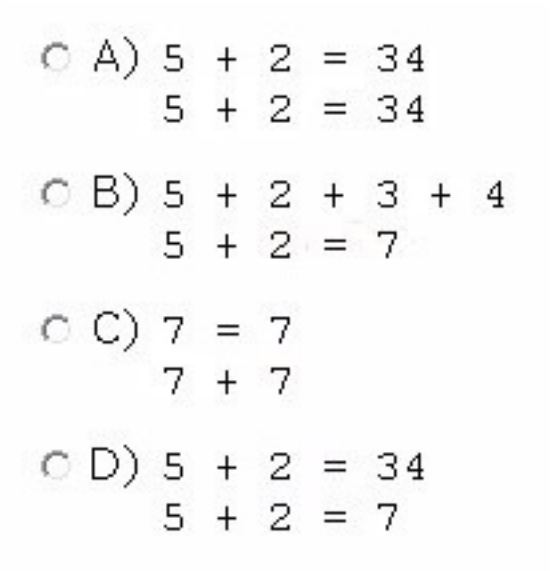
Answer: A

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:

```
class Test {
    int sum = 0;
    public void doCheck(int number) {
        if (number % 2 == 0) {
            break;
        } else {
            for (int i = 0; i < number; i++) {
                sum += i;
            }
        }
    }
    public static void main(String[] args) {
        Test obj = new Test();
        System.out.println("Red " + obj.sum);
        obj.doCheck(2);
        System.out.println("Orange " + obj.sum);
        obj.doCheck(3);
        System.out.println("Green " + obj.sum);
    }
}
```

What is the result?
break 不能用在循环和 switch 结构之外。

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 programmerdefined 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);
    }
}
```

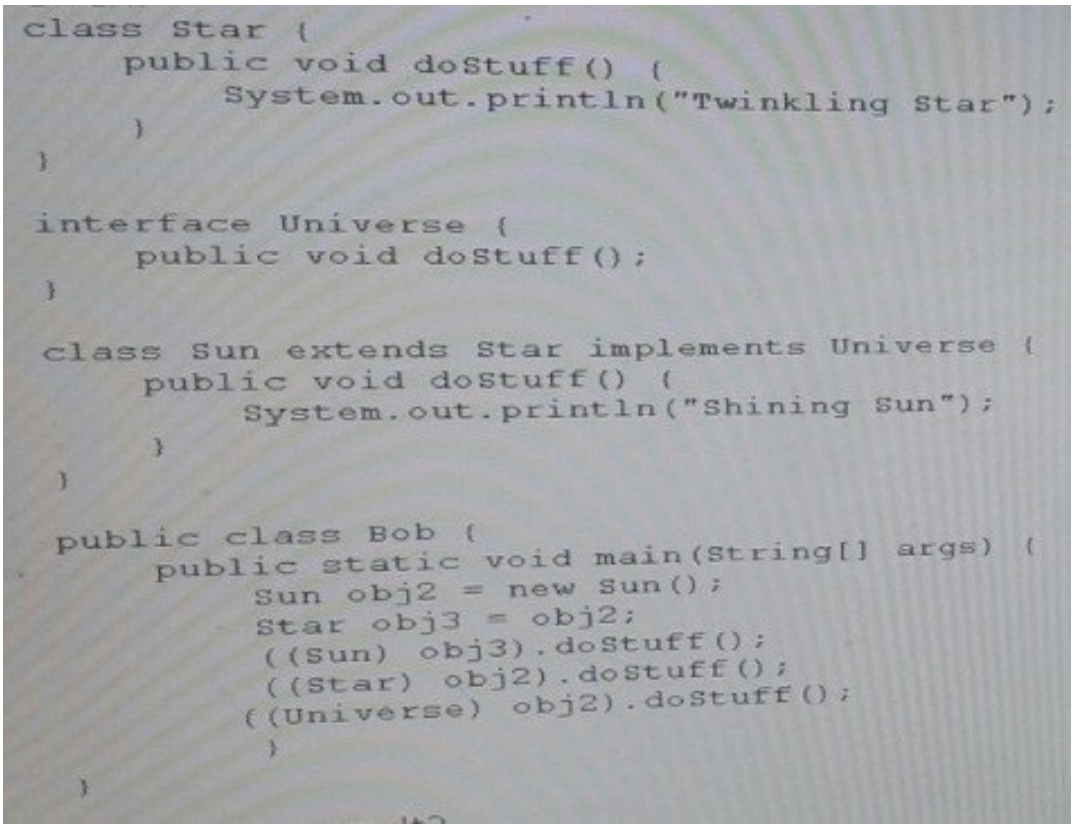
What is the result?

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 就这个为 **AE**

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 itsimplementation (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
}
```

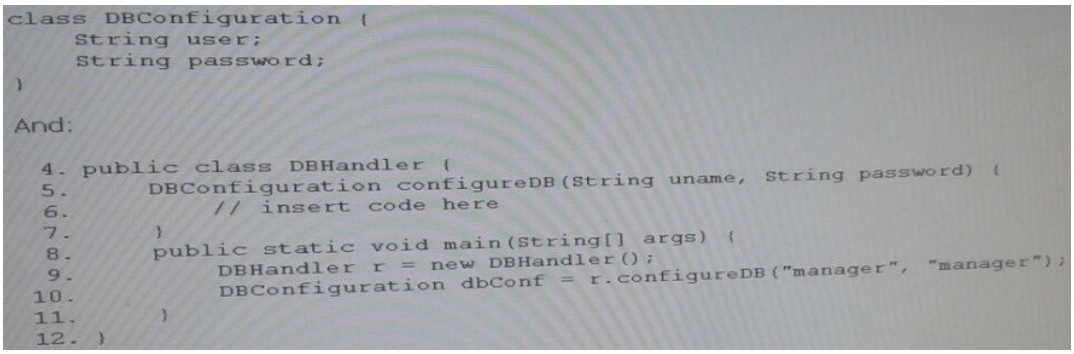
Which two options fail to compile when placed at line n1 of the main method?

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

Answer: A

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:

```
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] + main[1]);
    }
}

public 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
```

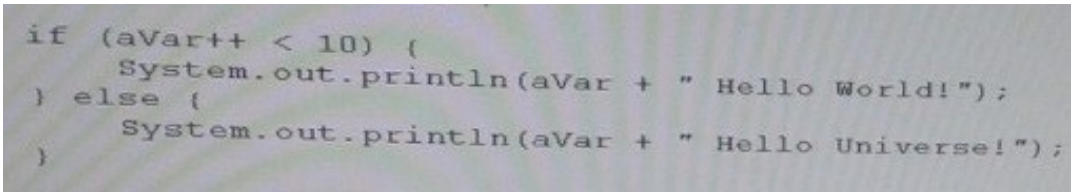
What is the result?

A. 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");
}
```

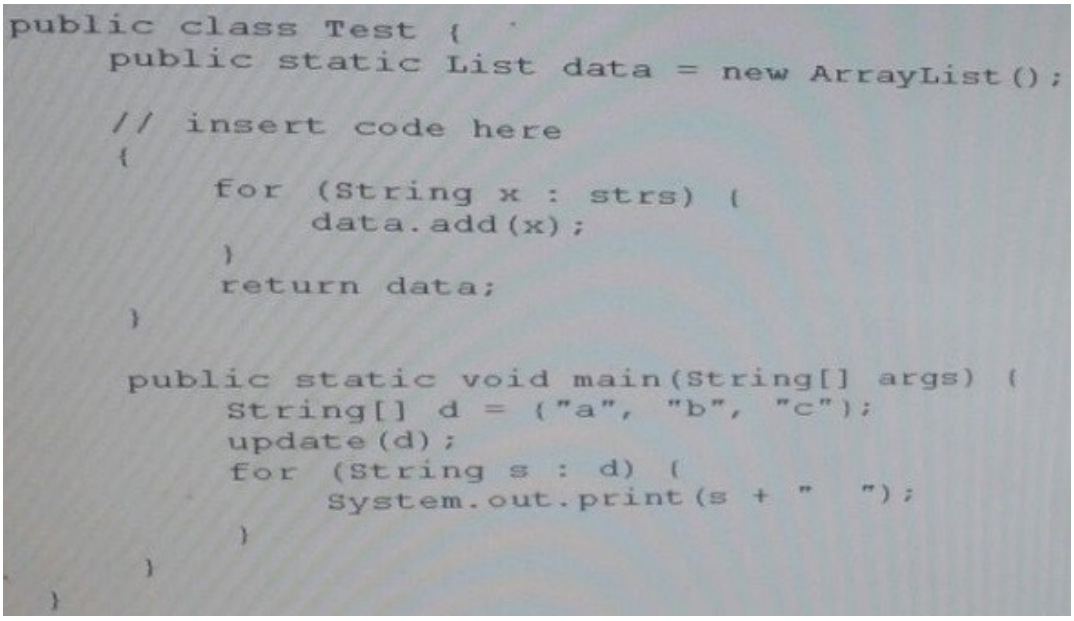
What is the result?

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:



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

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

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

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

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

update(d) 调用本类中的方法需要方法为 static，同时需要返回 list。答案为 c

Answer: E

Question No : 115

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 :

Answer: C

Question No : 116

Given the code fragment:

int b = 3;

if (!(b > 3)) { System.out.println("square ");

}{

System.out.println("circle ");

}

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

What is the result?

A. square...
B. circle...
C. squarecircle...
D. Compilation fails.

Answer: C

Question No : 117

Given:

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

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

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

Answer: B

Question No : 118

Given the code fragment:
int[] array = {I, 2, 3, 4, 5};

And given the requirements:

- Process all the elements of the array in the order of entry.
- Process all the elements of the array in the reverse order of entry.
- Process alternating elements of the array in the order of entry.

Which two statements are true?

A. Requirements 1, 2, and 3 can be implemented by using the enhanced for loop.
B. Requirements 1, 2, and 3 can be implemented by using the standard for loop.
C. Requirements 2 and 3 CANNOT be implemented by using the standard for loop.
D. Requirement 1 can be implemented by using the enhanced for loop.
E. Requirement 3 CANNOT be implemented by using either the enhanced for loop or the standard for loop.

Answer: D,E

Question No : 119

Given the code fragment:

// insert code here

```
arr[0] = new int[3];

arr[0][0] = 1;

arr[0][1] = 2;

arr[0][2] = 3;


arr[1] = new int[4];

arr[1][0] = 10;

arr[1][1] = 20;

arr[1][2] = 30;

arr[1][3] = 40;
```

Which two statements, when inserted independently at line // insert code here, enable the code to compile?

A. int [] [] arr = null;
B. int [] [] arr = new int [2];
C. int [] [] arr = new int [2] [ ];
D. int [] [] arr = new int [] [4];
E. int [] [] arr = new int [2] [0];
F. int [] [] arr = new int [0] [4];

Answer: C,E

Question No : 120

Given:

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

What is result?

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

Answer: C

Question No : 121

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.

Answer: A,D

Question No : 122

Given:

Given:

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

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

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

What is the result?

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

Answer: B

Question No : 123

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<Integer> list = new ArrayList<>();
8.                    list.add(21); list.add(13);
9.                    list.add(30); list.add(11);
10.                   list.add(2);
11.                   //insert here
12.                   System.out.println(list);
13.         }
14. }
```

Which inserted at line 11, will provide the following output?

