**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

**Question 5 :**

Given:

10. interface Foo {}

11. class Alpha implements Foo { }

12. class Beta extends Alpha {}

13. class Delta extends Beta {

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

15. Beta x = new Beta();

16. // insert code here

17. }

18. }

Which code, inserted at line 16, will cause a java.lang.ClassCastException?

A. Alpha a = x;

B. Foo f= (Delta)x;

C. Foo f= (Alpha)x;

D. Beta b = (Beta)(Alpha)x;

**Question 7 :**

Given:

20. public class CreditCard {

21.

22. private String cardlD;

23. private Integer limit;

24. public String ownerName;

25.

26. public void setCardlnformation(String cardlD,

27. String ownerName,

28. Integer limit) {

29. this.cardlD = cardlD;

30. this.ownerName = ownerName;

31. this.limit = limit;

32. }

33. }

Which is true?

A. The class is fully encapsulated.

B. The code demonstrates polymorphism.

C. The ownerName variable breaks encapsulation.

D. The cardlD and limit variables break polymorphism.

E. The setCardlnformation method breaks encapsulation.

**Question 8 :**

Assume that country is set for each class.

Given:

10. public class Money {

11. private String country, name;

12. public String getCountry() { return country; }

13. }

and:

24. class Yen extends Money {

25. public String getCountry() { return super.country; }

26. }

27.

28. class Euro extends Money {

29. public String getCountry(String timeZone) {

30. return super.getCountry();

31. }

32. }

Which two are correct? (Choose two.)

A. Yen returns correct values.

B. Euro returns correct values.

C. An exception is thrown at runtime.

D. Yen and Euro both return correct values.

E. Compilation fails because of an error at line 25.

F. Compilation fails because of an error at line 30.

**Question 9 :**

Which Man class properly represents the relationship “Man has a best friend who is a Dog”?

A. class Man extends Dog { }

B. class Man implements Dog { }

C. class Man { private BestFriend dog; }

D. class Man { private Dog bestFriend; }

E. class Man { private Dog<bestFriend> }

F. class Man { private BestFriend<dog> }

**Question 12 :**

12. Given:

13. public class Pass {

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

15. int x = 5;

16. Pass p = new Pass();

17. p.doStuff(x);

18. System.out.print(” main x = “+ x);

19. }

20.

21. void doStuff(int x) {

22. System.out.print(” doStuff x = “+ x++);

23. }

24. }

What is the result?

A. Compilation fails.

B. An exception is thrown at runtime.

C. doStuffx = 6 main x = 6

D. doStuffx = 5 main x = 5

E. doStuffx = 5 main x = 6

F. doStuffx = 6 main x = 5

**Question 15 :**

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; } }

**Question 18 :**

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 { }

**Question 19 :**

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.

**Question 20 :**

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.

**Question 21 :**

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();

**Question 22 :**

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; }

**Question 23 :**

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\*/ }

**Question 24 :**

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 \*/ }

}

**Question 25 :**

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 \*/ }

}

**Question 27 :**

Given:

11. public static void parse(String str) {

12. try {

13. float f= Float.parseFloat(str);

14. } catch (NumberFormatException nfe) {

15. f = 0;

16. } finally {

17. System.out.println(f);

18. }

19. }

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

21. parse(”invalid”);

22. }

What is the result?

A. 0.0

B. Compilation fails.

C. A ParseException is thrown by the parse method at runtime.

D. A NumberFormatException is thrown by the parse method at runtime.

**Question 29 :**

Given:

55. int []x= {1,2,3,4,5};

56. int y[] = x;

57. System.out.println(y[2]);

Which is true? //사실인 것?

A. Line 57 will print the value 2.

B. Line 57 will print the value 3.

C. Compilation will fail because of an error in line 55.

D. Compilation will fail because of an error in line 56.

**Question 30 :**

Given:

35. String #name = “Jane Doe”;

36. int $age=24;

37. double \_height = 123.5;

38. double ~temp = 37.5;

Which two are true? (Choose two.)

A. Line 35 will not compile.

B. Line 36 will not compile.

C. Line 37 will not compile.

D. Line 38 will not compile.

**Question 35 :**

Given:

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

12. Object obj =new int[] { 1,2,3 };

13. int[] someArray = (int[])obj;

14. for (int i: someArray) System.out.print(i +“ “)

15. }

What is the result?

