Module 2-类、接口以及枚举

一、选择题：

Question 1
Given:
11. public interface Status {
12. /* insert code here */ int MY_VALUE = 10;
13. }
Which three are valid on line 12? (Choose three.)
A. final
B. static
C. native
D. public
E. private
F. abstract
G. protected

Answer: ABD

Question 2
Given:
10. class Foo {
11. static void alpha() { /* more code here */ }
12. void beta() { /* more code here */ }
13. }
Which two are true? (Choose two.)
A. Foo.beta() is a valid invocation of beta().
B. Foo.alpha() is a valid invocation of alpha().
C. Method beta() can directly call method alpha().
D. Method alpha() can directly call method beta().

Answer: BC

Question 3
Click the Exhibit button.
11. class Payload {
12. private int weight;
13. public Payload(int wt) { weight = wt; }
14. public Payload() {}
15. public void setWeight(mt w) { weight = w; }
16. public String toString { return Integer.toString(weight); }
17. }
18.
19. public class TestPayload {
20. static void changePayload(Payload p) {
21. /* insert code here */

```
22. }
23.
24. public static void main(String[] args) {
25. Payload p = new Payload();
26. p.setWeight(1024);
27. changePayload(p);
28. System.out.println("The value of p is "+ p);
29. }
30. }
```

Which statement, placed at line 21, causes the code to print “The value of p is 420.”?

A. p.setWeight(420);
B. p.changePayload(420);
C. p = new Payload(420);
D. Payload.setWeight(420);
E. p = Payload.setWeight(420);
F. p = new Payload();
p.setWeight(420);

Answer: A

Question 4

Click the Exhibit button.

```
1. public class Item {
2. private String desc;
3. public String getDescription() { return desc; }
4. public void setDescription(String d) { desc = d; }
5.
6. public static void modifyDesc(Item item, String desc) {
7. item = new Item();
8. item.setDescription(desc);
9. }
10. public static void main(String[] args) {
11. Item it = new Item();
12. it.setDescription("Gobstopper");
13. Item it2 = new Item();
14. it2.setDescription("Fizzylifting");
15. modifyDesc(it, "Scrumdiddlyumptious");
16. System.out.println(it.getDescription());
17. System.out.println(it2.getDescription());
18. }
19. }
```

What is the outcome of the code?

A. Compilation fails.
B. Gobstopper
Fizzylifting

C. Gobstopper
Scrumdiddlyumptious
D. Scrumdiddlyumptious
Fizzylifltng
E. Scrumdiddlyumptious
Scrumdiddlyumptious

Answer: B

Question 5
Given:
11. public class ItemTest {
12. private final int id;
13. public ItemTest(int id) { this.id = id; }
14. public void updateId(int newId) { id = newId; }
15.
16. public static void main(String[] args) {
17. ItemTest fa = new ItemTest(42);
18. fa.updateId(69);
19. System.out.println(fa.id);
20. }
21. }
What is the result?
A. Compilation fails.
B. An exception is thrown at runtime.
C. The attribute id in the Item object remains unchanged.
D. The attribute id in the Item object is modified to the new value.
E. A new Item object is created with the preferred value in the id attribute.

Answer: A

Question 6
Click the Exhibit button.
10. class Inner {
11. private int x;
12. public void setX(int x) { this.x = x; }
13. public int getX() { return x; }
14. }
15.
16. class Outer {
17. private Inner y;
18. public void setY(Inner y) { this.y = y; }
19. public Inner getY() { return y; }
20. }
21.

```
22. public class Gamma {
23. public static void main( String[] args) {
24. Outer o = new Outer();
25. Inner i = new Inner();
26.int n=10;
27. i.setX(n);
28. o.setY(i);
29. // insert code here
30. System.out.println( o.getY().getX());
31. }
32. }
```

Which three code fragments, added individually at line 29, produce the output 100? (Choose three.)

