JAVA means DURGA SOFT

SCJP / OCJP

Question Bank

Chapter 2: Declarations and Access Control



DURGA M.Tech

(Sun certified & Realtime Expert)

Ex. IBM Employee

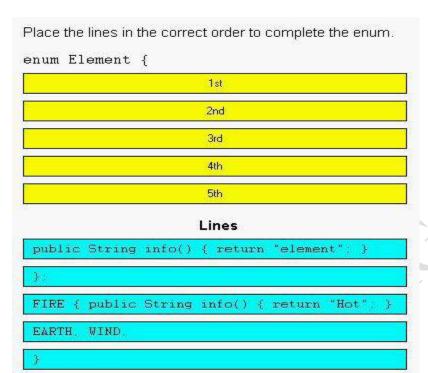
Trained Lakhs of Students for last 14 years across INDIA

India's No.1 Software Training Institute

8096969696, www.durgasoft.com Ph: 9246212143

2. Declarations and Access Control

Q: 1 Click the Task button.



Solution:



Q: 2 Given:

10. package com.sun.scjp;

11. public class Geodetics {

12. public static final double DIAMETER = 12756.32; // kilometers

13. }

Which two correctly access the DIAMETER member of the Geodetics class? (Choose two.)

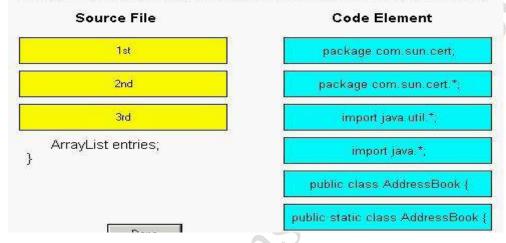
A. import com.sun.scjp.Geodetics; public class TerraCarta { public double halfway() { return Geodetics.DIAMETER/2.0; } B. import static com.sun.scjp.Geodetics; public class TerraCarta{

```
public double halfway() { return DIAMETER/2.0; } }
C. import static com.sun.scjp.Geodetics.*;
public class TerraCarta {
public double halfway() { return DIAMETER/2.0; } }
D. package com.sun.scjp;
public class TerraCarta {
public double halfway() { return DIAMETER/2.0; } }
```

Answer: A, C

Q: 3 Click the Task button.

Place the code elements in order so that the resulting Java source file will compile correctly, resulting in a class called com.sun.cert.AddressBook.



Solution:



Q: 4 Which two classes correctly implement both the java.lang.Runnable and the java.lang.Clonable interfaces? (Choose

```
A. public class Session
implements Runnable, Clonable {
public void run();
public Object clone();
}
B. public class Session
extends Runnable, Clonable {
public void run() { /* do something */ }
public Object clone() { /* make a copy */ }
C. public class Session
implements Runnable, Clonable {
```

```
public void run() { /* do something */ }
public Object clone() { /* make a copy */ }
D. public abstract class Session
implements Runnable, Clonable {
public void run() { /* do something */ }
public Object clone() { /*make a copy */ }
E. public class Session
implements Runnable, implements Clonable {
public void run() { /* do something */ }
public Object clone() { /* make a copy */ }
```

Q: 5 Given classes defined in two different files:

```
    package util;
    public class BitUtils {
    private static void process(byte[] b) {}
    }
```

1. package app;

2. public class SomeApp {

3. public static void main(String[] args) {

4. byte[] bytes = new byte[256];

5. // insert code here

6. }

7. }



What is required at line 5 in class SomeApp to use the process method of BitUtils?