A. 1 2 3

B. Compilation fails because of an error in line 12.

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

D. Compilation fails because of an error in line 14.

E. A ClassCastException is thrown at runtime.

**Question 36 :**

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().

**Question 38 :**

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 itemId instance variable?

A. itemID(int itemId)

B. update(int itemId)

C. setItemId(int itemId)

D. mutateItemId(int itemId)

E. updateItemID(int itemId)

**Question 40 :**

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()

**Question 42 :**

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 \*/ }

**Question 45 :**

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.

**Question 46 :**

46. 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();

**Question 47 :**

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();

**Question 48 :**

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.

**Question 50 :**

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.

**Question 70 :**

Given:

33. try {

34. // some code here

35. } catch (NullPointerException e1) {

36. System.out.print(”a”);

37. } catch (RuntimeException e2) {

38. System.out.print(”b”);

39. } finally {

40. System.out.print(”c”);

41. }

What is the result if a NullPointerException occurs on line 34?

A. c

B. a

C. ab

D. ac

E. bc

F. abc

**Question 72 :**

Given:

11.classA {

12. public void process() { System.out.print(”A “); } }

13. class B extends A {

14. public void process() throws RuntimeException {

15. super.process();

16. if (true) throw new RuntimeException();

17. System.out.print(“B”); }}

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

19. try { ((A)new B()).process(); }

20. catch (Exception e) { System.out.print(”Exception “); }

21. }

What is the result?

A. Exception

B. A Exception

C. A Exception B

D. A B Exception

E. Compilation fails because of an error in line 14.

F. Compilation fails because of an error in line 19.

**Question 73 :**

Given:

11. static class A {

12. void process() throws Exception { throw new Exception(); }

13. }

14. static class B extends A {

15. void process() { System.out.println(”B “); }

16. }

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

18. A a=new B();

19. a.process();

20. }

What is the result?

A. B

B. The code runs with no output.

C. An exception is thrown at runtime.

D. Compilation fails because of an error in line 15.

E. Compilation fails because of an error in line 18.

F. Compilation fails because of an error in line 19.

**Question 78 :**

Given:

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

12. try {

13. args=null;

14. args[0] = “test”;

15. System.out.println(args[0]);

16. } catch (Exception ex) {

17. System.out.println(”Exception”);

18. } catch (NullPointerException npe) {

19. System.out.println(”NullPointerException”);

20. }

21. }

What is the result?

A. test

B. Exception

C. Compilation fails.

D. NullPointerException

**Question 80 :**

Given:

11. static void test() {

12. try {

13. String x=null;

14. System.out.print(x.toString() +“ “);

15. }

16. finally { System.out.print(“finally “); }

17. }

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

19. try { test(); }

20. catch (Exception ex) { System.out.print(”exception “); }

21. }

What is the result?

A. null

B. finally

C. null finally

D. Compilation fails.

E. finally exception

**Question 84 :**

Given:

10. public class Foo {

11. static int[] a;

12. static { a[0]=2; }

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

14. }

Which exception or error will be thrown when a programmer attempts to run this code?

A. java.lang.StackOverflowError

B. java.lang.IllegalStateException

C. java.lang.ExceptionlnlnitializerError

D. java.lang.ArraylndexOutOfBoundsException

**Question 93 :**

Given:

11. public class Yikes {

12.

13. public static void go(Long n) {System.out.println(”Long “);}

14. public static void go(Short n) {System.out.println(”Short “);}

15. public static void go(int n) {System.out.println(”int “);}

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

17. short y= 6;

18. long z= 7;

19. go(y);

20. go(z);

21. }

22. }

What is the result?

A. int Long

B. Short Long

C. Compilation fails.

D. An exception is thrown at runtime.

**Question 94 :**

Given:

12. public class Wow {

13. public static void go(short n) {System.out.println(”short”); }

14. public static void go(Short n) {System.out.println(”SHORT”);}

15. public static void go(Long n) {System.out.println(” LONG”); }

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

17. Short y= 6;

18. int z=7;

19. go(y);

20. go(z);

21. }

22. }

What is the result?

A. short LONG

B. SHORT LONG

C. Compilation fails.

D. An exception is thrown at runtime.

**Question 114 :**

Which two code fragments will execute the method doStuff() in a separate thread? (Choose two.)