A. n = 100;
B. i.setX(100);
C. o.getY().setX(100);
D. i = new Inner(); i.setX(100);
E. o.setY(i); i = new Inner(); i.setX(100);
F. i = new Inner(); i.setX(100); o.setY(i);

Answer: BCF

Question 7

Click the Exhibit button.

```
10. class Foo {
11. private int x;
12. public Foo(int x) {this.x=x; }
13. public void setX( int x) { this.x = x; }
14. public int getX() { return x; }
15. }
16.
17. public class Gamma {
18.
19. static Foo fooBar( Foo foo) {
20. foo = new Foo( 100);
21. return foo;
22. }
23.
24. public static void main( String[] args) {
25. Foo foo = new Foo( 300);
26. System.out.print( foo.getX() + "-");
27.
28. Foo fooFoo = fooBar( foo);
29. System.out.print( foo.getX() + "-");
30. System.out.print( fooFoo.getX() + "-");
31.
```

32. foo = fooBar(fooFoo);
33. System.out.print(foo.getX() + "-");
34. System.out.prmt(fooFoo.getX());
35. }
36. }
What is the output of this program?
A. 300-100-100-100-100
B. 300-300-100-100-100
C. 300-300-300-100-100
D. 300-300-300-300-100

Answer: B

Question 8
Given:
1. interface DoStuff2 {
2. float getRange(int low, int high); }
3.
4. interface DoMore {
5. float getAvg(int a, int b, int c); }
6.
7. abstract class DoAbstract implements DoStuff2, DoMore { }
8.
9. class DoStuff implements DoStuff2 {
10. public float getRange(int x, int y) { return 3.14f; } }
11.
12. interface DoAll extends DoMore {
13. float getAvg(int a, int b, int c, int d); }
What is the result?
A. The file will compile without error.
B. Compilation fails. Only line 7 contains an error.
C. Compilation fails. Only line 12 contains an error.
D. Compilation fails. Only line 13 contains an error.
E. Compilation fails. Only lines 7 and 12 contain errors.
F. Compilation fails. Only lines 7 and 13 contain errors.
G. Compilation fails. Lines 7, 12, and 13 contain errors.

Answer: A

Question 9
Click the Exhibit button.
1. public class A {
2.
3. private int counter = 0;
4.
5. public static int getInstanceCount() {

```
6. return counter;
7. }
8.
9. public A() {
10. counter++;
11. }
12.
13. }
```

Given this code from Class B:

```
25.A a1 =new A();
26. A a2 =new A();
27. A a3 =new A();
28. System.out.printIn(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.

Answer: A

Question 10

Given:

```
1. public class A {
2. public void doit() {
3. }
4. public String doit() {
5. return “a”;
6. }
7. public double doit(int x) {
8. return 1.0;
9. }
10.}
```

What is the result?

A. An exception is thrown at runtime.
B. Compilation fails because of an error in line 7.
C. Compilation fails because of an error in line 4.
D. Compilation succeeds and no runtime errors with class A occur.

Answer: C

Question 11

Given:

```
10. class Line {
11. public static class Point { }
```

```
12. }
13.
14. class Triangle {
15. // insert code here
16. }
```

Which code, inserted at line 15, creates an instance of the Point class defined in Line?

A. Point p = new Point();
B. Line.Point p = new Line.Point();
C. The Point class cannot be instatiated at line 15.
D. Line 1 = new Line() ; 1.Point p = new 1.Point();

Answer: B

Question 12

Click the Exhibit button.