A. process(bytes);

B. BitUtils.process(bytes);

C. app.BitUtils.process(bytes);

D. util.BitUtils.process(bytes);

E. import util.BitUtils.*; process(bytes);

F. SomeApp cannot use the process method in BitUtils.

```
Q: 6 Given:
11. class Cup { }
12. class PoisonCup extends Cup { }
21. public void takeCup(Cup c) {
22. if (c instanceof PoisonCup) {
23. System.out.println("Inconceivable!");
24. } else if (c instanceof Cup) {
25. System.out.println("Dizzying intellect!");
26. } else {
27. System.exit(0);
28. }
29. }
And the execution of the statements:
Cup cup = new PoisonCup();
takeCup(cup);
What is the output?
A. Inconceivable!
B. Dizzying intellect!
C. The code runs with no output.
D. An exception is thrown at runtime.
E. Compilation fails because of an error in line 22,
Q: 7 Click the Exhibit button.
public class A
```

```
Q: 7 Click the Exhibit button.
public class A
{
    private int counter=0;
    public static int getInstanceCount()
    {
        return counter;
    }
    public A()
    {
        counter++;
    }
}
Given this code from Class B:
25. A a1 = new A();
26. A a2 = new A();
27. A a3 = new A();
28. System.out.println(A.getInstanceCount());
What is the result?
A. Compilation of class A fails.
B. Line 28 prints the value 3 to System.out.
C. Line 28 prints the value 1 to System.out.
```

D. A runtime error occurs when line 25 executes.

E. Compilation fails because of an error on line 28.

```
Q:8 Given:
11. String[] elements = { "for", "tea", "too" };
12. String first = (elements.length > 0)? elements[0]: null;
What is the result?
A. Compilation fails.
B. An exception is thrown at runtime.
C. The variable first is set to null.
D. The variable first is set to elements[0].
Q:09 Given:
11. interface DeclareStuff {
12. public static final int EASY = 3;
13. void doStuff(int t); }
14. public class TestDeclare implements DeclareStuff {
15. public static void main(String [] args) {
16. int x = 5;
17. new TestDeclare().doStuff(++x);
18. }
19. void doStuff(int s) {
20. s += EASY + ++s;
21. System.out.println("s" + s);
22. }
23. }
What is the result?
A. s 14
B. s 16
C. s 10
D. Compilation fails.
E. An exception is thrown at runtime.
Q: 10 Given:
1. public class TestString1 {
2. public static void main(String[] args) {
3. String str = "420";
4. str += 42;
5. System.out.print(str);
6. }
7.}
What is the output?
```

A. 42

```
C. 462
D. 42042
E. Compilation fails.
F. An exception is thrown at runtime.
O: 11 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.
```

LEARN FROM EXPERT & DIAMOND FACULTIES OF AMEERPET... JAVAMEANS DURGASOFT INDIA'S NO. 1 SOFTWARE TRAINING INSTITUTE #202 2nd Floor www.durgasoft.com 040-64512786 +91 9246212143 +91 8096969696

```
Q: 12 Given:
10. int x = 0;
11. int y = 10;
12. do {
13. y--;
14. ++x;
15. \} while (x < 5);
16. System.out.print(x + ", " + y);
What is the result?
A. 5,6
B. 5,5
C. 6,5
D. 6,6
Q: 13 Given:
1. public interface A {
2. String DEFAULT_GREETING = "Hello World";
```

C. An exception is thrown at runtime.

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

B. 420

```
3. public void method1();
4. }
A programmer wants to create an interface called B that has A as its parent. Which interface
declaration is correct?
A. public interface B extends A {}
B. public interface B implements A {}
C. public interface B instanceOf A {}
D. public interface B inheritsFrom A {}
Q: 14 Given:
11. public enum Title {
12. MR("Mr."), MRS("Mrs."), MS("Ms.");
13. private final String title;
14. private Title(String t) { title = t; }
15. public String format(String last, String first) {
16. return title + " " + first + " " + last;
17. }
18. }
19. public static void main(String[] args) {
20. System.out.println(Title.MR.format("Doe", "John"));
21. }
What is the result?
A. Mr. John Doe
B. An exception is thrown at runtime.
C. Compilation fails because of an error in line 12.
D. Compilation fails because of an error in line 15.
E. Compilation fails because of an error in line 20.
Q: 15 Given:
1. package test;
2.
3. class Target {
4. public String name = "hello";
What can directly access and change the value of the variable name?
A. any class
                                 B. only the Target class
C. any class in the test package
                                  D. any class that extends Target
Q: 16 Given:
11. public class Ball{
12. public enum Color { RED, GREEN, BLUE };
13. public void foo(){
14. // insert code here
15. { System.out.println(c); }
16. }
17. }
Which code inserted at line 14 causes the foo method to print RED, GREEN, and BLUE?
```

```
A. for( Color c : Color.values() )
```

