# 1z0-808

Number: 1z0-808
Passing Score: 800
Time Limit: 120 min
File Version: 1

1z0-808


https://www.gratisexam.com/

**Exam A**

**QUESTION 1**
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.

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 2**
Which statement is true about the `switch` statement?

A. It must contain the default section.
B. The break statement, at the end of each case block, is optional.
C. Its case label literals can be changed at runtime.
D. Its expression must evaluate to a collection of values.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 3**
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.

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 4**
Given these 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
}
```

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 `useElectricity` method.

An instance of the `Customer` class should never be able to tamper with or decrease the value of the member variable `bill`.

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`?

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;
}
```

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 5**
Given the code fragment:

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

    if (sb.toString().equals(s.toString())) {
        System.out.println("Match 1");
    } else if (sb.equals(s)) {
        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.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 6**
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
}
```

And given the code fragment:

Book book1 = new EBook();
book1.readBook();

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

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 7**
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 C D D
D. A B D
E. A B D C

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 8**
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

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 9**
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.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 10**
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;

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 11**
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 opt = "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.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 12**
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

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 13**
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: x;
System.out.println(d);
```

What is the result?

A. 100
B. 101

C. 102
D. 103
E. Compilation fails

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 14**
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.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 15**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 16**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 17**
Which three are advantages of the Java exception mechanism? (Choose three.)

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 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 customized to the particular program being created

**Correct Answer:** ACE
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Reference: http://javajee.com/introduction-to-exceptions-in-java

**QUESTION 18**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 19**
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(100);
        Alpha ref2 = new Alpha(50);
        Alpha ref3 = new Alpha(125);
        ref1.doPrint();
        ref2.doPrint();
        ref3.doPrint();
    }
}
```

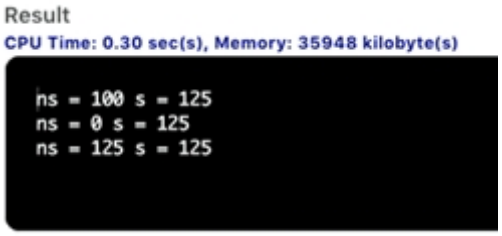
What is the result?

```
A) ns = 100 s = 125
   ns = 0 s = 125
   ns = 125 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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:



**QUESTION 20**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 21**
Given the code fragment:

```
LocalDate date1 = LocalDate.now();
LocalDate date2 = LocalDate.of(6, 20, 2014 );
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. An exception is thrown at runtime.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 22**
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? (Choose two.)

☐ 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

**Correct Answer:** AC
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 23**
Given:

```
public class Test {

    public static void main(String[] args) {
        if (args[0].equals("Hello") ? true : false) {
            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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 24**
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. 3
   5
C. 4
   7
D. 5
   4
E. 4
   5
F. 4
   21

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 25**
Given:

Acc.java:

```
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 via obj.
B. Only s is accessible via obj.
C. Both r and s are accessible via obj.
D. p, r, and s are accessible via obj.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 26**
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 use` is printed to the console.
E. The code fails to compile because a `throws` keyword is required.

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 27**
Given:

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

What is the result?

```
A) 5 + 2 = 34
   5 + 2 = 34

B) 5 + 2 + 3 + 4
   5 + 2 = 7

C) 7 = 7
   7 + 7

D) 5 + 2 = 34
   5 + 2 = 7
```

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

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 28**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 29**
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.

**Correct Answer:** D
**Section: (none)**

**Explanation**

**Explanation/Reference:**

**QUESTION 30**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 31**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 32**
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
```

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 33**
Given this 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 this 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].name);
    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
```

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 34**
You are asked to develop a program for a shopping application, and you are given this information:
- The application must contain the classes Toy, EduToy, and ConsToy. The Toy class is the superclass of the other two classes.
- The int calculatePrice (Toy t) method calculates the price of a toy.
- The void printToy (Toy t) method prints the details of a toy.

Which definition of the Toy class adds a valid layer of abstraction to the class hierarchy?

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 */ }
}
```

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 35**
Given this array:

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

Which two code fragments, independently, print each element in this array? (Choose two.)

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] +" ");
}
```

**Correct Answer:** BE
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 36**
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?

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.

**Correct Answer:** A
**Section: (none)**

**Explanation**

**Explanation/Reference:**

**QUESTION 37**
Given the code fragment:

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

And given the requirements:
1. Process all the elements of the array in the order of entry.
2. Process all the elements of the array in the reverse order of entry.
3. Process alternating elements of the array in the order of entry.

Which two statements are true? (Choose two.)

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.

**Correct Answer:** DE
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 38**
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.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 39**
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

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 40**
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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 41**
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.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 42**
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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 43**
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? (Choose two.)

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;

**Correct Answer:** CE
**Section: (none)**

**Explanation**

**Explanation/Reference:**

**QUESTION 44**
Which one of the following code examples uses valid Java syntax?

A.

```
public class Boat {

        public static void main (String [] args) {
               System.out.println  ("I float.");
               }
}
```

B.

```
public class Cake {
         public static void main (String [] ) {
               System.out.println  ("Chocolate");
               }
}
```

C.

```
public class Dog {
         public void main (String [] args) {
               System.out.println  ("Squirrel.");
               }
}
```

D.