A. new Thread() {

public void run() { doStuff(); }

}

B. new Thread() {

public void start() { doStuff(); }

}

C. new Thread() {

public void start() { doStuff(); }

} .run();

D. new Thread() {

public void run() { doStuff(); }

} .start();

E. new Thread(new Runnable() {

public void run() { doStuff(); }

} ).run();

F. new Thread(new Runnable() {

public void run() { doStuff(); }

}).start();

**Question 115 :**

Given:

1. public class Threads3 implements Runnable {

2. public void run() {

3. System.out.print(”running”);

4. }

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

6. Thread t = new Thread(new Threads3());

7. t.run();

8. t.run();

9. t.start();

10. }

11. }

What is the result?

A. Compilation fails.

B. An exception is thrown at runtime.

C. The code executes and prints “running”.

D. The code executes and prints “runningrunning”.

E. The code executes and prints “runningrunningrunning”.

**Question 117 :**

Given:

1. public class Threads5 {

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

3. new Thread(new Runnable() {

4. public void run() {

5. System.out.print(”bar”);

6. }}).start();

7. }

8. }

What is the result?

A. Compilation fails.

B. An exception is thrown at runtime.

C. The code executes normally and prints “bar”.

D. The code executes normally, but nothing prints.

**Question 118 :**

Given:

11. Runnable r = new Runnable() {

12. public void run() {

13. System.out.print(”Cat”);

14. }

15. };

16. Thread t=new Thread(r) {

17. public void run() {

18. System.out.print(”Dog”);

19. }

20. };

21. t.start();

What is the result?

A. Cat

B. Dog

C. Compilation fails.

D. The code runs with no output.

E. An exception is thrown at runtime.

**Question 121 :**

Given:

1. public class TestOne {

2. public static void main (String[] args) throws Exception {

3. Thread.sleep(3000);

4. System.out.println(”sleep”);

5. }

6. }

What is the result?

A. Compilation fails.

B. An exception is thrown at runtime.

C. The code executes normally and prints “sleep”.

D. The code executes normally, but nothing is printed.

**Question 138 :**

Given:

11. class ClassA {}

12. class ClassB extends ClassA {}

13. class ClassC extends ClassA {}

and:

21. ClassA p0 = new ClassA();

22. ClassB p1 = new ClassB();

23. ClassC p2 = new ClassC();

24. ClassA p3 = new ClassB();

25. ClassA p4 = new ClassC();

Which three are valid? (Choose three.)

A. p0 = p1;

B. p1 = p2;

C. p2 = p4;

D. p2 = (ClassC)p1;

E. p1 = (ClassB)p3;

F. p2 = (ClassC)p4;

**Question 139 :**

Given:

11. class Animal { public String noise() { return “peep”; } }

12. class Dog extends Animal { //Animal을 상속받았다.

13. public String noise() { return “bark”; }

14. }

15. class Cat extends Animal { //Animal을 상속받았다.

16. public String noise() { return “meow”; }

17. }

.....

30. Animal animal = new Dog();

31. Cat cat = (Cat)animal; //에러 : ClassCastException

32. System.out.println(cat.noise());

What is the result?

A. peep

B. bark

C. meow

D. Compilation fails.

E. An exception is thrown at runtime.

**Question 140 :**

Given:

11. abstract class Vehicle { public int speed() { return 0; } }

12. class Car extends Vehicle { public int speed() { return 60; } }

13. class RaceCar extends Car { public int speed() { return 150; }}

......

21. RaceCar racer = new RaceCar();

22. Car car = new RaceCar();

23. Vehicle vehicle = new RaceCar();

24. System.out.println(racer.speed() + “, ‘ + car.speed()

25. + “, “+ vehicle.speed());

What is the result?

A. 0, 0,0

B. 150, 60, 0

C. Compilation fails.

D. 150, 150, 150

E. An exception is thrown at runtime.

**Question 141 :**

Given:

10. abstract class A {

11. abstract void a1();

12. void a2() { }

13. }

14. class B extends A {

15. void a1() { }

16. void a2() { }

17. }

18. class C extends B { void c1() { } }

and:

A x = new B(); C y = new C(); A z = new C();

Which four are valid examples of polymorphic method calls? (Choose four.)