```
10. interface Foo {
11. int bar();
12. }
13.
14. public class Beta {
15.
16. class A implements Foo {
17. public int bar() { return 1; }
18. }
19.
20. public int fubar( Foo foo) { return foo.bar(); }
21.
22. public void testFoo() {
23.
24. class A implements Foo {
25. public int bar() { return 2; }
26. }
27.
28. System.out.println( fubar( new A()));
29. }
30.
31. public static void main( String[] argv) {
32. new Beta().testFoo();
33. }
34. }
```

Which three statements are true? (Choose three.)

A. Compilation fails.
B. The code compiles and the output is 2.
C. If lines 16, 17 and 18 were removed, compilation would fail.
D. If lines 24, 25 and 26 were removed, compilation would fail.

E. If lines 16, 17 and 18 were removed, the code would compile and the output would be 2.
F. If lines 24, 25 and 26 were removed, the code would compile and the output would be 1.

Answer: BEF

Question 13
Given:
1. public interface A {
2. String DEFAULT_GREETING = “Hello World”;
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 { }

Answer: A

Question 14
Given:
1. class TestA {
2. public void start() { System.out.println(”TestA”); }
3. }
4. public class TestB extends TestA {
5. public void start() { System.out.println(”TestB”); }
6. public static void main(String[] args) {
7. ((TestA)new TestB()).start();
8. }
9. }
What is the result?
A. TestA
B. TestB
C. Compilation fails.
D. An exception is thrown at runtime.

Answer: B

Question 15
Given:
11. public abstract class Shape {
12. int x;
13. int y;

14. public abstract void draw();
15. public void setAnchor(int x, int y) {
16. this.x = x;
17. this.y = y;
18. }
19. }
and a class Circle that extends and fully implements the Shape class.

Which is correct?
A. Shape s = new Shape();
s.setAnchor(10,10);
s.draw();

B. Circle c = new Shape();
c.setAnchor(10,10);
c.draw();

C. Shape s = new Circle();
s.setAnchor(10,10);
s.draw();

D. Shape s = new Circle();
s->setAnchor(10,10);
s->draw();

E. Circle c = new Circle();
c.Shape.setAnchor(10,10);
c.Shape.draw();

Answer: C

Question 16
Given:
10. abstract public class Employee {
11. protected abstract double getSalesAmount();
12. public double getCommision() {
13. return getSalesAmount() * 0.15;
14. }
15. }
16. class Sales extends Employee {
17. // insert method here
18. }
Which two methods, inserted independently at line 17, correctly complete the Sales class? (Choose two.)

A. double getSalesAmount() { return 1230.45; }

B. public double getSalesAmount() { return 1230.45; }
C. private double getSalesAmount() { return 1230.45; }
D. protected double getSalesAmount() { return 1230.45; }

Answer: BD

Question 17
Given:
10. interface Data { public void load(); }
11. abstract class Info { public abstract void load(); }
Which class correctly uses the Data interface and Info class?

A. public class Employee extends Info implements Data {
public void load() { /*do something*/ }
}

B. public class Employee implements Info extends Data {
public void load() { /*do something*/ }
}

C. public class Employee extends Info implements Data {
public void load() { /*do something */ }
public void Info.load() { /*do something*/ }
}

D. public class Employee implements Info extends Data {
public void Data.load() { /*d something */ }
public void load() { /*do something */ }
}

E. public class Employee implements Info extends Data {
public void load() { /*do something */ }
public void Info.load(){ /*do something*/ }
}

F. public class Employee extends Info implements Data{
public void Data.load() { /*do something*/ }
public void Info.load() { /*do something*/ }
}

Answer: A

Question 18
Given:
11. public abstract class Shape {
12. private int x;

13. private int y;
14. public abstract void draw();
15. public void setAnchor(int x, int y) {
16. this.x = x;
17. this.y = y;
18. }
19. }
Which two classes use the Shape class correctly? (Choose two.)

A. public class Circle implements Shape {
private int radius;
}

B. public abstract class Circle extends Shape {
private int radius;
}

C. public class Circle extends Shape {
private int radius;
public void draw();
}

D. public abstract class Circle implements Shape {
private int radius;
public void draw();
}

E. public class Circle extends Shape {
private int radius;
public void draw() {/* code here */}
}

F. public abstract class Circle implements Shape {
private int radius;
public void draw() { / code here */ }
}

Answer: BE

Question 19
Which two classes correctly implement both the java.lang.Runnable and the java.lang.Clonable interfaces? (Choose two.)

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 */ }
}
```

Answer: CD

Question 20
Given:
```
10. class One {
11. public One() { System.out.print(1); }
12. }
13. class Two extends One {
14. public Two() { System.out.print(2); }
15. }
16. class Three extends Two {
17. public Three() { System.out.print(3); }
18. }
19. public class Numbers{
20. public static void main( String[] argv) { new Three(); }
21. }
```
What is the result when this code is executed?
A. 1
B. 3

C. 123
D. 321
E. The code rims with no output.

Answer: C

Question 21
Given classes defined in two different files:
1. package packageA;
2. public class Message {
3. String getText() { return “text”; }
4. }
and:
1. package packageB;
2. public class XMLMessage extends packageA.Message {
3. String getText() { return “<msg>text</msg>”; }
4. public static void main(String[] args) {
5. System.out.println(new XMLMessage().getText());
6. }
7. }
What is the result of executing XMLMessage.main?
A. text
B. <msg>text</msg>
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 2 of XMLMessage.
E. Compilation fails because of an error in line 3 of XMLMessage.