```
public class Bank {
         public static void main (String () args) {
               System.out.println  ("Earn interest.");
               }
}
```

A. Option A

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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Reference: https://docs.oracle.com/javase/tutorial/getStarted/application/

**QUESTION 45**
Given the code fragment:

```
int n [] [] = {{1, 3}, {2, 4 }};
for (int i = n.length-1; i >= 0; i--) {
    for (int y : n[i]) {
        System.out.print (y);
    }
}
```

What is the result?

A. 1324
B. 2313
C. 3142
D. 4231

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 46**
Given:

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

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

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 47**
Given the code fragment:

```
public static void main(String[] args) {
    try {
        int num = 10;
        int div = 0;
        int ans = nim / div;
    } catch (ArithmeticExeption ae) {
        ans = 0                                    // line n1
    } catch (Exception e) {
        System.out.println("Invalid calculation");
    }
    System.out.println("Answer = " + ans);     // line n2
}
```

What is the result?

A. `Answer = 0`
B. `Invalid calculation`
C. Compilation fails only at `line n1`.
D. Compilation fails only at `line n2`.
E. Compilation fails at `line n1` and `line2`.

**Correct Answer:** E
**Section: (none)**

**Explanation**

**Explanation/Reference:**

**QUESTION 48**
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 = 'o';
        obj2.var = 'i';

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

What is the result?

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

D. a, a
   o, o

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 49**
Given the code fragment:

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

What is the result if the integer aVar is 9?

A. Compilation fails.
B. `10 Hello Universe!`
C. `10 Hello World!`
D. `9 Hello World!`

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 50**
Given the code fragment:

```
public static void main(String[] args) {
    int[][] arr = new int [2] [4];
    arr[0] = new int []{1, 3, 5, 7};
    arr[1] = new int []{1, 3};
    for (int[] a : arr) {
        for (int i : a) {
            System.out.print(i+ " ");
        }
        System.out.println();
    }
}
```

What is the result?

A. Compilation fails.

B.
```
1 3
1 3
```

C.
```
1 3
```
followed by an ArrayIndexOutOfBoundsException

D.
```
1 3
1 3 0 0
```

E.
```
1 3 5 7
1 3
```

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:


**QUESTION 51**
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 ();

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 52**
Given:

```
String stuff = "TV";
String res = null;

if (stuff.equals("TV")) {
    res = "Walter";
} else if (stuff.equals("Movie")) {
    res = "White";
} else {
    res = "No Result";
}
```

Which code fragment can replace the if block?

A.
```
stuff.equals ("TV") ? res= "Walter" : stuff.equals ("Movie") ?
res = "White" : res = "No Result";
```

B.
```
res = stuff.equals ("TV") ? "Walter" else stuff.equals
("Movie")? "White" : "No Result";
```

C.
```
res = stuff.equals ("TV") ? stuff.equals ("Movie")? "Walter" :
"White" : "No Result";
```

D.
```
res = stuff.equals ("TV")? "Walter" : stuff.equals ("Movie")?
"White" : "No Result";
```

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 53**
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 (p2);
```

B.
```
int f = ps.indexOf (Patient ("Mike") );
```

C.
```
int f = ps.indexOf (new Patient "Mike") );
```

D.
```
Patient p = new Patient("Mike");
int f = ps.indexOf(p)
```

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 54**
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 30.0
```

C.
```
integer sum is 30
double sum is 30.0
```

D.
```
integer sum is 30
float sum is 30.0
```

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 55**
Which three statements are true about the structure of a Java class? (Choose three.)

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

**Correct Answer:** ACE
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 56**
Given:

```
public class App {
    int count;
    public static void displayMsg () {
        count++;                                                    // line n1
    System.out.println ("Welcome "+"Visit Count: "+count); // line n2
    }
    public static void main (String [] args) {
        App.displayMsg ();                                          // line n3
        App.displayMsg ();                                          // line n4
    }
}
```

What is the result?

A. `Compilation fails at line n3 and line n4.`
B. `Compilation fails at line n1 and line n2.`
C. `Welcome Visit Count:1`
   `Welcome Visit Count: 1`
D. `Welcome Visit Count:1`
   `Welcome Visit Count: 2`

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 57**
Given the code fragment:

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

    Person(String name) {                          // 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;
    }

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

What is the result?

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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 58**
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. 5 : 5
B. 10 : 10
C. 5 : 10
D. Compilation fails.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

Version - JDK 1.8.0_66

**Your Code ...**

```
1  public class Test {
2
3          static int count = 0 ;
4          int i = 0;
5
6          public void changecount () {
7              while (i<5) {
8                  i++;
9                  count++;
10             }
11         }
12     public static void main (String [ ] args) {
13        Test check1 = new Test () ;
14        Test check2 = new Test () ;
15        check1.changecount () ;
16        check2.changecount () ;
17        System.out. print (check1.count + " : " + check2.count) ;
18     }
19 }
20
```

**External Libraries ...** Add External Library (from Maven Repo)

cs1.keyboard

**Input Arguments (args of Main Method)...**

**Interactive mode :** OFF

**Stdin Inputs...**

Execute | Save | My Projects | Recent | Collaborate | Others | Goto Another Language/DB

**Result...**

*compiled and executed in 1.357 second(s)*

```
10 : 10
```

**QUESTION 59**
Given the code fragment:

```
public static void main(String[] args) {
    ArrayList<Integer> points = new ArrayList<>();
    points.add(1);
    points.add(2);
    points.add(3);
    points.add(4);
    points.add(null);
    points.remove(1);
    points.remove(null);
    System.out.println(points);
}
```

What is the result?

A. A NullPointerException is thrown at runtime.
B. [1, 2, 4]
C. [1, 2, 4, null]
D. [1, 3, 4, null]
E. [1, 3, 4]
F. Compilation fails.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

**QUESTION 60**
Which two code fragments cause a compilation error? (Choose two.)

A.
```
float flt = 100.00F;
```
B.
```
float flt = (float) 1_11.00;
```
C.
```
Float flt = 100.00;
```
D.
```
double y1 = 203.22;
float flt = y1;
```
E.
```
int y2 = 100;
float flt = (float) y2 ;
```

**Correct Answer:** AD
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 61**
Given:

```
public class Fieldinit {
    char c;
    boolean b;
    float f;
    void printAll() {
        System.out.println ("c = " + c);
        System.out.println ("b = " + b);
        System.out.println ("f = " + f);
    }
    public static void main (String [] args) {
        FieldInit f = new FieldInit ();
        f.printAll ();
    }
}
```

What is the result?

A.
```
c=
b = false
f = 0.0
```

B.
```
c= null
b = true
f = 0.0
```

C.
```
c=0
b = false
f = 0.0f
```

D.
```
c= null
b = false
f = 0.0F
```

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 62**
Which three statements are true about exception handling? (Choose three.)

A. Only unchecked exceptions can be rethrown.
B. All subclasses of the `RuntimeException` class are not recoverable.
C. The parameter in a catch block is of `Throwable` type.
D. All subclasses of the `RuntimeException` class must be caught or declared to be thrown.
E. All subclasses of the `RuntimeException` class are unchecked exceptions.
F. All subclasses of the Error class are not recoverable.

**Correct Answer:** BCD
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 63**
Given the code fragment:

```
public static void main(String[] args) {
    int[] stack = {10, 20, 30};
    int size = 3;
    int idx = 0;
    /* line n1 */
    System.out.print("The Top element: " + stack[idx]);
}
```

Which code fragment, inserted at line n1, prints The Top element: 30?

A.
```
do {
    idx++;
} while (idx >=size);
```

B.
```
while (idx < size) {
    idx++;
}
```

C.
```
do {
    idx++;
} while (idx <size -1);
```

D.
```
do {
    idx++;
} while (idx<= size);
```

E.
```
while (idx <= size -1) {
    idx++
}
```

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 64**
Given the code fragment:

```
public static void main(String[] args) {
    String myStr = "Hello World ";
    myStr.trim();
    int i1 = myStr.indexOf(" ");
    System.out.println(i1);
}
```

What is the result?

A. An exception is thrown at runtime.
B. -1
C. 5
D. 10

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 65**
Given:

```
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.equals(str3));
       boolean b2 = (str1== str3);
       System.out.print (b1+",   "+b2);
    }
}
```



What is the result?

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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 66**
Given the code fragment:

```
public static void main(String[] args) {
    int data[] = {2010, 2013, 2014, 2015, 2014};
    int key = 2014;
    int count = 0;
    for (int e: data) {
        if (e != key) {
            continue;
            count++;
        }
    }
    System.out.print(count + " Found");
}
```

What is the result?

A. Compilation fails.
B. 0 Found
C. 1 Found
D. 3 Found

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 67**
Given the code fragment:

```
LocalDate Time dt= LocalDateTime.of (2014, 7, 31, 1, 1);
dt.plusDays (30);
dt. plusMonths (1);
System.out.print (dt format (DateTimeFormatter. ISO_DATE) );
```

What is the result?

A. An exception is thrown at runtime.
B. 07-31-2014
C. 2014-07-31
D. 2014-09-30

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 68**
Given:

```
public class Test {
    public static final int MIN = 1;
    public static void main(String[] args) {
        int x = args.length;
        if(checkLimit(x)){      // line n1
            System.out.println("Java SE");
        } else {
            System.out.println("Java EE");
        }
    }
    public static boolean checkLimit(int x) {
        return (x >= MIN) ? true : false;
    }
}
```

And given the commands:

```
javac Test.java
java Test 1
```

What is the result?

A. Java SE
B. Java EE
C. Compilation fails at line n1.
D. A NullPointerException is thrown at runtime.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 69**
Given this class:

```
public class CheckingAccount {
    public int amount;
    //line n1
}
```

And given this `main` method, located in another class:

```
public static void main(String[] args) {
    CheckingAccount acct = new CheckingAccount();
    //line n2
}
```

Which three pieces of code, when inserted independently, set the value of amount to 100?

A. At `line n1` insert:

```
public CheckingAccount(){
    amount = 100;
}
```

B. At `line n2` insert:

```
this.amount = 100;
```

C. At `line n2` insert:

```
amount = 100;
```

D. At `line n1` insert:

```
public CheckingAccount(){
    this.amount = 100;
}
```

E. At `line n2` insert:

```
acct.amount = 100;
```

F. At `line n1` insert:

```
public CheckingAccount(){
    acct.amount = 100;
}
```

**Correct Answer:** CDE
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 70**
Given the code fragments:

```
Interface Exportable {
    Void export();
}

class Tool implements Exportable {
    protected void export () {          //line n1
        System.out.println("Tool::export");
    }
}

class ReportTool extends Tool implements Exportable {

    public void export() {              //line n2
        System.out.println("RTool::export");
    }

    public static void main(String[] args) {
        Tool aTool = new ReportTool();
        Tool bTool = new Tool();
        callExport(aTool);
        callExport(bTool);
    }

    public static void callExport (Exportable ex) {
        ex.export();
    }
}
```

What is the result?

A. Compilation fails only at line n2.
B. RTool::export
   Tool::export
C. Tool::export
   Tool:export
D. Compilation fails only at line n1.
E. Compilation fails at both line n1 and line n2.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 71**
Given the code 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. An exception is thrown at runtime.
B. Compilation fails.
C. 13480.0
D. 13480.02

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 72**
Given:

```
class C2 {
    public void displayC2() {
        System.out.print("C2");
    }
}
interface I {
    public void displayI();
}
class C1 extends C2 implements I {
    public void displayI() {
        System.out.print("C1");
    }
}
```

And given the code fragment:

```
C2 obj1 = new C1();
I obj2 = new C1();

C2 s = obj2;
I t = obj1;

t.displayI();
s.displayC2()
```

What is the result?

A. C2C2
B. C1C2
C. C1C1
D. Compilation fails

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 73**
Given the code fragment:

```
package clothing.pants;
// line n1
public class Jeans {
    public void matchShirt(){
         // line n2
         if(color.equals("Green")) {
             System.out.print("Fit");
         }
    }
    public static void main(String[] args) {
        Jeans trouser = new Jeans();
        trouser.matchShirt();
    }
}
```

Which two sets of actions, independently, enable the code fragment to print `Fit`?

A. At line n1 insert: `import clothing.Shirt;`
   At line n2 insert: `String color = Shirt.getColor();`
B. At line n1 insert: `import clothing;`
   At line n2 insert: `String color = Shirt.getColor();`

C. At line n1 insert: `import static clothing.Shirt.getColor;`
   At line n2 insert: `String color = getColor();`

D. At line n1 no changes required.
   At line n2 insert: `String color = Shirt.getColor();`

E. At line n1 insert: `import Shirt;`
   At line n2 insert: `String color = Shirt.getColor();`

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 74**
Given the code fragments:

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

And:

```
4.public class Test {
5.   public static void main(String[] args) {
6.        Student s1 = new Student();
7.        Student s2 = new Student();
8.        Student s3 = new Student();
9.        s1 = s3;
10.       s3 = s2;
11.       s2 = null;
12.  }
13.}
```

Which statement is true?

A. After line 11, three objects are eligible for garbage collection.
B. After line 11, two objects are eligible for garbage collection.
C. After line 11, one object is eligible for garbage collection.
D. After line 11, none of the objects are eligible for garbage collection.

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 75**
Given the code fragment:

```
int wd = 0;
String days[] = ("sun", "mon", "wed", "sat");
for (String s:days) {
    switch (s) {
        case "sat":
        case "sun":
            wd -= 1:
            break;
        case "mon":
            wd++;
        case "wed":
            wd += 2;
    }
}
System.out.println(wd);
```

What is the result?

A. 3
B. 4
C. -1
D. Compilation fails.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 76**
Given the code fragment:

```
public static void main(String[] args) {
      LocalDate date = LocalDate.of(2012, 01, 32);
      date.plusDays(10);
      System.out.println(date);
}
```

What is the result?

A. `2012-02-10`
B. `2012-02-11`
C. Compilation fails
D. A `DateTimeException` is thrown at runtime.

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 77**
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 : 30 : 6

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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

**QUESTION 78**
Given:

```
interface Downloadable {
      public void download();
}

interface Readable extends Downloadable {         // line n1
      public void readBook();
}

abstract class Book implements Readable {         // line n2
      public void readBook() {
            System.out.println("Read Book");
      }
}

class EBook extends Book {                    // line n3
      public void readBook() {
            System.out.println("Read E-Book");
      }
}
```

And given the code fragment:

```
Book book1 = new EBook();
book1.readBook();
```

What is the result?

A. Compilation fails at line n2.
B. Read Book
C. Read E-Book
D. Compilation fails at line n1.
E. Compilation fails at line n3.

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 79**
Given this class:

```
public class Rectangle {
      private double length;
      private double height;
      private double area;

      public void setLength(double length) {
            this.length = length;
      }
      public void setHeight(double height) {
            this.height = height;
      }
      public void setArea() {
            area = length*height;
      }
}
```

Which two changes would encapsulate this class and ensure that the area field is always equal to `length * height` whenever the Rectangle class is used?

A. Call the setArea method at the end of the setHeight method.
B. Call the setArea method at the beginning of the setHeight method.
C. Call the setArea method at the end of the setLength method.
D. Call the setArea method at the beginning of the setLength method.
E. Change the setArea method to private.
F. Change the area field to public.

**Correct Answer:** AC
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 80**
Given the code fragment:

```
13. List colors = new ArrayList();
14. colors.add("green");
15. colors.add("blue");
16. colors.add("red");
17. colors.add("yellow");
18. colors.remove(2);
19. colors.add(3, "cyan");
20. System.out.print(colors);
```

What is the result?

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

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 81**
Given the code fragment:

```
abstract class Toy {
      int price;
      // line n1
}
```

Which three code fragments are valid at line n1? (Choose three.)

A.
```
public static void insertToy() {
     /* code goes here */
}
```

B.
```
final Toy getToy() {
    return new Toy();
}
```

C.
```
public void printToy();
```

D.
```
public int calculatePrice() {
     return price;
}
```

E.
```
public abstract int computeDiscount();
```

**Correct Answer:** ADE
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 82**
Given:

```
public class Test {
      int x, y;

      public Test(int x, int y) {
            initialize(x, y);
      }

      public void initialize(int x, int y) {
            this.x = x * x;
            this.y = y * y;
      }

      public static void main(String[] args) {
            int x = 3, y = 5;
            Test obj = new Test(x, y);
            System.out.println(x + " " + y);
      }
}
```

What is the result?

A. Compilation fails.
B. 3 5
C. 0 0
D. 9 25

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 83**

Given:

```
class Test
      int a1;

      public static void doProduct(int a) {
            a = a * a;
      )

      public static void doString(String s) {
            s.concat (" " + s);
      }

      public static void main(String[] args) {
            Test item = new Test();
            item.a1 = 11;
            String sb = "Hello";
            Integer i = 10;
            doProduct(i);
            doString(sb);
            doProduct(item.a1);
            System.out.println(i + " " + sb + " " + item.a1);
      }
}
```

What is the result?

A. `10 Hello Hello 11`
B. `10 Hello Hello 121`
C. `100 Hello 121`
D. `100 Hello Hello 121`

E. `10 Hello 11`

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 84**
Given the code fragment:

```
public static void main (String[] args) {
      String[] arr = (“Hi”, “How”, “Are”, “You”);
      List<String> arrList = new ArrayList<>(Arrays.asList(arr);
      if (arrList.removeIf((String s) -> (return s.length() <= 2;))) {
            System.out.println(s + “removed”)’
      }
}
```

What is the result?

A. Compilation fails.
B. Hi removed
C. An UnsupportedOperationException is thrown at runtime.
D. The program compiles, but it prints nothing.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 85**
Given:

```
public class Test {
      public static void main(String[] args) {
            int x = 1;
            int y = 0;
            if(x++ > ++y) {
                  System.out.print("Hello ");
            } else {
                  System.out.print("Welcome ");
            }
            System.out.print("Log " + x + ":" + y);
      }
}
```

What is the result?

A. Hello Log 1:0
B. Hello Log 2:1
C. Welcome Log 2:1
D. Welcome Log 1:0

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 86**
Given the code fragment:

```
public static void main(String[] args) {
    int[][] arr = new int[2][4];

    arr[0] = new int[]{1, 3, 5, 7};
    arr[1] = new int[]{1, 3};

    for (int[] a : arr) {
        for (int i=0; i < arr.length; i++) {
            System.out.print(a[i] + " ");
        }
        System.out.println();
    }
}
```

What is the result?

A. 1 3 5 7
   1 3
B. 1 3
   1 3
C. 1 3
   1 3 0 0
D. 1 3
   followed by an `ArrayIndexOutOfBoundsException`
E. Compilation fails.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

```
1  class Main {
2    public static void main(String[] args) {
3      int[][] arr = new int[2][4];
4
5      arr[0] = new int[] {1, 2, 3, 5, 7};
6      arr[1] = new int[] {1, 3};
7
8      for (int[] a : arr) {
9        for (int i=0; i <arr.length; i++){
10         System.out.print (a[i] + " ");
11       }
12       System.out.println();
13     }
14     }
```

```
Java(TM) SE Runtime Environment (build 1.8.0_31-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.31-b07, mixed mode)
> javac -classpath .:/run_dir/junit-4.12.jar:/run_dir/hamcrest-c
ore-1.3.jar:/run_dir/json-simple-1.1.1.jar -d . Main.java
> java -classpath .:/run_dir/junit-4.12.jar:/run_dir/hamcrest-co
re-1.3.jar:/run_dir/json-simple-1.1.1.jar Main
1 2
1 3
```

**QUESTION 87**
Given:

MainTest.java:

```
public class MainTest {

    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. `String main 1`

B. An exception is thrown at runtime
C. `String main 1 2 3`
D. `String main 123`

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 88**
Which two statements are true about Java byte code? (Choose two.)

A. It can be serialized across network.
B. It can run on any platform that has a Java compiler.
C. It can run on any platform.
D. It has ".java" extension.
E. It can run on any platform that has the Java Runtime Environment.

**Correct Answer:** AE
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 89**
Given the code fragment:

```
int n[][] = {{1, 3}, {2, 4}};
for (int i = n.length - 1; i >= 0; i--) {
    for (int j = n[i].length - 1; j >= 0; j--) {
        System.out.print(n[i][j]);
    }
}
```

What is the result?

A. 3142
B. 2413
C. 1324
D. 4231

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

Main.java saved

```
1
2   class C {
3     public C() {
4       System.out.print("C ");
5     }
6   }
7
8   class B extends C{
9     public B() {
10      System.out.print("B ");
11    }
12  }
13 public class A extends B{
14
15    public A(){
16      System.out.print("A ");
17    }
18    public static void main(String[] args) {
19      A a = new A();
20    }
21 }
```

```
java version "1.8.0_31"
Java(TM) SE Runtime Environment (build 1.8.0_31-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.31-b07, mixed mode)
> javac -classpath .:/run_dir/junit-4.12.jar:/run_dir/hamcrest-c
ore-1.3.jar:/run_dir/json-simple-1.1.1.jar -d . Main.java
Main.java:13: error: class A is public, should be declared in a
file named A.java
public class A extends B{
       ^
1 error
compiler exit status 1
```

**QUESTION 90**
Given the code fragments:

```
interface Exportable {
     void export();
}

class Tool implements Exportable {
    public void export() {                    // line n1
        System.out.println("Tool::export");
    }
}

class ReportTool extends Tool {

    void export() {                           // line n2
        System.out.println("RTool::export");
    }

    public static void main(String[] args) {
        Tool aTool = new ReportTool();
        Tool bTool = new Tool();
        callExport(aTool);
        callExport(bTool);
    }

    public static void callExport(Exportable ex)  {
        ex.export();
    }
}
```

What is the result?