A. x.a2();

B. z.a2();

C. z.c1();

D. z.a1();

E. y.c1();

F. x.a1();

**Question 142 :**

Given:

10. interface A { void x(); }

11. class B implements A { public void x() { } public void y() { } }

12. class C extends B { public void x() {} }

And:

20. java.util.List<A> list = new java.util.ArrayList<A>();

21. list.add(new B());

22. list.add(new C());

23. for (A a:list) {

24. a.x();

25. a.y();

26. }

What is the result?

A. The code runs with no output.

B. An exception is thrown at runtime.

C. Compilation fails because of an error in line 20.

D. Compilation fails because of an error in line 21.

E. Compilation fails because of an error in line 23.

F. Compilation fails because of an error in line 25.

**Question 143**

Given:

1. class SuperClass {

2. public A getA() {

3. return new A();

4. }

5. }

6. class SubClass extends SuperClass {

7. public B getA() {

8. return new B();

9. }

10. }

Which is true?

A. Compilation will succeed if A extends B.

B. Compilation will succeed if B extends A.

C. Compilation will always fail because of an error in line 7.

D. Compilation will always fail because of an error in line 8.

**Question 144**

Given:

1. interface A { public void aMethod(); }

2. interface B { public void bMethod(); }

3. interface C extends A,B { public void cMethod(); }

4. class D implements B {

5. public void bMethod() { }

6. }

7. class E extends D implements C {

8. public void aMethod() { }

9. public void bMethod() { }

10. public void cMethod() { }

11. }

What is the result?

A. Compilation fails because of an error in line 3.

B. Compilation fails because of an error in line 7.

C. Compilation fails because of an error in line 9.

D. If you define D e = new E(), then e.bMethod() invokes the version of bMethod() defined in Line 5.

E. If you define D e = (D)(new E()), then e.bMethod() invokes the version of bMethod() defined in Line 5.

F. If you define D e = (D)(new E()), then e.bMethod() invokes the version of bMethod() defined in Line 9.

**Question 145 :**

Given:

10. interface A { public int getValue() }

11. class B implements A {

12. public int getValue() { return 1; }

13. }

14. class C extends B {

15. // insert code here

16. }

Which three code fragments, inserted individually at line 15, make use of polymorphism? (Choose three.)

A. public void add(C c) { c.getValue(); }

B. public void add(B b) { b.getValue(); }

C. public void add(A a) { a.getValue(); }

D. public void add(A a, B b) { a.getValue(); }

E. public void add(C c1, C c2) { c1.getValue(); }

**Question 146**

Given:

1. class ClassA {

2. public int numberOfinstances;

3. protected ClassA(int numberOfinstances) {

4. this.numberOflnstances = numberOfinstances;

5. }

6. }

7. public class ExtendedA extends ClassA {

8. private ExtendedA(int numberOfinstances) {

9. super(numberOflnstances);

10. }

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

12. ExtendedA ext = new ExtendedA(420);

13. System.out.print(ext.numberOflnstances);

14. }

15. }

Which is true?

A. 420 is the output.

B. An exception is thrown at runtime.

C. All constructors must be declared public.

D. Constructors CANNOT use the private modifier.

E. Constructors CANNOT use the protected modifier.

**Answer: A**

**Question 147**

147. Given:

1. public class Base {

2. 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. foofoofoofoofoo

B. foobarfoobarbar

C. foobarfoofoofoo

D. foobarfoobarfoo

E. barbarbarbarbar

F. foofoofoobarbar

G. foofoofoobarfoo

**Question 148 :**

Which three statements are true? (Choose three.)

A. A final method in class X can be abstract if and only if X is abstract.

B. A protected method in class X can be overridden by any subclass of X.

C. A private static method can be called only within other static methods in class X.

D. A non-static public final method in class X can be overridden in any subclass of X.

E. A public static method in class X can be called by a subclass of X without explicitly referencing the class X.

F. A method with the same signature as a private final method in class X can be implemented in a subclass of X.

G. A protected method in class X can be overridden by a subclass of A only if the subclass is in the same package as X.

**Question 149 :**

Given:

1. class Pizza {

2. java.util.ArrayList toppings;

3. public final void addTopping(String topping) {

4. toppings.add(topping);

5. }

6. }

7. public class PepperoniPizza extends Pizza {

8. public void addTopping(String topping) {

9. System.out.println(”Cannot add Toppings”);

10. }

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

12. Pizza pizza = new PepperoniPizza();

13. pizza.addTopping(”Mushrooms”);

14. }

15. }

What is the result?