- B. for (Color c = RED; $c \le BLUE$; c++)
- C. for(Color c; c.hasNext(); c.next())
- D. for(Color c = Color[0]; $c \le Color[2]$; c++)
- E. for(Color c = Color.RED; $c \le Color.BLUE$; c++)





17 Click the Task button.

Insert six modifiers into the code such that it meets all of these requirements: 1. It must be possible to create instances of Alpha and Beta from outside the packages in which they are defined. 2. When an object of type Alpha (or any potential subclass of Alpha) has been created, the instance variable alpha may never be changed. 3. The value of the instance variable alpha must always be "A" for objects of type Alpha. Code package alpha; Place here | class Alpha { Modifiers Place here String alpha; private Place here Alpha() { this("A"); } Place here Alpha (String a) { alpha = a; } package beta; Place here class Beta extends alpha. Alpha { Place here Beta (String a) { super(a); } Done } **Solution: Q:** 18 Given: 1. public class Target { 2. private int i = 0; 3. public int addOne(){ 4. return ++i; **5.** } 6. }

Which change can you make to Target without affecting Client?

A. Line 4 of class Target can be changed to return i++;

2. public static void main(String[] args){3. System.out.println(new Target().addOne());

And:

4. } 5. }

1. public class Client {

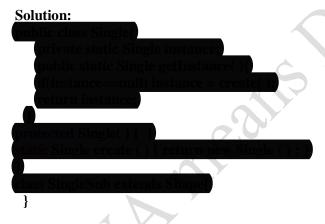
B. Line 2 of class Target can be changed to private int i = 1;

C. Line 3 of class Target can be changed to private int addOne() {
D. Line 2 of class Target can be changed to private Integer i = 0;

Q: 19 Click the Task button.

Replace two of the Modifiers that appear in the Single class to make the code compile. Note: Three modifiers will not be used and four modifiers in the code will remain unchanged.

```
Code
                                                           Modifiers
public class Single {
                                                            final
              static Single instance;
  private
   public
            static | Single getInstance() {
                                                           private
      if (instance == null) instance = create();
     return instance;
                                                           abstract
                                                            static
  private Single() { }
 protected Single create() { return new Single(); }
}
                                                              Done
class SingleSub extends Single {
```



```
Q: 20 Given:

12. public class Test {
13. public enum Dogs {collie, harrier};
14. public static void main(String [] args) {
15. Dogs myDog = Dogs.collie;
16. switch (myDog) {
17. case collie:
18. System.out.print("collie ");
19. case harrier:
20. System.out.print("harrier ");
21. }
22. }
23. }
What is the result?
```

A. collie B. harrier

C. Compilation fails. D. collie harrier

E. An exception is thrown at runtime.

Q: 21 Click the Exhibit button.

Given:

ClassA a = new ClassA();

a.methodA();