A. Compilation fails only at line n1.
B. Compilation fails only at line n2.
C. `Tool::export`

`Tool::export`

D. Compilation fails at both line n1 and line2.

E. `RTool::export`
`Tool::export`

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 91**
Given the code fragment:

```
public static void main(String[] args) {
    LocalDate date = LocalDate.of(2012, 1, 30);
    date.plusDays(10);
    System.out.println(date);
}
```

What is the result?

A. `2012-02-10 00:00`
B. `2012-01-30`
C. `2012-02-10`
D. A `DateTimeException` is thrown at runtime.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:


**QUESTION 92**
Given:

```
public class Test {
    public static void main(String[] args) {
        int x = 1;
        int y = 1;
        if(x++ < ++y){
            System.out.print("Hello ");
        } else {
            System.out.print("Welcome ");
        }
        System.out.print("Log " + x + ":" + y);
    }
}
```

What is the result?

A. `Hello Log 2:2`
B. `Welcome Log 1:2`
C. `Welcome Log 2:1`

D. `Hello Log 1:2`

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

```
1  public class Main {
2    public static void main(String[] args) {
3      int x = 1;
4      int y = 1;
5      if (x++ < ++y) {
6        System.out.print("Hello ");
7      } else {
8        System.out.print("Welcome ");
9      }
10     System.out.print("Log " +x+ ":" + y);
11   }
12 }
```

```
Java(TM) SE Runtime Environment (build 1.8.0_31-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.31-b07, mixed mode)
> javac -classpath .:/run_dir/junit-4.12.jar:/run_dir/hamcrest-c
ore-1.3.jar:/run_dir/json-simple-1.1.1.jar -d . Main.java
> java -classpath .:/run_dir/junit-4.12.jar:/run_dir/hamcrest-co
re-1.3.jar:/run_dir/json-simple-1.1.1.jar Main
Hello Log 2:2
```

**QUESTION 93**
Given:

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

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

public class A extends B{

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

What is the result?

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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 94**
Given this 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 3 > 10 ? new MyException() : new IOException();
       }
        catch(IOException ie) {
            System.out.println("I");
        }
       catch (Exception re) {
           System.out.print("B");
       }
    }
}
```

What is the result?

A. A
B. AB
C. A compile time error occurs at `line n1`.
D. B
E. I

**Correct Answer:** E

**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

**QUESTION 95**
Given the code fragment:

```
1. abstract class Planet {
2.     protected void revolve() {
3.     }
4.     abstract void rotate();
5. }
6.
7. class Earth extends Planet {
8.     private void revolve() {
9.     }
10.    private void rotate() {
11.    }
12. }
```

Which two modifications enable the code to compile?

A. Make the method at line 8 `protected`.
B. Make the method at line 8 `public`.
C. Make the method at line 10 `protected`.
D. Make the method at line 4 `public`.
E. Make the method at line 2 `public`.

**Correct Answer:** AC
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 96**
Given:

```
interface I {
    public void displayI();
}
abstract class C2 implements I {
    public void displayC2() {
        System.out.print("C2");
    }
}
class C1 extends C2  {
    public void displayI() {
        System.out.print("C1");
    }
}
```

And the code fragment:

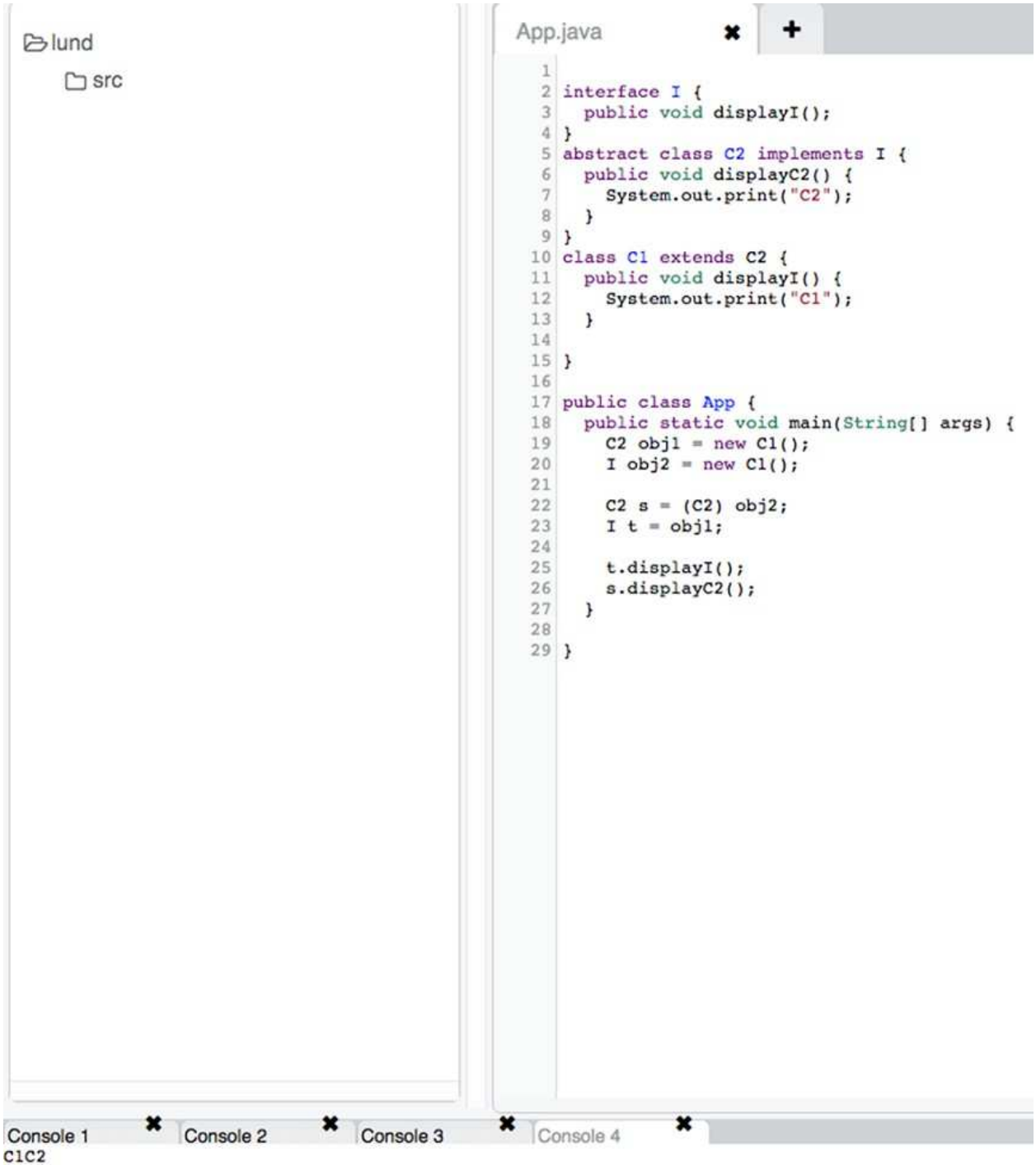
```
C2 obj1 = new C1();
I obj2 = new C1();