Answer: B

Question 22
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.

Answer: D

Question 23
41. Given:
10. class One {
11. public One foo() { return this; }
12. }
13. class Two extends One {
14. public One foo() { return this; }
15. }
16. class Three extends Two {
17. // insert method here
18. }
Which two methods, inserted individually, correctly complete the Three class? (Choose two.)
A. public void foo() { }
B. public int foo() { return 3; }
C. public Two foo() { return this; }
D. public One foo() { return this; }
E. public Object foo() { return this; }

Answer: CD

Question 24
Given:
10. class One {
11. void foo() {}
12. }
13. class Two extends One {
14. //insert method here
15. }
Which three methods, inserted individually at line 14, will correctly complete class Two? (Choose three.)
A. int foo() { /* more code here */ }
B. void foo() { /* more code here */ }
C. public void foo() { /* more code here */ }
D. private void foo() { /* more code here */ }
E. protected void foo() { /* more code here */ }

Answer: BCE

Question 25
Click the Exhibit button.

```
1. public interface A {
2. public void doSomething(String thing);
3. }
1. public class AImpl implements A {
2. public void doSomething(String msg) { }
3. }
1. public class B {
2. public A doit() {
3. // more code here
4. }
5.
6. public String execute() {
7. // more code here
8. }
9. }

1. public class C extends B {
2. public AImpl doit() {
3. // more code here
4. }
5.
6. public Object execute() {
7. // more code here
8. }
9. }
```

Which statement is true about the classes and interfaces in the exhibit?

A. Compilation will succeed for all classes and interfaces.
B. Compilation of class C will fail because of an error in line 2.
C. Compilation of class C will fail because of an error in line 6.
D. Compilation of class AImpl will fail because of an error in line 2.

Answer: C

Question 26
Click the Exhibit button.

```
11. class Person {
12. String name ="No name";
13. public Person(String nm) { name = nm; }
14. }
15.
16. class Employee extends Person {
17. String empID = “0000”;
```

18. public Employee(String id) { empID = id; }
19. }
20.
21. public class EmployeeTest {
22. public static void main(String[] args) {
23. Employee e = new Employee(”4321”);
24. System.out.println(e.empID);
25. }
26. }
What is the result?
A. 4321
B. 0000
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 18.

Answer: D

Question 27
Given:
1. public class Plant {
2. private String name;
3. public Plant(String name) { this.name = name; }
4. public String getName() { return name; }
5. }
1. public class Tree extends Plant {
2. public void growFruit() { }
3. public void dropLeaves() { }
4. }
Which is true?
A. The code will compile without changes.
B. The code will compile if public Tree() { Plant(); } is added to the Tree class.
C. The code will compile if public Plant() { Tree(); } is added to the Plant class.
D. The code will compile if public Plant() { this(”fern”); } is added to the Plant class.
E. The code will compile if public Plant() { Plant(”fern”); } is added to the Plant class.

Answer: D

Question 28
Click the Exhibit button.
11. public class Bootchy {
12. int bootch;
13. String snootch;

```
14.
15. public Bootchy() {
16. this(”snootchy”);
17. System.out.print(”first “);
18. }
19.
20. public Bootchy(String snootch) {
21. this(420, "snootchy");
22. System.out.print("second ");
23. }
24.
25. public Bootchy(int bootch, String snootch) {
26. this.bootch = bootch;
27. this.snootch = snootch;
28. System.out.print("third ");
29. }
30.
31. public static void main(String[] args) {
32. Bootchy b = new Bootchy();"
33. System.out.print(b.snootch +" " + b.bootch);
34. }
35. }
```

What is the result?