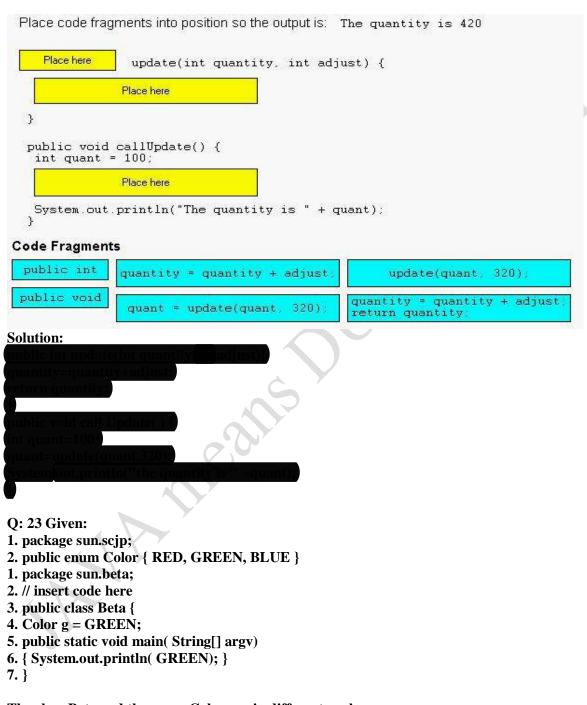
What is the result?

```
10. public class ClassA {
11. public void methodA() {
12. ClassB classB = new ClassB();
12.
13.
           classB.getValue();
14.
15. }
And:
20. class ClassB {
21.
22.
       public ClassC classC;
        public String getValue() {
   return classC.getValue();
23.
24.
25.
26. }
And:
30. class ClassC {
31.
      public String value;
32.
        public String getValue() {
  value = "ClassB";
33.
34.
35.
           return value;
36.
37. }
```

- A. Compilation fails.
- B. ClassC is displayed.
- C. The code runs with no output.
- D. An exception is thrown at runtime.



Q: 22 Click the Task button.



The class Beta and the enum Color are in different packages. Which two code fragments, inserted individually at line 2 of the Beta declaration, will allow this code to compile? (Choose two.)

A. import sun.scjp.Color.*;

```
B. import static sun.scjp.Color.*;
C. import sun.scjp.Color; import static sun.scjp.Color.*;
D. import sun.scjp.*; import static sun.scjp.Color.*;
E. import sun.scjp.Color; import static sun.scjp.Color.GREEN;
Ouestion 24
Given:
10. public class Fabric
11. public enum Color {
12. RED(0xff0000), GREEN(0x00ff00), BLUE(0x0000ff);
13. private final int rgb;
14. Color( int rgb) { this.rgb = rgb; }
15. public int getRGB() { return rgb; }
16. };
17. public static void main(String[] argv) {
18. // insert code here
19. }
20. }
Which two code fragments, inserted independently at line 18, allow the
Fabric class to compile? (Choose two.)
A. Color skyColor = BLUE;
B. Color treeColor = Color.GREEN;
C. Color purple = new Color( 0xff00ff);
D. if( RED.getRGB() < BLUE.getRGB() ) {}
E. Color purple = Color.BLUE + Color.RED;
F. if(Color.RED.ordinal() < Color.BLUE.ordinal()) {}
Question 25
Given:
11. public class Rainbow {
12. public enum MyColor {
13. RED(0xff0000), GREEN(0x00ff00), BLUE(0x0000ff);
14. private final int rgb;
15. MyColor(int rgb) { this.rgb = rgb; }
16. public int getRGB() { return rgb; }
17. };
18. public static void main(String[] args) {
19. // insert code here
20. }
21. }
Which code fragment, inserted at line 19, allows the Rainbow class to compile?
A. MyColor skyColor = BLUE;
B. MyColor treeColor = MyColor.GREEN;
```

```
E. MyColor purple = new MyColor(0xff00ff):
F. MyColor purple = MyColor.BLUE + MyColor.RED;
Question 26
Given:
21. class Money {
22. private String country = "Canada";
23. public String getC() { return country; }
24. }
25. class Yen extends Money {
26. public String getC() { return super.country; }
27. }
28. public class Euro extends Money {
29. public String getC(int x) { return super.getC(); }
30. public static void main(String[] args) {
31. System.out.print(new Yen().getC() + " " + new Euro().getC());
32. }
33. }
What is the result?
A. Canada
B. null Canada
C. Canada null
D. Canada Canada
E. Compilation fails due to an error on line 26.
F. Compilation fails due to an error on line 29.
```

C. if(RED.getRGB() < BLUE.getRGB()) { }

D. Compilation fails due to other error(s) in the code.