C2 s = (C2) obj2;
I t = obj1;

t.displayI();
s.displayC2();
```

What is the result?

A. `C1C2`
B. `C1C1`
C. Compilation fails.
D. `C2C2`

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:


lund
src
App.java
1
2 interface I {
3 public void displayI();
4 }
5 abstract class C2 implements I {
6 public void displayC2() {
7 System.out.print("C2");
8 }
9 }
10 class C1 extends C2 {
11 public void displayI() {
12 System.out.print("C1");
13 }
14
15 }
16
17 public class App {
18 public static void main(String[] args) {
19 C2 obj1 = new C1();
20 I obj2 = new C1();
21
22 C2 s = (C2) obj2;
23 I t = obj1;
24
25 t.displayI();
26 s.displayC2();
27 }
28
29 }
Console 1
Console 2
Console 3
Console 4
C1C2

**QUESTION 97**
Given the code fragment:

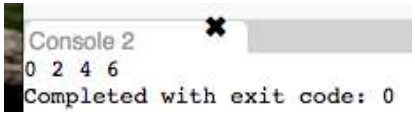
```
public static void main(String[] args) {
    int ii = 0;
    int jj = 7;
    for (ii = 0; ii < jj; ii = ii + 2) {
        System.out.print(ii + " ");
     }
}
```

What is the result?

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

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:




**QUESTION 98**
Given the code fragment:

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

And given the requirements:
1. Process all the elements of the array in the reverse order of entry.
2. Process all the elements of the array in the order of entry.
3. Process alternating elements of the array in the order of entry.

Which two statements are true? (Choose two.)

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 2 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.

**Correct Answer:** BC
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 99**
Given:

```
public class Test {

    public static void main(String[] args) {

        String[][] chs = new String[5][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();
        }
    }
}
```

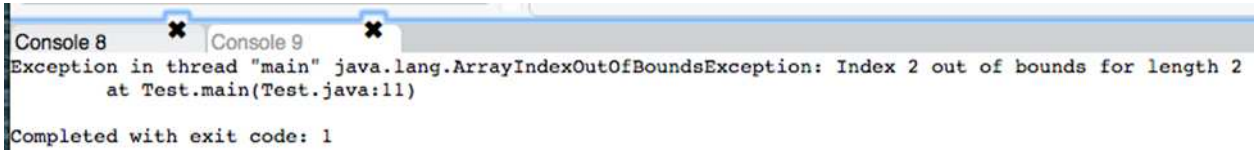
What is the result?

A. 97 98
99 100 null null null

B. 97 98
99 100 101 102 103
C. Compilation fails.
D. A NullPointerException is thrown at runtime.
E. An ArrayIndexOutOfBoundsException is thrown at runtime.

**Correct Answer:** E
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:




**QUESTION 100**
Given the code fragment:

```
LocalDateTime dt = LocalDateTime.of(2014, 7, 31, 1, 1);
dt.plusDays(30);
dt.plusMonths(1);
System.out.println(dt.format(DateTimeFormatter.ISO_DATE_TIME));
```

What is the result?

A. An exception is thrown at runtime.
B. `2014-07-31T01:01:00`
C. `2014-07-31`
D. `2014-09-30T00:00:00`

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 101**
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 main(String[] args) {
        doSum(10, 20);
        doSum(10.0, 20.0);
    }
}
```

What is the result?