A. Compilation fails.

B. Cannot add Toppings

C. The code runs with no output.

D. A NullPointerException is thrown in Line 4.

**Question 150 :**

Given:

1. class Super {

2. private int a;

3. protected Super(int a) { this.a = a; }

4. }

.....

11. class Sub extends Super {

12. public Sub(int a) { super(a); }

13. public Sub() { this.a= 5; }

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); }

**Question 155 :**

Given:

1. public class Blip {

2. protected int blipvert(int x) { return 0; }

3. }

4. class Vert extends Blip {

5. // insert code here

6. }

Which five methods, inserted independently at line 5, will compile? (Choose five.)

A. public int blipvert(int x) { return 0; }

B. private int blipvert(int x) { return 0; }

C. private int blipvert(long x) { return 0; }

D. protected long blipvert(int x) { return 0; }

E. protected int blipvert(long x) { return 0; }

F. protected long blipvert(long x) { return 0; }

G. protected long blipvert(int x, int y) { return 0; }

**Question156**

Given:

10. public class Foo {

11. public int a;

12. public Foo() { a = 3; } //Foo 객체가 생성될 때 Foo의 a값이 3이 된다.

13. public void addFive() { a += 5; }

14. }

and:

20. public class Bar extends Foo {

21. public int a;

22. public Bar() { a = 8; } //Bar 객체가 생성될 때 Bar의 a값이 8이 된다.

23. public void addFive() { this.a +=5; }

24. }

invoked with:

30. Foo foo = new Bar();

31. foo.addFive();

32. System.out.println(”Value: “+ foo.a);

What is the result?

A. Value: 3

B. Value: 8

C. Value: 13

D. Compilation fails.

E. The code runs with no output.

F. An exception is thrown at runtime.

**Question 157**

Given:

10. public class SuperCaic {

11. protected static int multiply(int a, int b) { return a \* b; }

12. }

and:

20. public class SubCalc extends SuperCalc {

21. public static int multiply(int a, int b) {

22. int c = super.multiply(a, b);

23. return c;

24. }

25. }

and:

30. SubCalc sc = new SubCalc();

31. System.out.println(sc.multiply(3,4));

32. System.out.println(SubCalc.multiply(2,2));

What is the result?

A. 12

4

B. The code runs with no output.

C. An exception is thrown at runtime.

D. Compilation fails because of an error in line 21.

E. Compilation fails because of an error in line 22.

F. Compilation fails because of an error in line 31.

**Question 158 :**

Given:

1. public class Team extends java.util.LinkedList {

2. public void addPlayer(Player p) {

3. add(p);

4. }

5. public void compete(Team opponent) { /\* more code here \*/ }

6. }

7. class Player { /\* more code here \*/ }

Which two are true? (Choose two.)

A. This code will compile.

B. This code demonstrates proper design of an is-a relationship.

C. This code demonstrates proper design of a has-a relationship.

D. A Java programmer using the Team class could remove Player objects from a Team object.

**Question 159 :**

Which four are true? (Choose four.)

A. Has-a relationships should never be encapsulated.

B. Has-a relationships should be implemented using inheritance.

C. Has-a relationships can be implemented using instance variables.

D. Is-a relationships can be implemented using the extends keyword.

E. Is-a relationships can be implemented using the implements keyword.

F. The relationship between Movie and Actress is an example of an is-a relationship.

G. An array or a collection can be used to implement a one-to-many has-a relationship.

**Question 160 :**

Which two are true about has-a and is-a relationships? (Choose two.)

A. Inheritance represents an is-a relationship.

B. Inheritance represents a has-a relationship.

C. Interfaces must be used when creating a has-a relationship.

D. Instance variables can be used when creating a has-a relationship.

**Question 161 :**

Given:

10. interface Jumper { public void jump(); }

......

20. class Animal {}

......

30. class Dog extends Animal {

31. Tail tail;

32. }

......

40. class Beagle extends Dog implements Jumper {

41. public void jump() { }

42. }

.......

50. class Cat implements Jumper {

51. public void jump() { }

52. }

Which three are true? (Choose three.)

A. Cat is-a Animal

B. Cat is-a Jumper

C. Dog is-a Animal

D. Dog is-a Jumper

E. Cat has-a Animal

F. Beagle has-a Tail

G. Beagle has-a Jumper

**Question 222 :**

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.