[21, 15, 11]

A. list.removeIf(e > e%2 != 0);
B. list.removeIf(e -> e%2 != 0);
C. Ust.removeIf(e -> e%2 = 0); **D.** list.remove(e -> e%2 = 0);
E. None of the above.

Answer: C

Explanation:
In output we can see that only odd numbers present, so we need to remove only even numbers to get expected output. From Java SE 8, there is new method call removeIf which takes predicate object and remove elements which satisfies predicate condition. Predicate has functional method call take object and check if the given condition met or not, if met it returns true, otherwise false. Option C we have passed correct lambda expression to check whether the number is odd or even that matches to the functional method of predicate interface.

Option A is incorrect as it is invalid lambda expression. Option B is incorrect as it removes all odd numbers.

Option D is incorrect as there is no remove method that takes predicate as argument.
https://docs.oracle.eom/javase/8/docs/api/java/util/ArrayList.html

Question No : 124

Given the following code:

```
int[] intArr = {15, 30, 45, 60, 75};
intArr[2] = intArr[4];
intArr[4] = 90;
```

What are the values of each element in intArr after this code has executed?

A. 15, 60, 45, 90, 75
B. 15, 90, 45, 90, 75
C. 15, 30, 75, 60, 90
D. 15, 30, 90, 60, 90
E. 15, 4, 45, 60, 90

Answer: C

Question No : 125

Given the code from the Greeting.Java file:

```
public class Greeting {
    public static void main(String[] args) {
        System.out.println("Hello " + args[0]);
    }
}
```

Which set of commands prints Hello Duke in the console?

A)
```
javac Greeting
java Greeting Duke
```

B)
```
javac Greeting.java Duke
java Greeting
```

C)
```
javac Greeting.java
java Greeting Duke
```

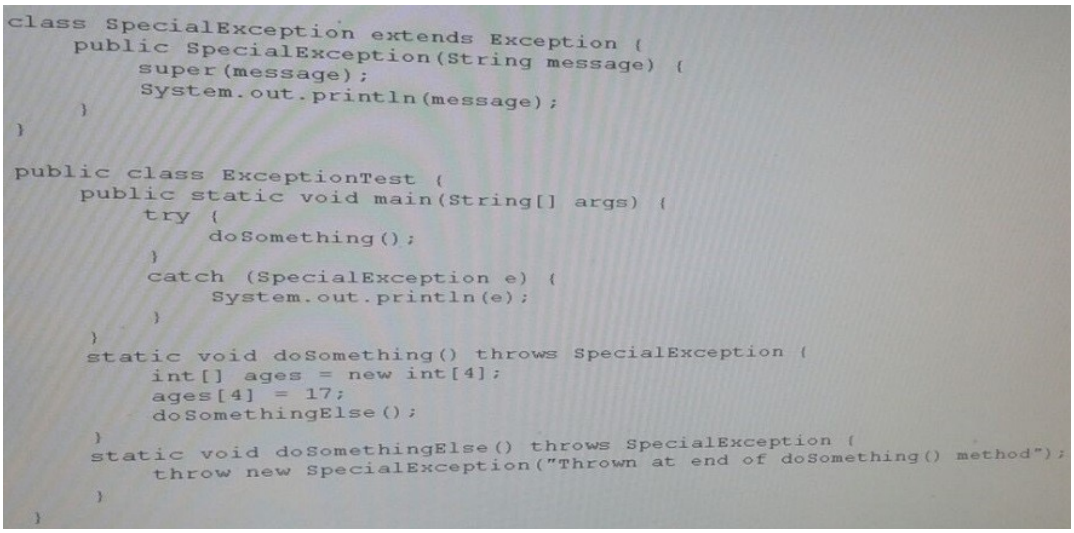
D)
```
javac Greeting.java
java Greeting.class Duke
```

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

Answer: C

Question No : 126

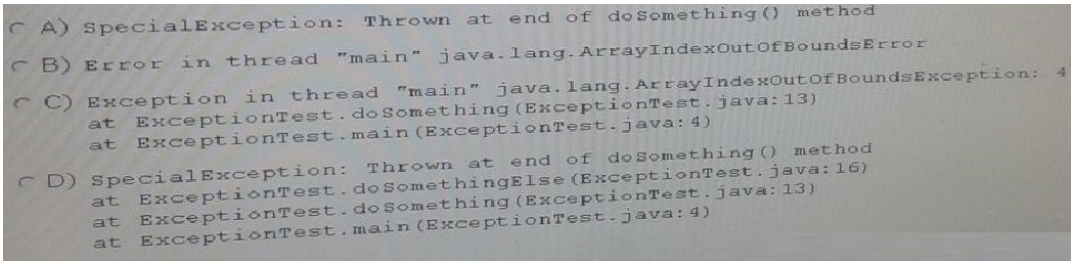
Given:



```
class SpecialException extends Exception {
    public SpecialException(String message) {
        super(message);
        System.out.println(message);
    }
}

public 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 end of doSomething() method");
    }
}
```

What will be the output?



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

Answer: D

Question No : 127

Given the code fragment:

```
abstract class Planet {
    protected void revolve() {            //line n1
    }

    abstract void rotate();               //line n2
}

class Earth extends Planet {
    void revolve() {                      //line n3
    }

    protected void rotate() {             //line n4
    }
}
```

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

A. Make the method at line n1 public.
B. Make the method at line n2 public.
C. Make the method at line n3 public.
D. Make the method at line n3 protected.

E. Make the method at line n4 public.

Answer: C,D

Question No : 128

```
Class StaticField {

static int i = 7;

public static void main(String[] args) {

StaticFied obj = new StaticField();

obj.i++;

StaticField.i++;

obj.i++;

System.out.println(StaticField.i + " "+ obj.i);

}

}
```

What is the result?

A. 10 10
B. 8 9
C. 9 8
D. 7 10

Answer: A

Question No : 129

Given:

Given:

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

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

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

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

Answer: C

Question No : 130

Given:

MainTest.java:

```
public class MainTest {

    public static void main(int[] args) {
        System.out.println("int main " + args[0]);
    }
    public static void main(Object[] args) {
        System.out.println("Object main " + args[0]);
    }
    public static void main(String[] args) {
        System.out.println("String main " + args[0]);
    }
}
```

and commands:

```
javac MainTest.java
java MainTest 1 2 3
```

What is the result?

A. int main 1
B. Object main 1
C. String main 1
D. Compilation fails
E. An exception is thrown at runtime

Answer: C

Question No : 131

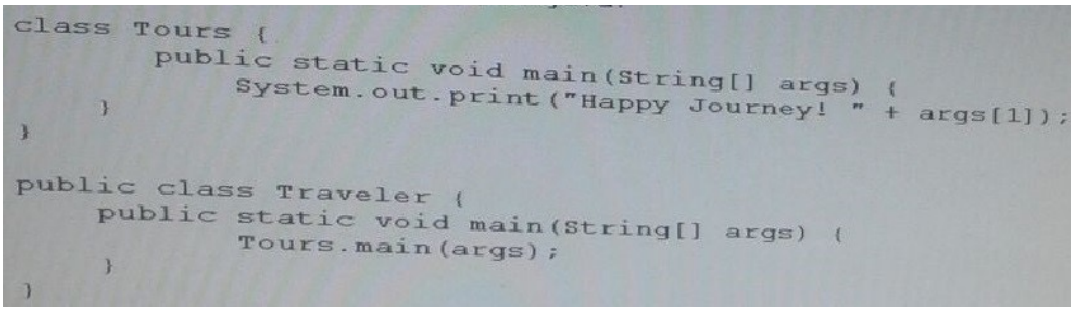
Which three statements are benefits of encapsulation?

A. Allows a class implementation to change without changing t he clients
B. Protects confidential data from leaking out of the objects
C. Prevents code from causing exceptions
D. Enables the class implementation to protect its invariants
E. Permits classes to be combined into the same package
F. Enables multiple instances of the same class to be created safely

Answer: A,B,D

Question No : 132

Given the code in a file Traveler.java:



```
class Tours {
        public static void main(String[] args) {
            System.out.print("Happy Journey! " + args[1]);
    }
}

public class Traveler {
    public static void main(String[] args) {
            Tours.main(args);
    }
}
```

And the commands:

Javac Traveler.java

Java Traveler Java Duke

What is the result?

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

Answer: D

Question No : 133

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 obj3 = null;
        obj2.num = 60;
        graceMarks(obj2);
    }
}
```

How many MarkList instances are created in memory at runtime?

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

Answer: A

Question No : 134

Given:

```
public class Calculator {
    public static void main(String[] args) {
        int num = 5;
        int sum;

        do {
            sum += num;
        } while ((num--) > 1);

