Exam Code: 310-055

Exam Name: Sun Certified Programmer for the Java 2

Platform.SE 5.0

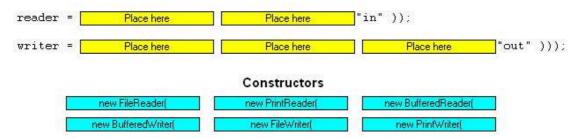
Vendor: Sun

Version: DEMO

Part: A

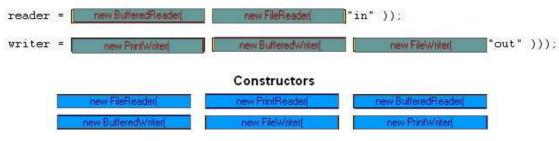
1: Click the Task button.

Chain these constructors to create objects to read from a file named "in" and to write to a file named "out."



Correct Answers:

Chain these constructors to create objects to read from a file named "in" and to write to a file named "out."



2: Given:

- 11. class Converter {
- 12. public static void main(String[] args) {
- 13. Integer i = args[0];
- 14. int j = 12;
- 15. System.out.println("It is " + (j==i) + " that j==i.");
- 16.
- 17. }

What is the result when the programmer attempts to compile the code and run it with the command java Converter 12?

A.It is true that j==i.

B.It is false that j==i.

C.An exception is thrown at runtime.

D.Compilation fails because of an error in line 13.

Correct Answers: D

- 3: Given:
- 22. StringBuilder sb1 = new StringBuilder("123");
- 23. String s1 = "123";
- 24. // insert code here
- 25. System.out.println(sb1 + " " + s1);

Which code fragment, inserted at line 24, outputs "123abc 123abc"?

```
A.sb1.append("abc"); s1.append("abc");
B.sb1.append("abc"); s1.concat("abc");
C.sb1.concat("abc"); s1.append("abc");
D.sb1.concat("abc"); s1.concat("abc");
E.sb1.append("abc"); s1 = s1.concat("abc");
F.sb1.concat("abc"); s1 = s1.concat("abc");
G.sb1.append("abc"); s1 = s1 + s1.concat("abc");
H.sb1.concat("abc"); s1 = s1 + s1.concat("abc");
Correct Answers: E
4: Given:
1. d is a valid, non-null Date object
2. df is a valid, non-null DateFormat object set to the current locale
What outputs the current locale's country name and the appropriate version of d's date?
A.Locale loc = Locale.getLocale();
System.out.println(loc.getDisplayCountry()
+ " " + df.format(d));
B.Locale loc = Locale.getDefault();
System.out.println(loc.getDisplayCountry()
+ " " + df.format(d));
C.Locale loc = Locale.getLocale();
System.out.println(loc.getDisplayCountry()
+ " " + df.setDateFormat(d));
D.Locale loc = Locale.getDefault();
System.out.println(loc.getDisplayCountry()
+ " " + df.setDateFormat(d));
Correct Answers: B
5: Click the Task button.
Place the code fragments into position to produce the output:
          true true false
 Code
 Scanner scanner = new Scanner( "One, 5, true, 3, true, 6, 7, false");
 scanner.useDelimiter(",");
                    Place here
                                        ) {
 while (
    if (
                   Place here
                                       ) {
      System.out.print(
                                       Place here
                      Place here
    } else
                      Code Fragments
      scanner.hasNextBoolean()
                                       scanner.nextBoolean()
                                                                        Done
           scanner.next()
                                         scanner.hasNext()
```

Correct Answers:

Place the code fragments into position to produce the output:

```
true true false
```

Code

Code Fragments



- 6: Given:
- 11. String test = "This is a test";
- 12. String[] tokens = test.split("\s");
- 13. System.out.println(tokens.length);

What is the result?

A.0

B.1

C.4

D.Compilation fails.

E.An exception is thrown at runtime.

Correct Answers: D

- 7: Given:
- 1. package geometry;
- 2. public class Hypotenuse {
- 3. public InnerTriangle it = new InnerTriangle();
- 4. class InnerTriangle {
- 5. public int base;
- 6. public int height;
- 7. }
- 8. }

Which statement is true about the class of an object that can reference the variable base?

A.It can be any class.

- B.No class has access to base.
- C.The class must belong to the geometry package.
- D.The class must be a subclass of the class Hypotenuse.

Correct Answers: C

```
8: Given:
1. class Super {
     private int a;
     protected Super(int a) { this.a = a; }
4. } ...
11. class Sub extends Super {
      public Sub(int a) { super(a); }
      public Sub() { this.a = 5; }
13.
14. }
Which two, independently, will allow Sub to compile? (Choose two.)
A.Change line 2 to:
public int a;
B.Change line 2 to:
protected int a;
C.Change line 13 to:
public Sub() { this(5); }
D.Change line 13 to:
public Sub() { super(5); }
E.Change line 13 to:
public Sub() { super(a); }
Correct Answers: C D
9: Click the Task button.
Given:
    class A {
      String name = "A";
      String getName() {
         return name;
      Śtring greeting(){
return "class A";
    class B extends A {
   String name = "B";
      String greeting() { return "class B";
    public class Client {
      public static void main( String[] args ) {
         A = new A();
         A b = new B()
         System.out.println(a.greeting() + " has name " + a.getName());
         System.out.println(b.greeting() + " has name " + b.getName());
    Place the names "A" and "B" in the following output.
                                                                          Names
                                          Place here
         class Place here has name
         class Place here
                             has name
                                           Place here
                                                                                  Done
```

Correct Answers:

```
Given:
    class A {
      String name = "A",
      String getName() {
         return name;
      Śtring greeting(){
return "class A"
    class B extends A {
  String name = "B";
      String greeting() { return "class B";
   public class Client {
   public static void main( String[] args ) {
         A = new A();
         A b = new B()
         System.out.println(a.greeting() + " has name " + a.getName());
         System.out.println(b.greeting() + " has name " + b.getName());
    }
    Place the names "A" and "B" in the following output.
                                                                         Names
         class
                             has name
         class
                              has name
                                                                                Done
10: Given:
1. public class Base {
     public static final String FOO = "foo";
```

- 3. public static void main(String[] args) {
- 4. Base b = new Base();
- 5. Sub s = new Sub();
- 6. System.out.print(Base.FOO);
- 7. System.out.print(Sub.FOO);
- 8. System.out.print(b.FOO);
- 9. System.out.print(s.FOO);
- 10. System.out.print(((Base)s).FOO);
- 11. } }
- 12. class Sub extends Base {public static final String FOO="bar";}

What is the result?

A.foofoofoofoo

B.foobarfoobarbar

C.foobarfoofoo

D.foobarfoo

E.barbarbarbar

F.foofoofoobarbar

G.foofoobarfoo

Correct Answers: D