A. snootchy 420 third second first
B. snootchy 420 first second third
C. first second third snootchy 420
D. third second first siiootchy 420
E. third first second snootchy 420
F. first second first third snootchy 420

Answer: D

Question 29

Given:

```
1. interface TestA { String toString(); }
2. public class Test {
3. public static void main(String[] args) {
4. System.out.println(new TestA() {
5. public String toString() { return "test"; }
6. });
7. }
8. }
```

What is the result?

A. test
B. null
C. An exception is thrown at runtime.

D. Compilation fails because of an error in line 1.
E. Compilation fails because of an error in line 4.
F. Compilation fails because of an error in line 5.

Answer: A

Question 30
Given:
10. interface Foo { int bar(); }
11. public class Sprite {
12. public int fubar(Foo foo) { return foo.bar(); }
13. public void testFoo() {
14. fubar(
15. // insert code here
16.);
17. }
18. }
Which code, inserted at line 15, allows the class Sprite to compile?
A. Foo { public int bar() { return 1; } }
B. new Foo { public int bar() { return 1; } }
C. new Foo() { public int bar(){return 1; } }
D. new class Foo { public int bar() { return 1; } }

Answer: C

Question 31
Given:
10. class Line {
11. public class Point { public int x,y; }
12. public Point getPoint() { return new Point(); }
13. }
14. class Triangle {
15. public Triangle() {
16. // insert code here
17. }
18. }
Which code, inserted at line 16, correctly retrieves a local instance of a Point object?
A. Point p = Line.getPoint();
B. Line.Point p = Line.getPoint();
C. Point p = (new Line()).getPoint();
D. Line.Point p = (new Line()).getPoint();

Answer: D

Question 32

A JavaBeans component has the following field:
11. private boolean enabled;
Which two pairs of method declarations follow the JavaBeans standard for accessing this field? (Choose two.)

A. public void setEnabled(boolean enabled)
public boolean getEnabled()

B. public void setEnabled(boolean enabled)
public void isEnabled()

C. public void setEnabled(boolean enabled)
public boolean isEnabled()

D. public boolean setEnabled(boolean enabled)
public boolean getEnabled()

Answer: AC

Question 33
A programmer is designing a class to encapsulate the information about an inventory item. A JavaBeans component is needed to do this. The Inventoryltem class has private instance variables to store the item information:
10. private int itemId;
11. private String name;
12. private String description;
Which method signature follows the JavaBeans naming standards for modifying the itemld instance variable?

A. itemID(int itemId)
B. update(int itemId)
C. setItemId(int itemId)
D. mutateItemId(int itemId)
E. updateItemID(int itemId)

Answer: C

Question 34
Given:
10. class Nav{
11. public enum Direction { NORTH, SOUTH, EAST, WEST }
12. }
13. public class Sprite{
14. // insert code here

15. }
Which code, inserted at line 14, allows the Sprite class to compile?
A. Direction d = NORTH;
B. Nav.Direction d = NORTH;
C. Direction d = Direction.NORTH;
D. Nav.Direction d = Nav.Direction.NORTH;

Answer: D

Question 35
Given:
1. package sun.scjp;
2. public enum Color { RED, GREEN, BLUE }
1. package sun.beta;
2. // insert code here
3. public class Beta {
4. Color g = GREEN;
5. public static void main(String[] argv)
6. { System.out.println(GREEN); }
7. }
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;

Answer: CE

Question 36
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 <= BLUE; c++)
C. for(Color c; c.hasNext() ; c.next())

D. for(Color c = Color[0]; c <= Color[2]; c++)
E. for(Color c = Color.RED; c <= Color.BLUE; c++)

Answer: A

Question 37
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.

Answer: A

Question 38
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()) {}

Answer: BF

Question 39
Given:
11. public class Test {
12. public enum Dogs {collie, harrier, shepherd};
13. public static void main(String [] args) {
14. Dogs myDog = Dogs.shepherd;
15. switch (myDog) {
16. case collie:
17. System.out.print("collie ");
18. case default:
19. System.out.print("retriever ");
20. case harrier:
21. System.out.print("harrier ");
22. }
23. }
24. }
What is the result?
A. harrier
B. shepherd
C. retriever
D. Compilation fails.
E. retriever harrier
F. An exception is thrown at runtime.

Answer: D

Question 40
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?

