**1) What will be the output of the program?**

class A

{

final public int GetResult(int a, int b) { return 0; }

}

class B extends A

{

public int GetResult(int a, int b) {return 1; }

}

public class Test

{

public static void main(String args[])

{

B b = new B();

System.out.println("x = " + b.GetResult(0, 1));

}

}

A. x = 0

B. x = 1

C. Compilation fails.

D. An exception is thrown at runtime.

**2) What will be the output of the program?**

class SC2

{

public static void main(String [] args)

{

SC2 s = new SC2();

s.start();

}

void start()

{

int a = 3;

int b = 4;

System.out.print(" " + 7 + 2 + " ");

System.out.print(a + b);

System.out.print(" " + a + b + " ");

System.out.print(foo() + a + b + " ");

System.out.println(a + b + foo());

}

String foo()

{

return "foo";

}

}

A. 9 7 7 foo 7 7foo

B. 72 34 34 foo34 34foo

C. 9 7 7 foo34 34foo

D. 72 7 34 foo34 7foo

**3) What will be the output of the program?**

class BoolArray

{

boolean [] b = new boolean[3];

int count = 0;

void set(boolean [] x, int i)

{

x[i] = true;

++count;

}

public static void main(String [] args)

{

BoolArray ba = new BoolArray();

ba.set(ba.b, 0);

ba.set(ba.b, 2);

ba.test();

}

void test()

{

if ( b[0] && b[1] | b[2] )

count++;

if ( b[1] && b[(++count - 2)] )

count += 7;

System.out.println("count = " + count);

}

}

A. count = 0

B. count = 2

C. count = 3

D. count = 4

**4) What will be the output of the program?**

Float f = new Float("12");

switch (f)

{

case 12: System.out.println("Twelve");

case 0: System.out.println("Zero");

default: System.out.println("Default");

}

A.Zero

B.Twelve

C.Default

D.Compilation fails

**5)**

public class BoolTest

{

public static void main(String [] args)

{

Boolean b1 = new Boolean("false");

boolean b2;

b2 = b1.booleanValue();

if (!b2)

{

b2 = true;

System.out.print("x ");

}

if (b1 & b2) /\* Line 13 \*/

{

System.out.print("y ");

}

System.out.println("z");

}

}

A.z

B.x z

C.y z

D.Compilation fails.

**6) What will be the output of the program?**

class Two

{

byte x;

}

class PassO

{

public static void main(String [] args)

{

PassO p = new PassO();

p.start();

}

void start()

{

Two t = new Two();

System.out.print(t.x + " ");

Two t2 = fix(t);

System.out.println(t.x + " " + t2.x);

}

Two fix(Two tt)

{

tt.x = 42;

return tt;

}

}

A.null null 42

B.0 0 42

C.0 42 42

D.0 0 0

**7) What will be the output of the program?**

public class If2

{

static boolean b1, b2;

public static void main(String [] args)

{

int x = 0;

if ( !b1 ) /\* Line 7 \*/

{

if ( !b2 ) /\* Line 9 \*/

{

b1 = true;

x++;

if ( 5 > 6 )

{

x++;

}

if ( !b1 )

x = x + 10;

else if ( b2 = true ) /\* Line 19 \*/

x = x + 100;

else if ( b1 | b2 ) /\* Line 21 \*/

x = x + 1000;

}

}

System.out.println(x);

}

}

A. 0

B. 1

C. 101

D. 111

**8) What will be the output of the program?**

int i = 0;

while(1)

{

if(i == 4)

{

break;

}

++i;

}

System.out.println("i = " + i);

A. i = 0

B. i = 3

C. i = 4

D. Compilation fails.

**9) public class Test { }**

What is the prototype of the default constructor?

A.Test( )

B.Test(void)

C.public Test( )

D.public Test(void)

**10) import java.awt.Button;**

class CompareReference

{

public static void main(String [] args)

{

float f = 42.0f;

float [] f1 = new float[2];

float [] f2 = new float[2];

float [] f3 = f1;

long x = 42;

f1[0] = 42.0f;

}

}

11) which three statements are true?

f1 == f2

f1 == f3

f2 == f1[1]

x == f1[0]

f == f1[0]

A. 1, 2 and 3

B. 2, 4 and 5

C. 3, 4 and 5

D. 1, 4 and 5

**12) public class MyOuter**

{

public static class MyInner

{

public static void foo() { }

}

}

which statement, if placed in a class other than MyOuter or MyInner, instantiates an instance of the nested class?

A. MyOuter.MyInner m = new MyOuter.MyInner();

B. MyOuter.MyInner mi = new MyInner();

C. MyOuter m = new MyOuter();

MyOuter.MyInner mi = m.new MyOuter.MyInner();

D. MyInner mi = new MyOuter.MyInner();

**13)**

interface Base

{

boolean m1 ();

byte m2(short s);

}

which two code fragments will compile?

interface Base2 implements Base {}

abstract class Class2 extends Base

{ public boolean m1(){ return true; }}

abstract class Class2 implements Base {}

abstract class Class2 implements Base

{ public boolean m1(){ return (7 > 4); }}

abstract class Class2 implements Base

{ protected boolean m1(){ return (5 > 7) }}

A. 1 and 2

B. 2 and 3

C. 3 and 4

D. 1 and 5

**14)**

class A

{

protected int method1(int a, int b)

{

return 0;

}

}

Which is valid in a class that extends class A?

A. public int method1(int a, int b) {return 0; }

B. private int method1(int a, int b) { return 0; }

C. public short method1(int a, int b) { return 0; }

D. static protected int method1(int a, int b) { return 0; }