        System.out.println("The sum is " + sum + ".");
    }
}
```

What is the result?
sum 需要初始化

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

Answer: E

Question No : 135

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

Answer: C

Question No : 136

Given:

public class Equal {

public static void main(String[] args) {

String str1 = "Java";

String[] str2 = {"J","a","v","a"};

String str3 = "";

for (String str : str2) {

str3 = str3+str;

}

boolean b1 = (str1 == str3);

boolean b2 = (str1.equals(str3));

System.out.print(b1+", "+b2);

}

What is the result?

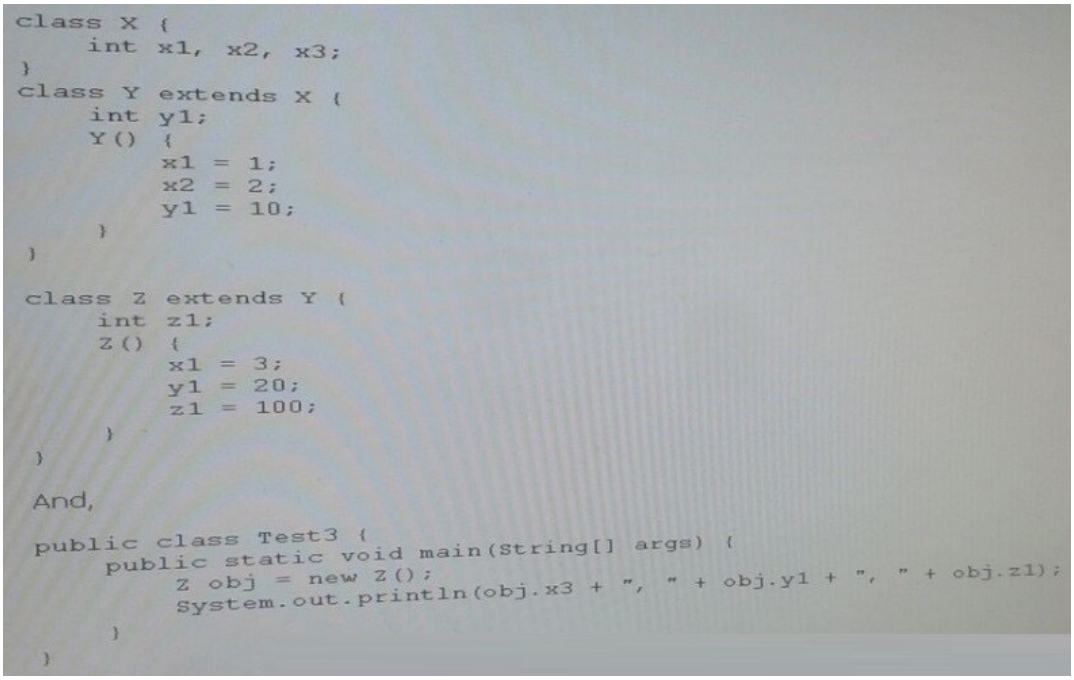
A. true, false
B. false, true
C. true, true
D. false, false

Answer: B

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

Question No : 137

Given:



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

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

And,

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

Which constructor initializes the variable x3?

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

Answer: C

Question No : 138

Given the classes:

* AssertionError

* ArithmeticException
* ArrayIndexOutofBoundsException

* FileNotFoundException

* IllegalArgumentException

* IOError

* IOException

* NumberFormatException

* SQLException

Which option lists only those classes that belong to the unchecked exception category?

A. AssertionError, ArrayIndexOutOfBoundsException, ArithmeticException
B. AssertionError, IOError, IOException
C. ArithmeticException, FileNotFoundException, NumberFormatException
D. FileNotFoundException, IOException, SQLException
E. ArrayIndexOutOfBoundException, IllegalArgumentException, FileNotFoundException

Answer: A

Explanation: Not B: IOError and IOException are both checked errors.

Not C, not D, not E: FileNotFoundException is a checked error.

Note:

Checked exceptions:

* represent invalid conditions in areas outside the immediate control of the program (invaliduser input, database problems, network outages, absent files)

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

Note:

Unchecked exceptions:

* represent defects in the program (bugs) - often invalid arguments passed to a non-privatemethod. To quote from The Java Programming Language, by Gosling, Arnold, and Holmes: "Unchecked runtime exceptions represent conditions that, generally speaking, reflect errors in your program's logic and cannot be reasonably recovered from at run time."
* are subclasses of RuntimeException, and are usually implemented using IllegalArgumentException, NullPointerException, or IllegalStateException
* method is not obliged to establish a policy for the unchecked exceptions thrown by itsimplementation (and they almost always do not do so)

Question No : 139

Given the code fragment:

```
public class App {
    public static void main(String[] args) {
        String str1 = "Java";
        String str2 = new String("java");
        //line n1
        {
           System.out.println("Equal");
        } else {
           System.out.println("Not Equal");
        }
    }
}
```

Which code fragment, when inserted at line n1, enables the App class to print Equal?

A)
```
String str3 = str2;
if (str1 == str3)
```

B)
```
if (str1.equalsIgnoreCase(str2))
```

C)
```
String str3 = str2;
if (str1.equals(str3))
```

D)
```
if (str1.toLowerCase() == str2.toLowerCase())
```

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

Answer: B

Question No : 140

Consider

Integer number = Integer.valueOff 808.1");

Which is true about the above statement?

A. The value of the variable number will be 808.1 **B.**
The value of the variable number will be 808
C. The value of the variable number will be 0.
D. A NumberFormatException will be throw.
E. It will not compile.

Answer: D

Explanation:

The Integer class value of 0 returns an Integer from given string. But we need to pass string which has correct format for integer otherwise it will throw a NumberFormatException. In this case we have passed string which is not an integer value (since what we passed is fractional number), so option D is correct.

Question No : 141

Given:

```
package p1;
public interface DoInterface {
    void m1(int n);                    // line n1
    public void m2(int n);
}

package p3;
import p1.DoInterace;
public class DoClass implements DoInterface{
    int x1,x2;
    DoClass(){
        this.x1 = 0;
        this.x2 = 10;
    }
    public void m1(int p1) { x1+=p1; System.out.println(x1); }   // line n2
    public void m2(int p1) { x2+=p1; System.out.println(x2); }
}

package p2;
import p1.*;
import p3.*;
class Test {
    public static void main(String[] args){
        DoInterface doi= new DoClass();          // line n3
        doi.method1(100);
        doi.method2(200);
    }
}
```

What is the result?

A. 100
210
B. Compilation fails due to an error in line n1
C. Compilation fails due to an error at line n2
D. Compilation fails due to an error at line n3

Answer: C

Question No : 142

Which two are benefits of polymorphism?

A. Faster code at runtime
B. More efficient code at runtime
C. More dynamic code at runtime
D. More flexible and reusable code
E. Code that is protected from extension by other classes

Answer: C,D

Question No : 143

Given:

```
class Product {
    double price;
}

public class Test {
    public void updatePrice(Product product, double price) {
        price = price * 2;
        product.price = product.price + price;
    }
    public static void main(String[] args) {
        Product prt = new Product();
        prt.price = 200;
        double newPrice = 100;

        Test t = new Test();
        t.updatePrice(prt, newPrice);
        System.out.println(prt.price + " : " + newPrice);
    }
}
```

What is the result?

A. 200.0 : 100.0
B. 400.0 : 200.0
C. 400.0 : 100.0
D. Compilation fails.

Answer: C

Question No : 144

Given:

class Base {

// insert code here

}

public class Derived extends Base{

public static void main(String[] args) {

Derived obj = new Derived();

obj.setNum(3);

System.out.println("Square = " + obj.getNum() * obj.getNum());

}

}

Which two options, when inserted independently inside class Base, ensure that the class is being properly encapsulated and allow the program to execute and print the square of the number?

A. private int num; public int getNum() { return num; }public void setNum(int num) { this.num = num;}
B. public int num; protected public int getNum() { return num; }protected public void setNum(int num) { this.num = num;}
C. private int num;public int getNum() {return num;} private void setNum(int num) { this.num = num;}
D. protected int num; public int getNum() { return num; } public void setNum(int num) { this.num = num;}
E. protected int num; private int getNum() { return num; } public void setNum(int num) { this.num = num;}

Answer: A,D

Explanation:

Incorrect:

Not B: illegal combination of modifiers: protected and public not C: setNum method cannot be private. not E: getNum method cannot be private.

Question No : 145

You are asked to develop a program for a shopping application, and you are given the following information:

- ☞ The application must contain the classes Toy, EduToy, and consToy. The Toy class is the superclass of the other two classes.
- ☞ The int caicuiatePrice (Toy t) method calculates the price of a toy.
- ☞ The void printToy (Toy t) method prints the details of a toy.

A)
```
public abstract class Toy{
        public abstract int calculatePrice(Toy t);
        public void printToy(Toy t) { /* code goes here */ }
    }
```

B)
```
public abstract class Toy {
        public int calculatePrice(Toy t) ;
        public void printToy(Toy t) ;
    }
```

C)
```
public abstract class Toy {
        public int calculatePrice(Toy t);
        public final void printToy(Toy t){ /* code goes here */ }
    }
```

D)
```
public abstract class Toy {
        public abstract int calculatePrice(Toy t) { /* code goes here */ }
        public abstract void printToy(Toy t) { /* code goes here */ }
    }
```

A. Option A
B. Option B
C. Option C
D. Option D
Answer: A

Question No : 146

Given the code fragment:

```
public static void main(String[] args) {
    List<String> names = new ArrayList<>();
    names.add("Robb");
    names.add("Bran");
    names.add("Rick");
    names.add("Bran");

    if (names.remove("Bran")) {
        names.remove("Jon");
    }
    System.out.println(names);
}
```

What is the result?

A. [Robb, Rick, Bran]
B. [Robb, Rick]
C. [Robb, Bran, Rick, Bran]
D. An exception is thrown at runtime.

Answer: A

Question No : 147

Given the code fragment:

```
for (int ii = 0; ii < 3;ii++) {

int count = 0;

for (int jj = 3; jj > 0; jj--) {

if (ii == jj) {
++count;

break;

}

}
System.out.print(count);

continue;

}
```

What is the result?

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

Answer: A

Question No : 148

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

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

Answer: C

Question No : 149

Given:

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

System.out.println("Base " + args[2]);

}

}

public class Sub extends Base{


public static void main(String[] args) {

System.out.println("Overriden " + args[1]);

}

}
```

And the commands:

javac Sub.java

java Sub 10 20 30

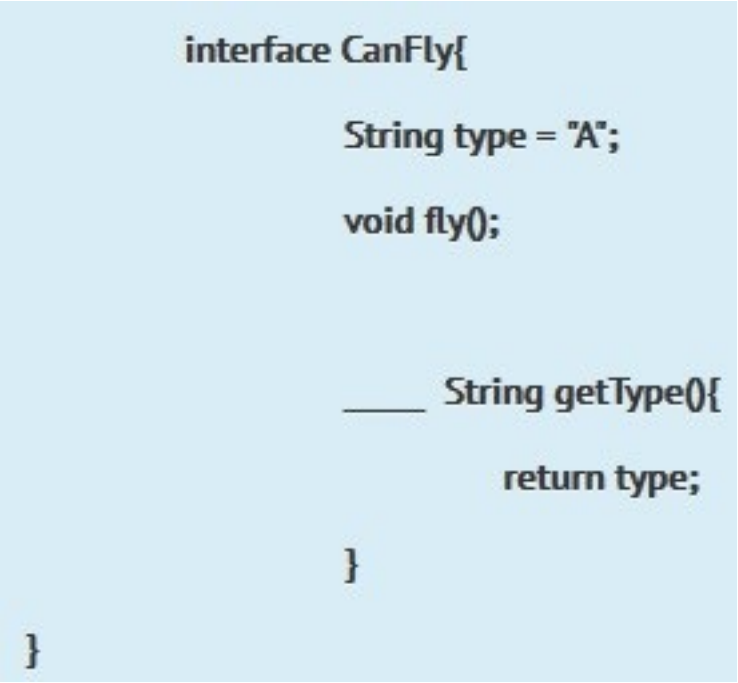
What is the result?

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

Answer: B

Question No : 150

Which of the following can fill in the blank in this code to make it compile?



A. abstract
B. public
C. default
D. It will not compile with any as interfaces cannot have non abstract methods.

E. It will compile without filling the blank.

Answer: C

Explanation:

From Java SE 8, we can use static and/or default methods in interfaces, but they should be non abstract methods. SO in this case using default in blank is completely legal. Hence option C is correct.

Option A is incorrect as given method is not abstract, so can't use abstract there.