A.
```
float sum is 30.0
double sum is 30.0
```

B.
```
double sum is 30.0
float sum is 30.0
```

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

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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 102**
Given:

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

class B extends A {
    public void test() {
        System.out.println("B ");
    }
}

public class C extends A {
    public void test() {
        System.out.println("C ");
    }

    public static void main(String[] args) {
        A b1 = new A();
        A b2 = new C();
        A b3 = (B) b2;              //line n1
        b1 = (A) b2;                //line n2
        b1.test();
        b3.test();
    }
}
```

What is the result?

A. A
   B
B. A
   C
C. C
   C

D. A `ClassCastException` is thrown only at `line n1`.
E. A `ClassCastException` is thrown only at `line n2`.

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 103**
Given:

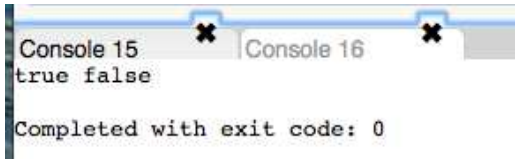
```
public class Test {
    public static void main(String[] args) {
        Test ts = new Test();
        System.out.print(isAvailable + " ");
        isAvailable= ts.doStuff();
        System.out.println(isAvailable);
    }
    public static boolean doStuff() {
        return !isAvailable;
    }
 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

What is the result?

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

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:




**QUESTION 104**
Given the code fragments:

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

And:

```
4.public class Test {
5.   public static void main(String[] args) {
6.        Student s1 = new Student();
7.        Student s2 = new Student();
8.        Student s3 = new Student();
9.        s1 = s3;
10.       s3 = s2;
11.       s1 = s2;
12.  }
13.}
```

Which statement is true?

A. After line 11, three objects are eligible for garbage collection.
B. After line 11, two objects are eligible for garbage collection.
C. After line 11, one object is eligible for garbage collection.
D. After line 11, none of the objects are eligible for garbage collection.

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

Result

CPU Time: sec(s), Memory: kilobyte(s)

```
/Test.java:3: error: cannot find symbol
        Student s1 = new Student() ;
        ^
  symbol:   class Student
  location: class Test
/Test.java:3: error: cannot find symbol
        Student s1 = new Student() ;
                     ^
  symbol:   class Student
  location: class Test
/Test.java:4: error: cannot find symbol
        Student s2 = new Student() ;
        ^
  symbol:   class Student
  location: class Test
/Test.java:4: error: cannot find symbol
        Student s2 = new Student() ;
                     ^
  symbol:   class Student
  location: class Test
/Test.java:5: error: cannot find symbol
        Student s3 = new Student() ;
        ^
  symbol:   class Student
  location: class Test
/Test.java:5: error: cannot find symbol
        Student s3 = new Student() ;
                     ^
  symbol:   class Student
  location: class Test
6 errors
```

**QUESTION 105**
Given the code snippet from a compiled Java source file:

```
public class MyFile
{
      public static void main (String[] args)
      {
            String arg1 = args[0];
            String arg2 = args[1];
            String arg3 = args[2];
            System.out.println("Arg is " + arg3);
      }
}
```

and this output:

```
Arg is 2
```

Which command should you run to obtain this output?

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

**Correct Answer:** A
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 106**
Given the code fragment:

```
10. public static void main(String[] args {
11.     List<String> lst = Arrays.asList("A", "B", "C", "D");
12.     Iterator<String> itr = lst.iterator();
13.     while(itr.hasNext()) {
14.         String e = itr.next();
15.         if (e == "C") {
16.             break;
17.         }
18.         else {
19              continue;
20.             System.out.print(e);
21.         }
22.     }
23. }
```

Which action enables it to print AB?

A. Comment lines 18 to 21.
B. Comment line 20.
C. Comment line 19.
D. Comment line 16.

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 107**
Given the definitions of the `Bird` class and the `Peacock` class:

```
public class Bird {
    public void fly() {
        System.out.print ("Fly.");
    }
}

public class Peacock extends Bird {
    public void dance() {
        System.out.print("Dance.");
    }
}
```

and the code fragment:

```
/*insert code snippet here */
p.fly();
p.dance();
```

Which code snippet can be inserted to print Fly.Dance. ?

A. `Bird p = new Peacock();`

B.
```
Bird b = new Bird();
Peacock p = (Peacock) b;
```

C.
```
Peacock b = new Peacock ();
Bird p = (Bird) b;
```

D.
```
Bird b = new Peacock ();
Peacock p = (Peacock) b;
```

**Correct Answer:** B
**Section: (none)**
**Explanation**

**Explanation/Reference:**

**QUESTION 108**
Given the code fragment:

```
int x = 10;
int y = ++x;
int z = 0;
if (y >= 10 | y <= ++x) {
    z = x;
} else {
    z = x++;
}
System.out.println(z);
```

What is the result?

A. 11
B. 10
C. 12
D. A compile time error occurs.

**Correct Answer:** C
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

Result

CPU Time: 0.14 sec(s), Memory: 32028 kilobyte(s)

```
12
```

**QUESTION 109**
Given the code fragment:

```
int a = 3;
int b = 2;
int c = 1;
int r1 = a * b / c + 1;
int r2 = a / b * c + 1;
int r3 = a * (b / (c + 1));
System.out.println(r1 + " : " + r2 + " : " + r3);
```

What is the result?

A. `2 : 7 : 3`
B. `7 : 7 : 9`
C. `2 : 7 : 0`
D. `7 : 2 : 3`

**Correct Answer:** D
**Section: (none)**
**Explanation**

**Explanation/Reference:**
Explanation:

Result

CPU Time: 0.32 sec(s), Memory: 35824 kilobyte(s)

```
7 : 2 : 3
```