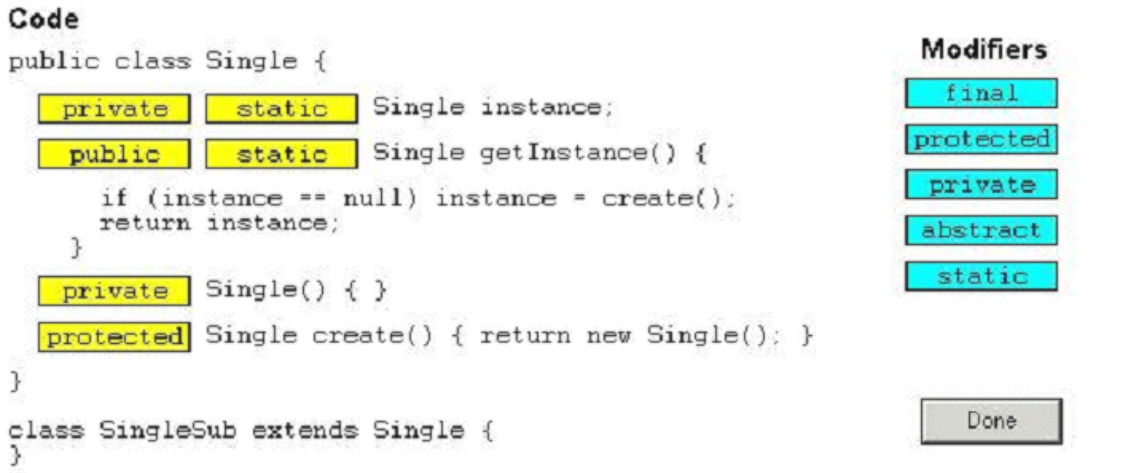
A. collie
B. harrier
C. Compilation fails.
D. collie harrier
E. An exception is thrown at runtime.

Answer: D

二、拖拽题：

Question 1:

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.

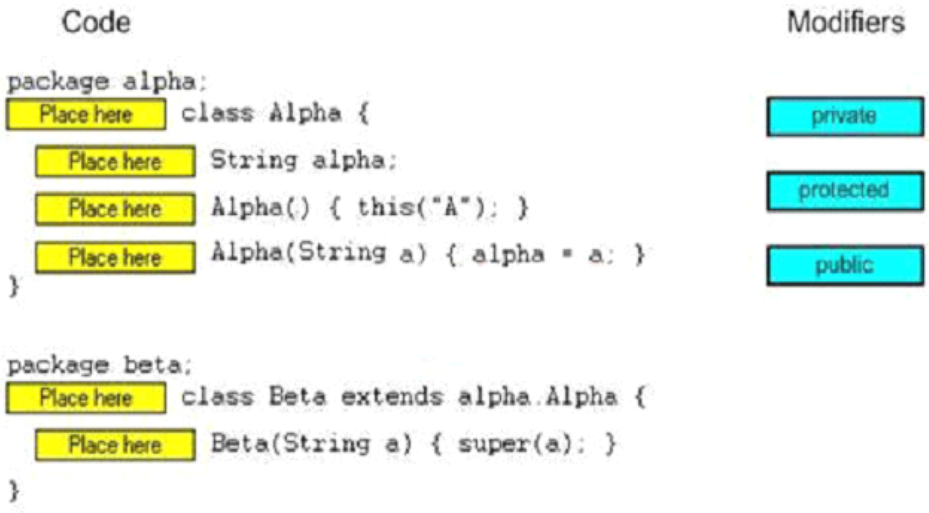


Answer:

```
public calss Single{
    private static Single instance;
    public static Single getInstance(){
        if(instance==null) instance=create();
        return instance;
    }
    protected Single(){}
    static Single create(){return new Single();}
}
```

Question 2:

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.