Options B and E are incorrect as we can't have non abstract method interface if they are not default or static.

https;//docs.oraclexom/javase/tutorial/java/landl/defaultmethods.html

Question No : 151

Given the code fragment:

```
float x = 22.00f % 3.00f;

int y = 22 % 3;

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

What is the result?

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

Answer: A

Question No : 152

Given:

```
public class Access {
    private int x = 0;
    private int y = 0;

    public static void main(String[] args) {
        Access accApp = new Access();
        accApp.printThis(1, 2);
        accApp.printThat(3, 4);
    }

    public void printThis(int x, int y) {
        x = x;
        y = y;
        System.out.println("x:" + this.x + " y:" + this.y);
    }

    public void printThat(int x, int y) {
        this.x = x;
        this.y = y;
        System.out.println("x:" + this.x + " y:" + this.y);
    }
}
```

What is the result?

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

G. x: 0 y: 0
H. 0 y: 0

Answer: C

Question No : 153

Given the following class declarations:

- public abstract class Animal
- public interface Hunter
- public class Cat extends Animal implements Hunter
- public class Tiger extends Cat

Which answer fails to compile?

A)
```
ArrayList<Animal> myList = new ArrayList<>();
myList.add(new Tiger());
```

B)
```
ArrayList<Hunter> myList = new ArrayList<>();
myList.add(new Cat());
```

C)
```
ArrayList<Hunter> myList = new ArrayList<>();
myList.add(new Tiger());
```

D)
```
ArrayList<Tiger> myList = new ArrayList<>();
myList.add(new Cat());
```

E)
```
ArrayList<Animal> myList = new ArrayLIst<>();
myList.add(new Cat());
```

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

Answer: E

Explanation: Look at the right side of the declaration ArrayLIst() rather than ArrayList

Question No : 154

Given:

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

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

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

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

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

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

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

Answer: A

Question No : 155

Given:

```
public class Test {
public static void main(String[] args) {
int ax = 10, az = 30;
int aw = 1, ay = 1;
try {
aw = ax % 2;
ay = az / aw;
} catch (ArithmeticException e1) {
System.out.println("Invalid Divisor");
} catch (Exception e2) {
aw = 1;
System.out.println("Divisor Changed");
}
ay = az /aw; // Line 14
System.out.println("Succesful Division " + ay);

}
}
```

What is the result?

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

Answer: C

Question No : 156

Given:

```
1. public class Whizlabs {
2.
3.          public static void main(String[] args) {
4.                    String s = "A";
5.
6.                    switch (s) {
7.                    case "a":
8.                              System.out.print("simaple A ");
9.                    default:
10.                             System.out.print("default ");
11.                   case "A":
12.                             System.out.print("Capital A ");
13.                   }
14.         }
15. }
```

What is the result?

A. simaple A
B. Capital A
C. simaple A default Capital A **D.** simaple A default
E. Compilation fails.

Answer: C

Explanation:

Here we have to use two ternary operators combined. SO first we can use to check first condition which is x > 10, as follows; x>10?">": (when condition false) Now we have to use another to check if x<10 as follows; x<10?V:"=" We can combine these two by putting last ternary statement in the false position of first ternary statement as follows; x>10?">":x<10?'<':"=" https;//docs.oraclexom/javase/tutorial/java/nutsandbolts/if.html

Question No : 157

Given:

```
1. public class Whizlabs{
2.
3.          public static void main(String[] args){
4.                    try{
5.                              Double number = Double.valueOf("120D");
6.                    }catch(NumberFormatException ex){
7.                    }
8.                    System.out.println(number);
9.          }
10. }
```

What is the result?

A. 120
B. 120D
C. A NumberFormatException will be thrown.
D. Compilation fails due to error at line 5.
E. Compilation tails due to error at line 8.

Answer: E

Explanation:

At line 5, we have created a wrapper object of double by passing 120D, which is convertible to a Double, so there won't be any exception there. But if you check carefully, you can see the variable number is declared inside try block, so the scope of the variable number is limited to that block, so trying to access it outside causes a compile time error.

httpsy/docs.oracle.com/javase/tutorial/iava/nutsandbolts/variables.html

Question No : 158

Given the code fragment:

```
class Student {
int rollnumber;
String name;
List cources = new ArrayList();
// insert code here
public String toString() {
return rollnumber + " : " + name + " : " + cources;
}
}
```

And,

```
public class Test {
public static void main(String[] args) {
List cs = newArrayList();
cs.add("Java");
cs.add("C");
Student s = new Student(123,"Fred", cs);
```

```
System.out.println(s);
}

}
```

Which code fragment, when inserted at line // insert code here, enables class Test to print 123 : Fred : [Java, C]?

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

Answer: C

Explanation:

Incorrect:

Not A: Student has private access line: Student s = new Student(123,"Fred", cs);

Not D: Cannot be applied to given types. Line: Student s = new Student(123,"Fred", cs);

Question No : 159

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 {

}
```

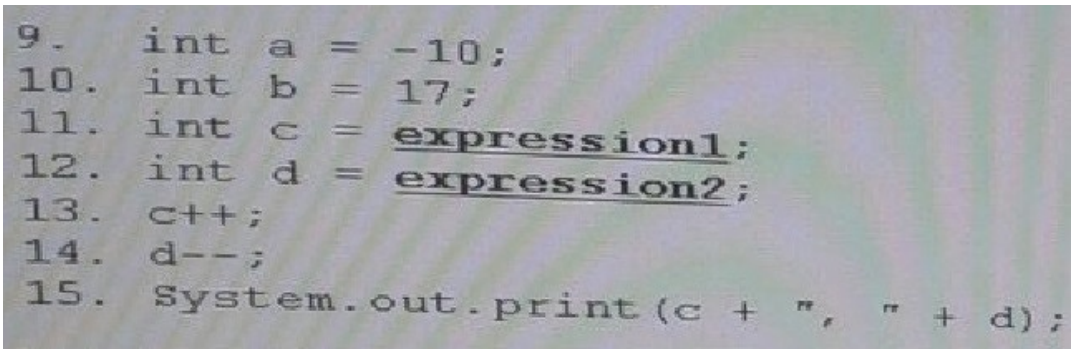
What is the result?

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

Answer: D

Question No : 160

Given the code fragment:



What could **expression1** and **expression2** be, respectively, in order to produce output –8, 16?

A. + +a, - -b
B. + +a, b- - **C.** A+ +, - - b
D. A + +, b - -
Answer: D

Question No : 161

Given the code fragment:

```
public static void main(String[] args) {
    Short s1 = 200;
    Integer s2 = 400;
    Long s3 = (long) s1 + s2;          //line n1
    String s4 = (String) (s3 * s2);    //line n2
    System.out.println("Sum is " + s4);
}
```

What is the result?

A. Sum is 600
B. Compilation fails at line n1.
C. Compilation fails at line n2.
D. A ClassCastException is thrown at line n1.
E. A ClassCastException is thrown at line n2.

Answer: C

Question No : 162

Given the following four Java file definitions:

// Foo.java

package facades;

public interface Foo { }

```
// Boo.java

package facades;

public interface Boo extends Foo { }
// Woofy.java

package org.domain

// line n1

public class Woofy implements Boo, Foo { }


// Test.java

package.org;

public class Test {

public static void main(String[] args) {

Foo obj=new Woofy();
```

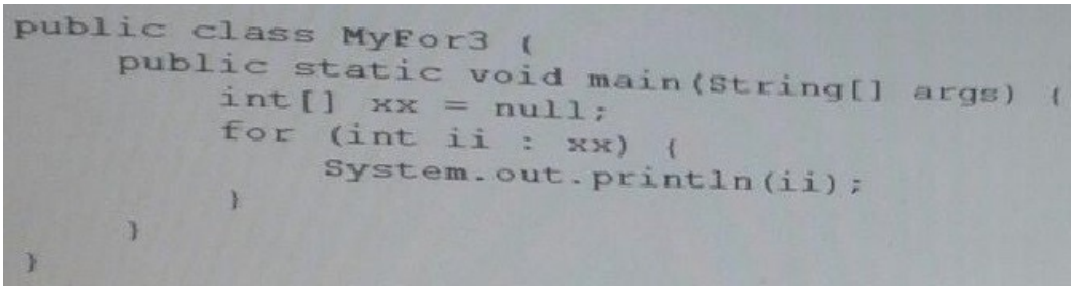
Which set modifications enable the code to compile and run?

A. At line n1, Insert: import facades;At line n2, insert:import facades;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;

Answer: E

Question No : 163

Given:



```
public class MyFor3 {
    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

Answer: C

Question No : 164

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

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

Answer: D

Question No : 165

Given:

```
1.    public class Whizlabs {
2.       public static void main(String[] args) {
3.                int sum = 0;
4.
5.                for(int x = 0;x<=10;x++)
6.                         sum += x;
7.                System.out.print("Sum for 0 to " + x);
8.                System.out.println(" = " + sum);
9.       }
10.  }
```

Which is true?

A. Sum for 0 to 0 = 55
B. Sum for 0 to 10 = 55
C. Compilation fails due to error on line 6.
D. Compilation fails due to error on line 7.
E. An Exception is thrown at the runtime.

Answer: D

Explanation:

Loop variables scope limited to that enclosing loop. So in this case, the scope of the loop variable x declared at line 5, limited to that for loop. Trying to access that variable at line 7, which is out of scope of the variable x, causes a compile time error. So compilation fails due to error at line 7. Hence option D is correct.

Options A and B are incorrect, since code fails to compile.

Reference: httpsy/docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

Question No : 166

Given:

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

What is the result?

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

Answer: E

Question No : 167

Given the following main method:

```
public static void main(String[] args) {
     int num = 5;
     do {
          System.out.print(num-- +" ");
     } while(num == 0);
}
```

What is the result?

A. 5 4 3 2 1 0
B. 5 4 3 2 1

C. 4 2 1
D. 5
E. Nothing is printed

Answer: D

Explanation:

Loop will run only once and after that num == 0 will break it After first cycle of the loop.

Question No : 168

Given the definitions of the MyString class and the Test class:

MyString.java:

```
package p1;
class MyString {
    String msg;
    MyString(String msg) {
        this.msg = msg;
    }
}
```

Test.java:

```
package p1;
public class Test {
    public static void main(String[] args) {
        System.out.println("Hello " + new StringBuilder("Java SE 8"));
        System.out.println("Hello " + new MyString("Java SE 8"));
    }
}
```

What is the result?

A) Hello Java SE 8
Hello Java SE 8

B) Hello java.lang.StringBuilder@<<hashcode1>>
Hello p1.MyString@<<hashcode2>>

C) Hello Java SE 8
Hello p1.MyString@<<hashcode>>

D) Compilation fails at the Test class.

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

Answer: C

Question No : 169

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

Answer: C

Question No : 170

Given:

```
public class MyClass {

public static void main(String[] args) {

String s = " Java Duke ";

int len = s.trim().length();

System.out.print(len);

}

}
```

What is the result?

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

Answer: B

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

Question No : 171

Given:

```
public class App {

    public static void main(String[] args) {
        Boolean[] bool = new Boolean[2];

        bool[0] = new Boolean(Boolean.parseBoolean("true"));
        bool[1] = new Boolean(null);

        System.out.println(bool[0] + " " + bool[1]);
    }
}
```

What is the result?

A. True false
B. True null
C. Compilation fails
D. A NullPointerException is thrown at runtime

Answer: A

Question No : 172

Which statement is/are true?

I. Default constructor only contains "super();" call.

II. We can't use any access modifier with a constructor.

III. A constructor should not have a return type.

A. Only I.
B. Only II.
C. Only I and II.
D. Only I and III.
E. AIL

Answer: D

Explanation:

Statement I is correct as the default constructor only contains super0 call

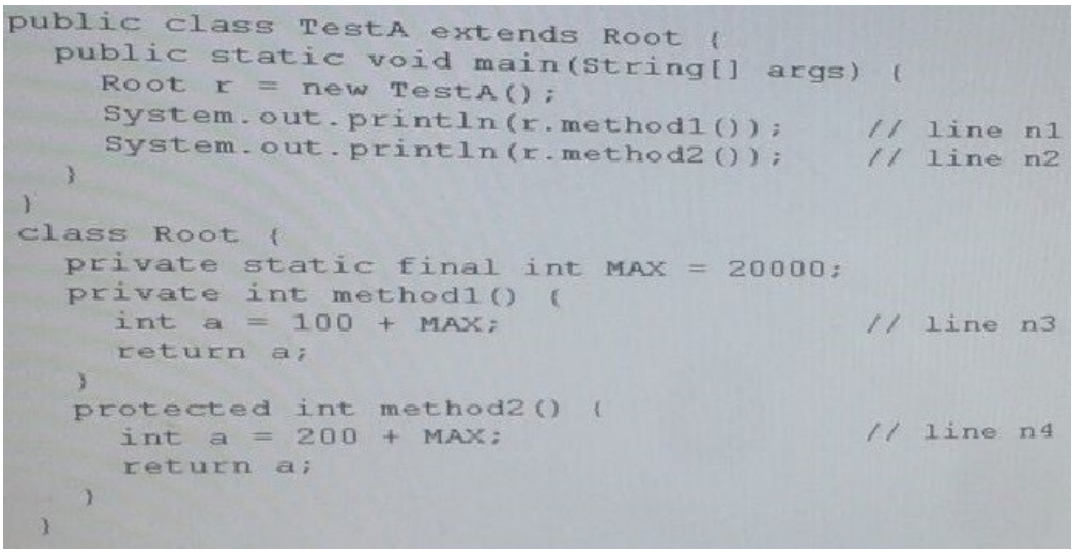
Statement II is incorrect as we can use any access modifier with a constructor.

Statement III is correct as constructor can't have return type, even void.

So option D is correct.
httpsy/docs.oracle.com/javase/tutorial/iava/javaOO/construaors.html

Question No : 173

Given the fragments:



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

Which line causes a compilation error?

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

Answer: A

Question No : 174

Consider following method

```
default void print(){

                    }
```

Which statement is true?

A. This method is invalid.
B. This method can be used only in an interface.
C. This method can return anything.
D. This method can be used only in an interface or an abstract class.
E. None of above.

Answer: B

Explanation:

Given method is declared as default method so we can use it only inside an interface.

Hence option B is correct and option D is incorrect.

Option A is incorrect as it is valid method. Option C is incorrect as return type is void, which means we can't return anything.

Question No : 175

Given:

```
package p1;

public class Test {

static double dvalue;

static Test ref;
```

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

System.out.println(ref);
System.out.println(dvalue);

}

}
```

What is the result?

A. p1.Test.class
0.0
B. <the summary address refrenced by ref>
0.000000
C. Null
0.0
D. Compilation fails
E. A NullPointerException is thrown at runtime

Answer: C

Question No : 176

Given:

```
public class App {

    String myStr = "7007";

    public void doStuff(String str) {
        int myNum = 0;
        try {
            String myStr = str;
            myNum = Integer.parseInt(myStr);
        } catch (NumberFormatException ne) {
            System.err.println("Error");
        }
        System.out.println(
            "myStr: " + myStr + ", myNum: " + myNum);
    }

    public static void main(String[] args) {
        App obj = new App();
        obj.doStuff("9009");
    }
}
```

What is the result?

A. myStr: 9009, myNum: 9009
B. myStr: 7007, myNum: 7007
C. myStr: 7007, myNum: 9009
D. Compilation fails

Answer: C

Question No : 177

int i, j=0;

i.= (3* 2 +4 +5) ;

j.= (3 * ((2+4) + 5));

System.out.println("i:"+ i + "\nj":+j);

What is the result?

A. i: 16
j: 33
B. i: 15
j: 33
C. i: 33
j: 23
D. i: 15
j: 23

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

Answer: B

Question No : 178

Given:

interface Pet { }

class Dog implements Pet { }

public class Beagle extends Dog{ }

Which three are valid?
注意接口，不可以实例化。

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();

Answer: A,D,E

Explanation:

Incorrect:

Not B, not C: Pet is abstact, cannot be instantiated.

Not F: incompatible type. Required Beagle, found Dog.

Question No : 179

Given:

Test.java

```
public class Test {
    public static void main(String[] args) {
        Integer num = Integer.parseInt(args[1]);
        System.out.println("Number is : " + num);
    }
}
```

And the commands:

Javac Test.java

Java Test 12345

What is the result?

A. Number us : 12345
B. A NullPointerException is thrown at runtime
C. A NumberFormatException is thrown at runtime
D. AnArrayIndexOutOfBoundException is thrown at runtime.

Answer: A

Question No : 180

Given:

```
public class ColorTest {

public static void main(String[] args) {

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

int count = 0;

for (String c : colors) {

if (count >= 4) {

break;

}

else {

continue;

}

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

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

}

count++;

}

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

}

}
```

What is the result?
if else 中的 break 和 continue

A. Yellow
B. Maroon
C. Compilation fails

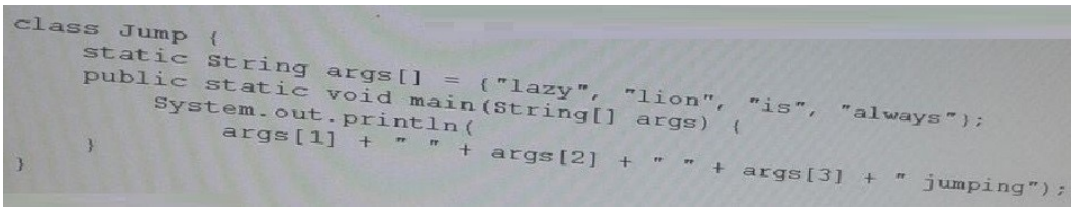
D. A StringIndexOutOfBoundsException is thrown at runtime.

Answer: C

Explanation:

Question No : 181

Given:



```
class Jump {
    static String args[] = {"lazy", "lion", "is", "always"};
    public static void main(String[] args) {
        System.out.println(
            args[1] + " " + args[2] + " " + args[3] + " jumping");
    }
}
```

And the commands:

Javac Jump.java
Java Jump crazy elephant is always

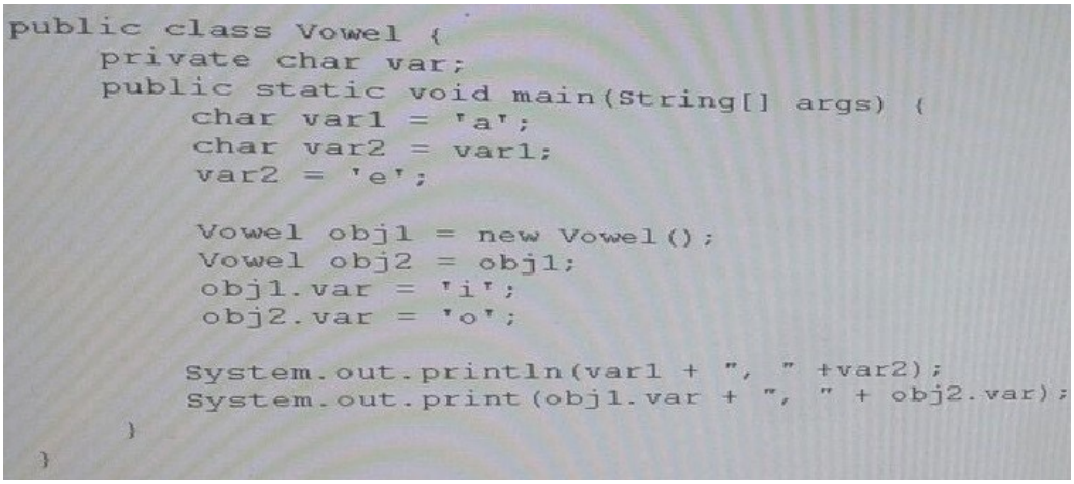
What is the result?

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

Answer: B

Question No : 182

Given:



```
public class Vowel {
    private char var;
    public static void main(String[] args) {
        char var1 = 'a';
        char var2 = var1;
        var2 = 'e';

        Vowel obj1 = new Vowel();
        Vowel obj2 = obj1;
        obj1.var = 'i';
        obj2.var = 'o';

        System.out.println(var1 + ", " +var2);
        System.out.print(obj1.var + ", " + obj2.var);
    }
}
```

A. a, e
i, o **B.**
a, e o,
o **C.** e,
e l, o
D. e, e
o, o

Answer: B

Question No : 183

Which code fragment cause a compilation error?

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

Answer: B

Question No : 184

Given:

```
class MarksOutOfBoundsException extends IndexOutOfBoundsException { }

public class GradingProcess {

void verify(int marks) throws IndexOutOfBoundsException {

if (marks > 100) {

throw new MarksOutOfBoundsException();

}

if (marks > 50) {

System.out.print("Pass");

} else {

System.out.print("Fail");

}
```

```
}

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

try {

new GradingProcess().verify(marks));

} catch(Exception e) {

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

}

}

}
```

And the command line invocation:

Java grading process 89 50 104

What is the result?

A. Pass
B. Fail
C. Class MarketOutOfBoundsException
D. Class IndexOutOfBoundsException
E. Class Exception

Answer: C
Explanation: The value 104 will cause a MarketOutOfBoundsException

Question No : 185

Given:

```
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

Answer: C

Question No : 186

Given:

```
public class Test2 {
    public static void doChange(int[] arr) {
        for(int pos = 0; pos < arr.length; pos++){
            arr[pos] = arr[pos] + 1;
        }
    }
    public static void main(String[] args) {
        int[] arr = {10, 20, 30};
        doChange(arr);
        for(int x: arr) {
            System.out.print(x + ", ");
        }
        doChange(arr[0], arr[1], arr[2]);
        System.out.print(arr[0] + ", " + arr[1] + ", " + arr[2]);
    }
}
```

What is the result?

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

Answer: D

Question No : 187

What is the name of the Java concept that uses access modifiers to protect variables and hide them within a class?

A. Encapsulation
B. Inheritance
C. Abstraction
D. Instantiation
E. Polymorphism

Answer: A

Question No : 188

int [] array = {1,2,3,4,5};

for (int i: array) {

if (i < 2) {

```
keyword1 ;

}

System.out.println(i);

if ( i == 3) {

keyword2 ;

}}
```

What should keyword1 and keyword2 be respectively, in oreder to produce output 2345?

A. continue, break **B.**
break, break
C. break, continue
D. continue, continue

Answer: D

Question No : 189

Given the code fragments:

Person.java:

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

    public Person(String n, int a) {
        name = n;
        age = a;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }
}
```

Test.java:

```
public static void checkAge(List<Person> list, Predicate<Person> predicate) {
    for (Person p : list) {
        if (predicate.test(p)) {
            System.out.println(p.name + " ");
        }
    }
}

public static void main(String[] args) {
    List<Person> iList = Arrays.asList(new Person("Hank", 45),
                                       new Person("Charlie", 40),
                                       new Person("Smith", 38));
    //line n1
}
```

Which code fragment, when inserted at line ni, enables the code to print Hank?