Answer:

```
public class Alpha {
        private String alpha;
        public Alpha(){
                this("A");
        }
        protected Alpha(String a){
                this.alpha=a;
        }
}

public class Beta extends alpha.Alpha{
        public Beta(String a){
                super(a);
        }
}
```

Question 3:

Place the code fragments in position to complete the Displayable interface.

```
interface Reloadable {
  public void reload();
}

class Edit {
  public void edit() { /* Edit Here */ }
}

interface Displayable
```

[Place here] [Place here] {

[Place here]

```
}
```

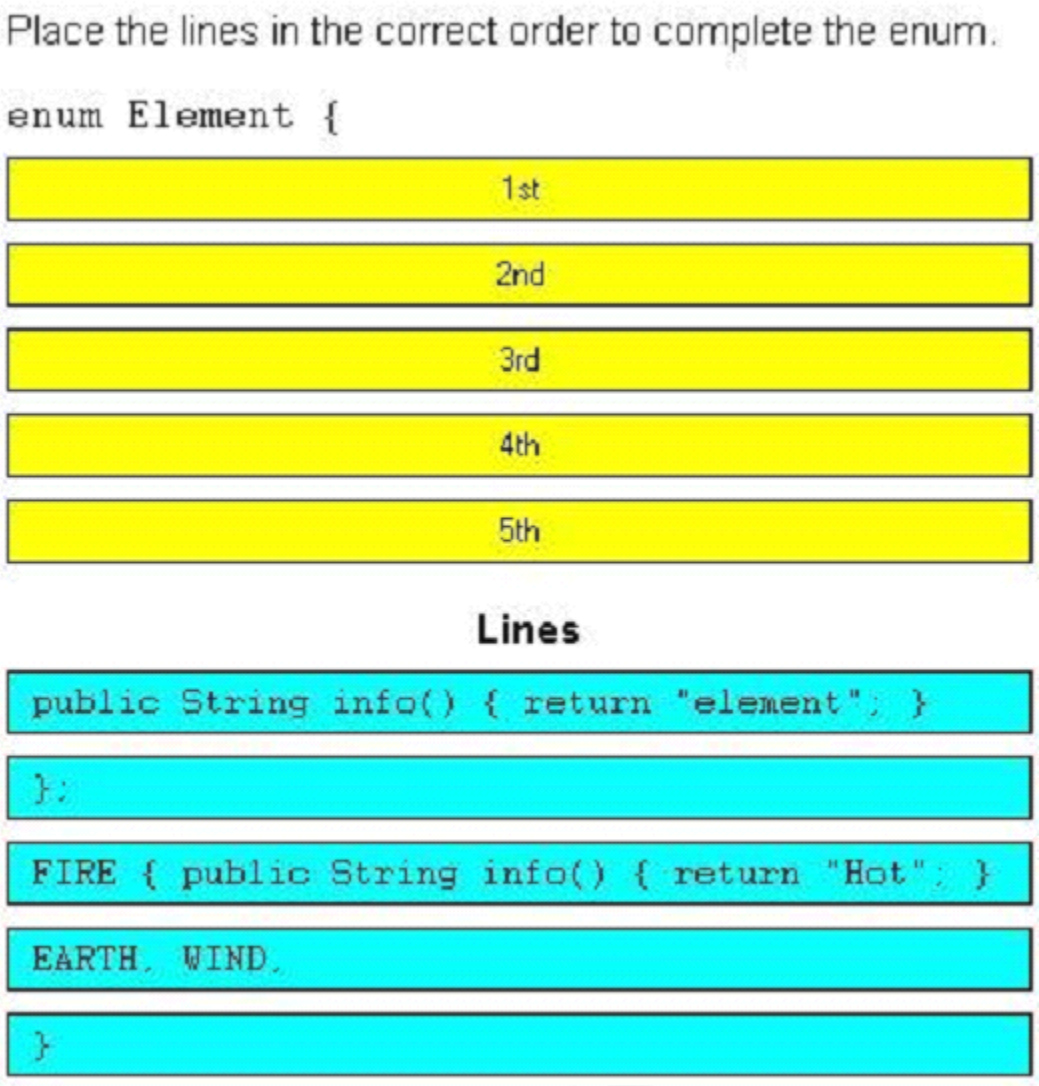
Code Fragments

extends	public void display();	Reloadable
implements	public void display() { /* Display*/ };	Edit

Answer:

```
interface Displayable extends Reloadable{
      public void display();
}
```

Question 4:



Answer:

```
enum Element{
    EARTH,WIND,
    FIRE{public String info(){return "Hot";}
    };
    public String info(){return "element";}
 }
```