A. checkAge (iList, ( ) -> p. get Age ( ) > 40);
B. checkAge(iList, Person p -> p.getAge( ) > 40);
C. checkAge (iList, p -> p.getAge ( ) > 40);
D. checkAge(iList, (Person p) -> { p.getAge() > 40; });
Answer: C

Question No : 190

Given:

```
package p1;
public class Acc {
    int p;
    private int q;
    protected int r;
    public int s;
}
```

Test.java:

```
package p2;
import p1.Acc;
public class Test extends Acc {
    public static void main(String[] args) {
        Acc obj = new Test();
    }
}
```

Which statement is true?

A. Both p and s are accessible by obj.
B. Only s is accessible by obj.
C. Both r and s are accessible by obj.
D. p, r, and s are accessible by obj.

Answer: B

Question No : 191

Given:

public class TestLoop {

public static void main(String[] args) {

int array[] = {0, 1, 2, 3, 4};

int key = 3;

```
for (int pos = 0; pos < array.length; ++pos) {

if (array[pos] == key) {

break;

}

}

System.out.print("Found " + key + "at " + pos);


}

}
```

What is the result?

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

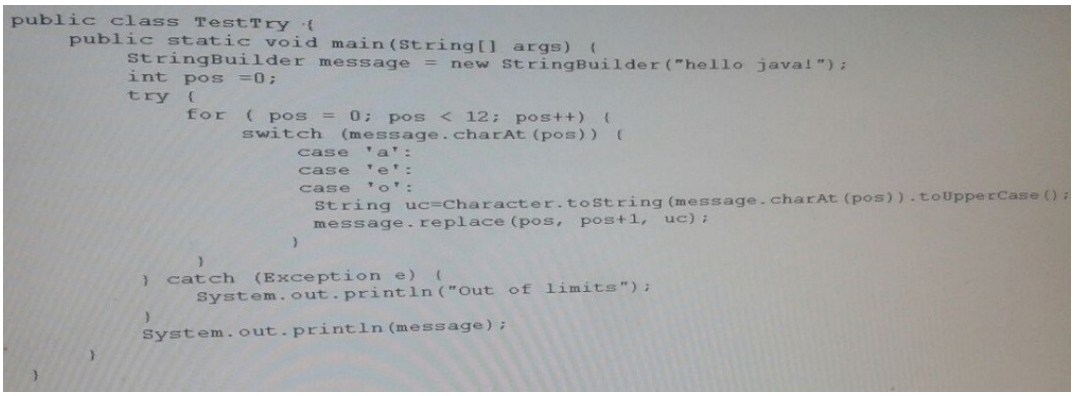
Answer: C

Explanation: The following line does not compile:
System.out.print("Found " + key + "at " + pos);

The variable pos is undefined at this line, as its scope is only valid in the for loop.
Any variables created inside of a loop are LOCAL TO THE LOOP.

Question No : 192

Given:



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

What is the result?

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

Answer: C

Question No : 193

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

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

~~Answer: C,D~~

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

Question No : 194

Given:

```
public class Natural {

private int i;

void disp() {

while (i <= 5) {

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

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

i++;

}

i++;

}

}

public static void main(String[] args) {

new Natural().disp();

}

}
```

What is the result?

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

Answer: D
Explanation: 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5

Question No : 195

Given the code fragment

```
int var1 = -5;

int var2 = var1--;

int var3 = 0;

if (var2 < 0) {

var3 = var2++;

} else {

var3 = --var2;

}

System.out.println(var3);
```

What is the result?

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

Answer: C

Question No : 196

Given:

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

Which two code fragments are valid?

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

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

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

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

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

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

~~Answer: B,C~~

Explanation: When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class (C). However, if it does not, 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 : 197

Given the following code for the classes MyException and Test:

```
public class MyException extends RuntimeException {}

public class Test {
    public static void main(String[] args) {
       try {
           method1();
       }
       catch (MyException ne) {
           System.out.print("A");
       }
    }
    public static void method1() { // line n1
        try {
           throw Math.random() > 0.5 ?new MyException() :new RuntimeException();
       }
       catch (RuntimeException re) {
           System.out.print("B");
       }
    }
}
```

What is the result?

A. A
B. B
C. Either A or B
D. A B
E. A compile time error occurs at line n1

Answer: B

Question No : 198

Given the following code:

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

    System.out.println(planets.length);
    System.out.println(planets[1].length());
}
```

What is the output?

A. 4
4

B. 5 : 5 2
C. bo 3
D. nb 3

7
4
E. 4
5
F. 4
21

Answer: E

Question No : 199

Which of the following data types will allow the following code snippet to compile?

```
float i = 4;

float j = 2;

_____ z = i + j;
```

A. long
B. double
C. int
D. float
E. byte

Answer: B,D

Explanation:

Option B and D are the correct answer.

Since the variables I and j are floats, resultant will be float type too. So we have to use float or primitive type which can hold float, such a primitive type is double, it has wider range and also can hold floating point numbers, hence we can use double or float for the blank.

As explained above options B and D are correct.

long and int can't be used with floating point numbers so option A is incorrect. Option E is incorrect as it have smaller range and also can't be used with floating point numbers.

hnpsy/docs.oracle.com/javase/tutorial/java/javaOO/variables.html

Question No : 200

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 { }

Answer: A,E

Explanation: extends is for extending a class.

implements is for implementing an interface. Java allows for a class to implement many interfaces.

Question No : 201

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

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

Answer: B,D Reference:
http://docs.oracle.com/j
avase/tutorial/java/java
OO/methods.html

Question No : 202

Given the code fragment:

System.out.println(2 + 4 * 9 - 3); //Line 21

System.out.println((2 + 4) * 9 - 3); // Line 22

System.out.println(2 + (4 * 9) - 3); // Line 23

System.out.println(2 + 4 * (9 - 3)); // Line 24

System.out.println((2 + 4 * 9) - 3); // Line 25

Which line of codes prints the highest number?

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

Answer: B

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

Question No : 203

Given:

```
public class MainMethod {

void main() {

System.out.println("one");

}

static void main(String args) {

System.out.println("two");

}

public static void main(String[] args) {

System.out.println("three");

}

void mina(Object[] args) {

System.out.println("four");

}

}
```

What is printed out when the program is excuted?

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

Answer: C

Question No : 204

Given the code fragment:

```
public static void main(String[] args) {
    String[][] arr = {{"A", "B", "C"}, {"D", "E"}};
    for (int i = 0; i < arr.length; i++) {
        for (int j = 0; j < arr[i].length; j++) {
            System.out.print(arr[i][j] + " ");
            if (arr[i][j].equals("B")) {
                break;
            }
        }
        continue;
    }
}
```

What is the result?

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

Answer: C

Question No : 205

Given the code fragment:

List colors = new ArrayList();

colors.add("green");

colors.add("red");

colors.add("blue");

colors.add("yellow");

colors.remove(2);

colors.add(3,"cyan");

System.out.print(colors);

What is the result?

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

Answer: A

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

Question No : 206

Given the code fragment:

```
 3. public static void main(String[] args) {
 4.       int x = 5;
 5.       while (isAvailable(x)) {
 6.           System.out.print(x);
 7.
 8.       }
 9. }
10.
11. public static boolean isAvailable(int x) {
12.     return x-- > 0 ? true : false;
13. }
```

Which modification enables the code to print 54321?

A. Replace line 6 with System, out. print (--x) ;
B. At line 7, insert x --;
C. Replace line 6 with --x; and, at line 7, insert system, out. print (x);
D. Replace line 12 With return (x > 0) ? false: true;

Answer: B

Question No : 207

Given:

```
public class SumTest {

    public static void doSum(Integer x, Integer y) {
        System.out.println("Integer sum is " + (x + y));
    }

    public static void doSum(double x, double y) {
        System.out.println("double sum is " + (x + y));
    }

    public static void doSum(float x, float y) {
        System.out.println("float sum is " + (x + y));
    }

    public static void doSum(int x, int y) {
        System.out.println("int sum is " + (x + y));
    }

    public static void main(String[] args) {
        doSum(10, 20);
        doSum(10.0, 20.0);
    }
}
```

What is the result?

A)
```
int sum is 30
float sum is 30.0
```

B)
```
int sum is 30
double sum is 30
```

C)
```
Integer sum is 30
double sum is 30.0
```

D)
```
Integer sum is 30
float sum is 30.0
```

A. Option A

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

Answer: B

Question No : 208

Given the code fragment:

```
String[] strs = new String[2];
int idx = 0;
for (String s : strs) {
      strs[idx].concat(" element " + idx);
      idx++;
}
for (idx = 0; idx < strs.length; idx++) {
      System.out.println(strs[idx]);
}
```

What is the result?

A. Element 0
Element 1
B. Null element 0
Null element 1
C. Null
Null
D. A NullPointerException is thrown at runtime.

Answer: D

Question No : 209

public class StringReplace {

public static void main(String[] args) {

String message = "Hi everyone!"; System.out.println("message = " + message.replace("e", "X")); }

}

What is the result?

A. message = Hi everyone!
B. message = Hi XvXryonX!
C. A compile time error is produced.
D. A runtime error is produced.
E. message =
F. message = Hi Xveryone!

Answer: B

Question No : 210

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) {

Answer: A,D

Explanation:

Incorrect:
Not B: No main class found. Not
C: Main method not found not E:
Main method is not static.

Question No : 211

Given:

```
interface Readable {
    public void readBook();
    public void setBookMark();
}

abstract class Book implements Readable {    // line n1
    public void readBook() { }
    // line n2
}

class EBook extends Book {                   // line n3
    public void readBook() { }
    // line n4
}
```

Which option enables the code to compile?

A) Replace the code fragment at line n1 with:
```
class Book implements Readable {
```

B) At line n2 insert:
```
public abstract void setBookMark();
```

C) Replace the code fragment at line n3 with:
```
abstract class EBook extends Book {
```

D) At line n4 insert:
```
public void setBookMark() { }
```

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

Answer: C,D

Question No : 212

Given the code fragment:

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

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

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

Answer: A

Question No : 213

Given the code fragment:

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

int counter =0;

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

counter++;

} else if ("tom".equals(cartoons[1])) {

counter++;

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

counter++;

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

counter++;

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

What is the result?

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

Answer: A

Explanation: Counter++ will be executed only once because of the else if constructs.

Question No : 214

Given:

```
1.    public class Whizlabs{
2.       private String name;
3.       private boolean pass;
4.
5.       public static void main(String[] args) {
6.                Whizlabs wb = new Whizlabs();
7.                System. out.print("name = " + wb.name);
8.                System. out.print(", pass = " + wb.pass);
9.       }
10.  }
```

What would be the output, if it is executed as a program?

A. name =, pass =
B. name = null, pass = null
C. name = null, pass = false **D.** name = null pass = true
E. Compile error.

Answer: C

Explanation:

Both name and pass variables are instance variables, and we haven't given them any values, so they take their default values. For Boolean default value is false and for string which is not a primitive type default is null So at line 7, null will printed as the value of the variable name, and at line 8 false will be printed. Hence Option C is correct.

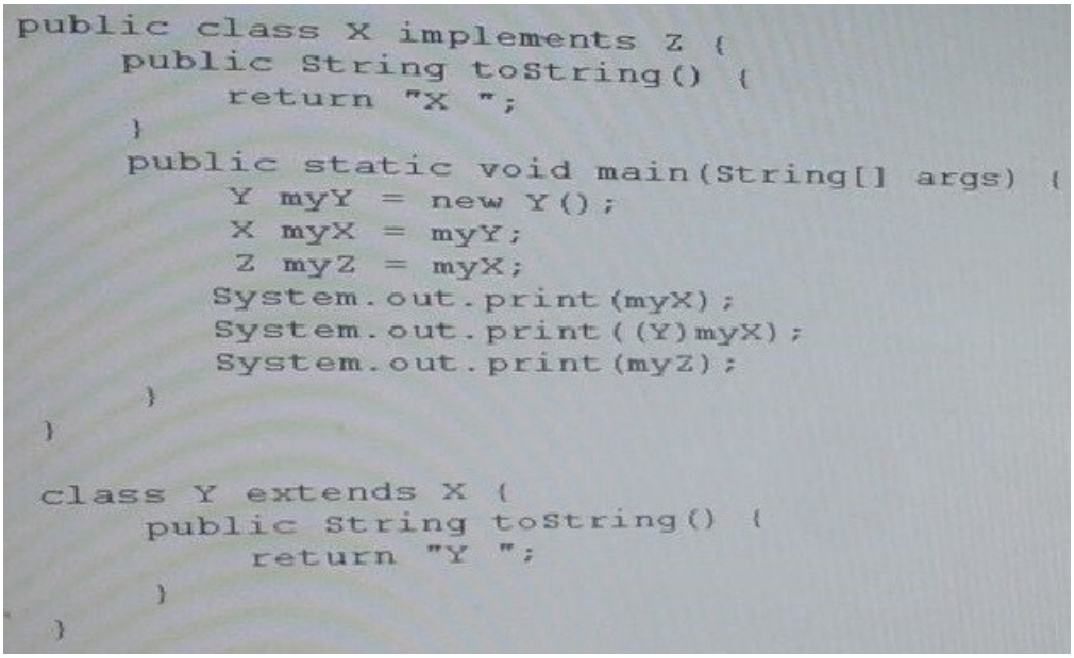
As explained above options A, B and D are incorrect.

Code compiles fine so option E is incorrect.
Reference:

https://docs.oracle.com/javaseAutorial/java/javaOOAariables.html

Question No : 215

Given:



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

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

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

Answer: D

Question No : 216

Which of the following can fill in the blank in this code to make it compile?

```
public class Exam {
        void method() { }
}

public class OCAJP extends Exam{
 _____ void method() { }
}
```

A. abstract
B. final
C. private
D. default
E. int

Answer: C

Explanation:

From Java SE 8, we can use static and/or default methods in interfaces, but they should be non abstract methods. SO in this case using default in blank is completely legal. Hence option C is correct.

Option A is incorrect as given method is not abstract, so can't use abstract there.

Options B and E are incorrect as we can't have non abstract method interface if they are not default or static.

httpsy/docs.oracle.com/javase/tutorial/iava/landl/defaultmethods.html

Question No : 217

Given the code fragment:

```
int x = 100;
int a = x++;
int b = ++x;
int c = x++;
int d = (a < b) ? (a < c) ? a: (b <c )? b: c;
System.out.println(d);
```

What is the result?

A. 100
B. 101
C. 102
D. 103
E. Compilation fails

Answer: E

Question No : 218

Given:

```
package p1;

public interface DoInterface {

void method1(int n1); // line n1

}

package p3;

import p1.DoInterface;
public class DoClass implements DoInterface {

public DoClass(int p1) { }

public void method1(int p1) { } // line n2

private void method2(int p1) { } // line n3
```

```
}

public class Test {

public static void main(String[] args) {

DoInterface doi= new DoClass(100); // line n4

doi.method1(100);

doi.method2(100);

}

}
```

Which change will enable the code to compile?

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

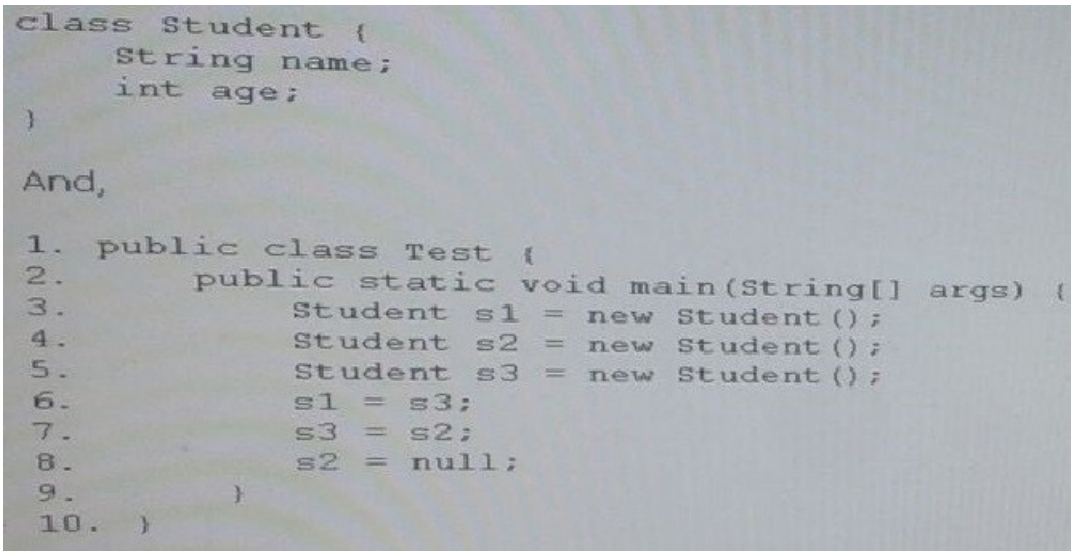
Answer: C

Explanation: Private members (both fields and methods) are only accessible inside the class they are declared or inside inner classes. private keyword is one of four access modifier provided by Java and its a most restrictive among all four e.g. public, default(package), protected and private.

Read more: http://javarevisited.blogspot.com/2012/03/private-in-java-why-should-youalways.html#ixzz3Sh3mOc4D

Question No : 219

Given the code fragment:



```
class Student {
    String name;
    int age;
}

And,

1. public class Test {
2.     public static void main(String[] args) {
3.         Student s1 = new Student();
4.         Student s2 = new Student();
5.         Student s3 = new Student();
6.         s1 = s3;
7.         s3 = s2;
8.         s2 = null;
9.     }
10. }
```

Which statement is true?

A. After line 8, three objects are eligible for garbage collection
B. After line 8, two objects are eligible for garbage collection
C. After line 8, one object is eligible for garbage collection
D. After line 8, none of the objects are eligible for garbage collection

Answer: C

Question No : 220

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

Answer: C

Explanation: 1200 Strawberry

1230 Chocolate

Question No : 221

Which of the following exception will be thrown due to the statement given here?

int array[] = new int[-2];

A. NullPointerException
B. NegativeArraySizeException
C. ArrayIndexOutOfBoundsException
D. IndexOutOfBoundsException
E. This statement does not cause any exception.

Answer: B

Explanation:

In given statement we can see that, we have passed negative value for creating int array, which results a NegativeArraySize Except ion. Hence option B is correct.

Option A is incorrect as it is thrown when an application attempts to use null in a case where an object is required.

Option D is incorrect as IndexOutOfBoundsException thrown to indicate that an index of some sort (such as to an array, to a string, or to a vector) is out of range.

REFERENCE

rhttpy/docs.oracle.com/iavase/S/docs/api/java/lang/NegativeArraySizeException.html

Question No : 222

Given:

```
class Vehicle {
    String type = "4W";
    int maxSpeed = 100;

    Vehicle(String type, int maxSpeed) {
        this.type = type;
        this.maxSpeed = maxSpeed;
    }
}

class Car extends Vehicle {
    String trans;

    Car(String trans) {          //line n1
        this.trans = trans;
    }

    Car(String type, int maxSpeed, String trans) {
        super(type, maxSpeed);
        this(trans);             //line n2
    }
}
```

And given the code fragment:

```
 7. Car c1 = new Car("Auto");
 8. Car c2 = new Car("4W", 150, "Manual");
 9. System.out.println(c1.type + " " + c1.maxSpeed + " " + c1.trans);
10. System.out.println(c2.type + " " + c2.maxSpeed + " " + c2.trans);
```

What is the result?

A. 4W 100 Auto
4W 150 Manual **B.**
Null 0 Auto
4W 150 Manual
C. Compilation fails only at line n1
D. Compilation fails only at line n2
E. Compilation fails at both line n1 and line n2

Answer: E

Explanation:

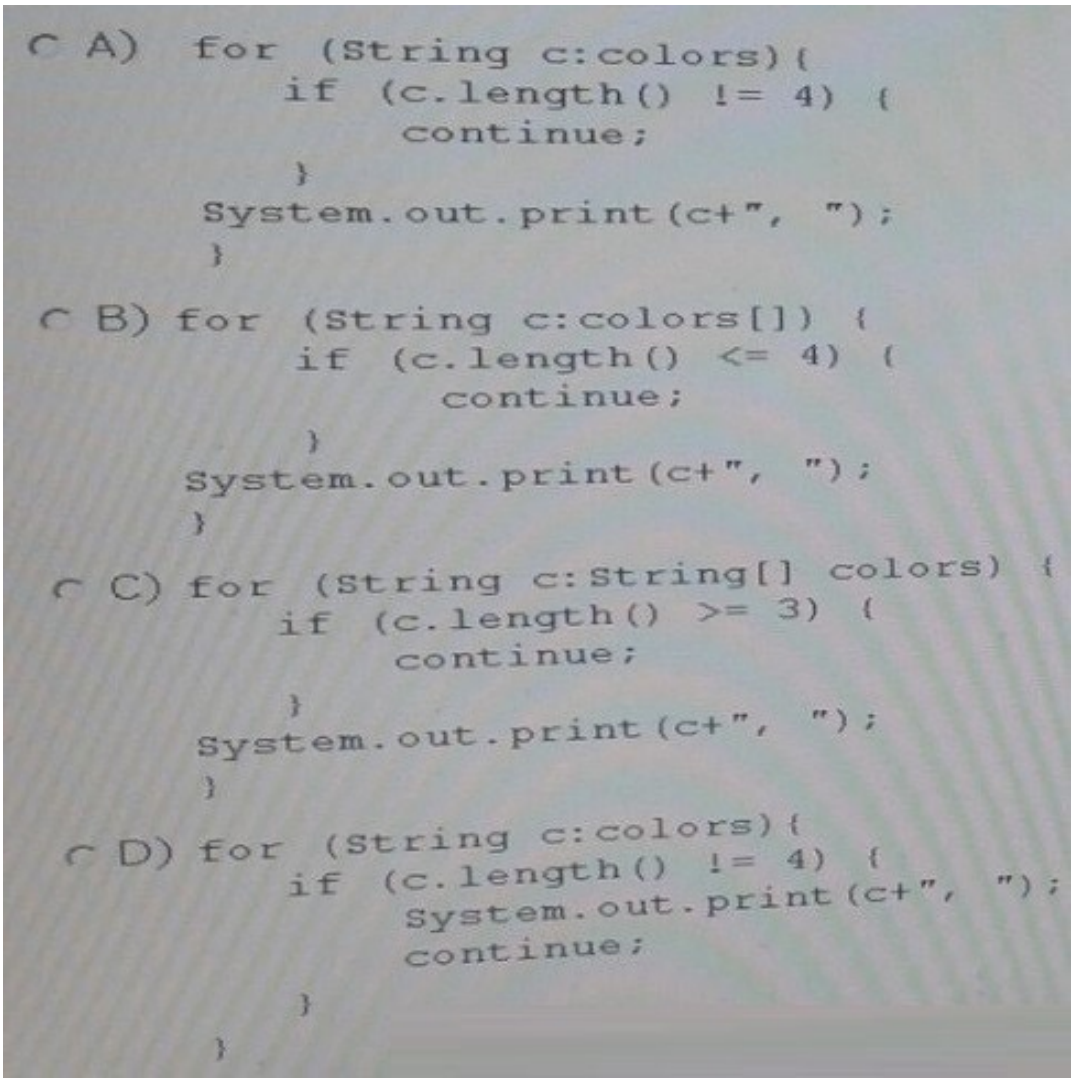
On line n1 implicit call to parameterized constructor is missing and n2 this() must be the first line.

Question No : 223

Given the code fragment:

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

Which code fragment prints blue, cyan, ?



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

Answer: A

Question No : 224

Given the code fragment:

```
int num[][] = new int[1][3];
for (int i = 0; i < num.length; i++) {
    for (int j = 0; j < num[i].length; j++) {
        num[i][j] = 10;
    }
}
```

Which option represents the state of the num array after successful completion of the outer loop?

A)
```
num[0][0]=10
num[0][1]=10
num[0][2]=10
```

B)
```
num[0][0]=10
num[1][0]=10
num[2][0]=10
```

C)
```
num[0][0]=10
num[0][1]=0
num[0][2]=0
```

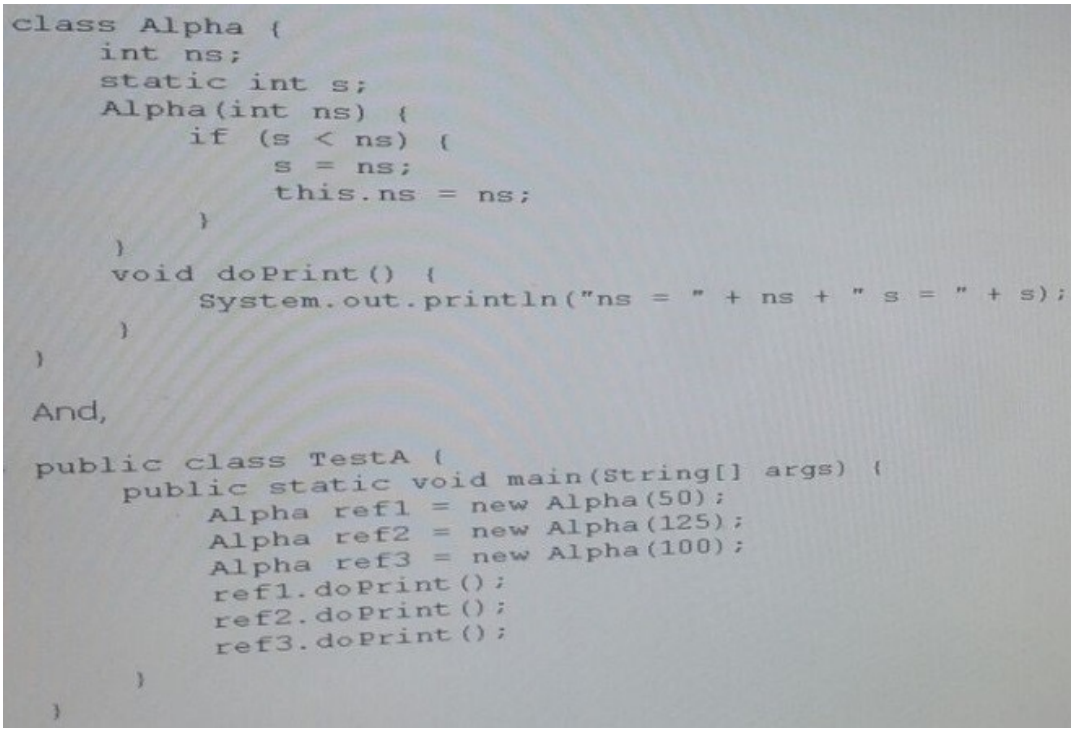
D)
```
num[0][0]=10
num[0][1]=10
num[0][2]=10
num[0][3]=10
num[1][0]=0
num[1][1]=0
num[1][2]=0
num[1][3]=0
```

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

Answer: A

Question No : 225

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();
    }
}
```

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 **Answer: B**

Question No : 226

Given:

```
public 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

Answer: B

Question No : 227

Given:

```
public class Test2 {
    public static void main(String[] args) {
        int ar1[] = {2, 4, 6, 8};
        int ar2[] = {1, 3, 5, 7, 9};
        ar2 = ar1;
        for (int e2 : ar2) {
            System.out.print(" " + e2);
        }
    }
}
```

What is the result?

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

Answer: D

Question No : 228

Given:

```
class Sports {

int num_players;

String name, ground_condition;

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

num_players = np;

name = sname;

ground_condition = sground;

}

}

class Cricket extends Sports {

int num_umpires;

int num_substitutes;
```

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

A.
```
Cricket() { super(11, "Cricket",
"Condidtion OK"); num_umpires
=3;
num_substitutes=2;
}
```

B.
```
Cricket()
{ super.ground_condition =
"Condition OK";
```

```
    super.name="Cricket";
    super.num_players = 11;
    num_umpires =3;
    num_substitutes=2;
}
```

C.
```
Cricket() { this(3,2); super(11,
    "Cricket", "Condidtion OK");
}
Cricket(int nu, ns)
{ this.num_umpires =nu;
this.num_substitutes=ns;
}
```

D.
```
Cricket() { this.num_umpires =3;
    this.num_substitutes=2; super(11,
    "Cricket", "Condidtion OK");
}
```

Answer: A

Explanation:

Incorrect:

not C, not D: call to super must be the first statement in constructor.

Question No : 229

Which usage represents a valid way of compiling java source file with the name "Main"?

A. javac Main.java
B. java Main.class
C. java Main.java
D. javac Main
E. java Main

Answer: A

Explanation: The compiler is invoked by the javac command. When compiling a Java class, you must include the file name, which houses the main classes including the Java extension. So to run Main.java file we have to use command in option A.

TO execute Java program we can use Java command but can't use it for compiling.

https://docs.oracle.com/javase/tutorial/getStarted/application/index.html

Question No : 230

Given the code fragment:

```
public static void main(String[] args) {
    String[] arr = {"A", "B", "C", "D"};
    for (int i = 0; i < arr.length; i++) {
        System.out.print(arr[i] + " ");
        if (arr[i].equals("C")) {
            continue;
        }
        System.out.println("Work done");
        break;
    }
}
```

What is the result?

A. A B C Work done
B. A B C D Work done
C. A Work done
D. Compilation fails

Answer: C

Question No : 231

Which two are valid array declaration?

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

Answer: A,C

Question No : 232

Given the code fragment:

```
public class Test {
    public static void main(String[] args) {
        //line n1
        switch (x) {
            case 1:
                System.out.println("One");
                break;
            case 2:
                System.out.println("Two");
                break;
        }
    }
}
```

Which three code fragments can be independently inserted at line nl to enable the code to print one?

A. Byte x = 1;
B. short x = 1;
C. String x = "1";
D. Long x = 1;
E. Double x = 1;
F. Integer x = new Integer ("1");

Answer: A,B,F

Question No : 233

Given:

```
1. public class Whizlabs{
2.         public static void main(String[] args){
3.                 StringBuilder sb = new StringBuilder("1Z0");
4.                 sb.concat("-808");
5.                 System.out.println(sb);
6.         }
7. }
```

What is the output?

A. 1Z0
B. 1Z0-808
C. An exception will be thrown.
D. Compilation fails due to error at line 3.
E. Compilation tails due to error at line 4.

Answer: E

Explanation:

Option E is the correct answer.

Code fails to compile because there is no method called concert in StringBuilder class. The concert method is in String class. Hence option E is correct Here we should have used append method of StringBuilder class, in that case option B would be correct.

https://docs.oracle.com/javase/tutorial/java/data/buffers.html

Question No : 234

Given the code fragment:

```
public static void main(String[] args) {
    StringBuilder sb = new StringBuilder(5);
    String s = "";

    if (sb.equals(s)) {
        System.out.println("Match 1");
    } else if (sb.toString().equals(s.toString())) {
        System.out.println("Match 2");
    } else {
        System.out.println("No Match");
    }
}
```

What is the result?

A. Match 1
B. Match 2
C. No Match
D. A NullPointerException is thrown at runtime.

Answer: B

Explanation:

it will compare the string contents of the StringBuilder with string object.

Question No : 235

Given the following class:

```
public class CheckingAccount {
    public int amount;
    public CheckingAccount(int amount){
        this.amount = amount;
    }
    public int getAmount(){
        return amount;
    }
    public void changeAmount(int x){
        amount += x;
    }
}
```

And given the following main method, located in another class:

```
public static void main(String[] args) {
    CheckingAccount acct = new CheckingAccount((int)(Math.random()*1000));
    //line n1
    System.out.println(acct.getAmount());
}
```

Which three lines, when inserted independently at line n1, cause the program to print a o balance?

A. this.amount = 0;
B. amount = 0;
C. acct (0) ;
D. acct.amount = 0;
E. acct. getAmount () = 0;
F. acct.changeAmount(0);
G. acct.changeAmount(-acct.amount);
H. acct.changeAmount(-acct.getAmount());

Answer: D,G,H

Question No : 236

Given:

```
class Overloading {

int x(double d) {

System.out.println("one");

return 0;

}
String x(double d) {

System.out.println("two");

return null;

}

double x(double d) {

System.out.println("three");

return 0.0;

}

public static void main(String[] args) {

new Overloading().x(4.0);

}

}
```

What is the result?

A. One
B. Two
C. Three
D. Compilation fails.

Answer: D