JAVA Programming Language Homework Ⅰ - OO concept

Student ID: Name:

1. Which of the following techniques can be used to prevent the instantiation of a class by any code outside of the class?

A. Declare all constructors with a void return type.

B. Declare all constructors using the private access modifier.

C. Do not declare any constructors inside a class definition.

D. Do not include a return statement in the constructor.

E. None of the above.

Answer:

2. Which of the following statements are true?

A. A constructor can invoke the constructor of the direct superclass using the superclass constructor invocation statement “super”.

B. By using constructor invocation statement “this”, a constructor can invoke another constructor of the same class.

C. The constructor invocation statement, “this”, can legally appear anywhere in the constructor body.

D. By using the constructor invocation statement “this”, a constructor can invoke itself.

E. None of the above.

Answer:

3. Given the following Java code:

```
1.    public class Hello {
2.            String title=””;
3.            int value;
4.            public Hello( ) {
5.                    title = title + “ World”;
6.                    System.out.print(title);
7.            }
8.            public Hello(int value) {
9.                    this.value = value;
10.                   title = “Hello”;
11.                   this( );
12.           }
```

```
    public static void main(String[] args){
        Hello b = new Hello (5);
    }
}
```

What is the result?

A. Hello

B. Hello World

C. Compilation fails

D. Hello World 5

E. Hello Hello

Answer:

4. Given the following Java code:

```
class Num {
    public static String b( ) { return “One”; }
    public static String b( int i ) { return “Two”; }
    public static String b( int i, int j ) throws Exception { return “Three”; }
    public static void main( String[] args ) {
        System.out.println( b(2) );
    }
}
```

What is the result?

A. One

B. Two

C. Compilation fails

D. Three

E. None of the above

Answer:

5. Given the following Java code:

Exhibit:

```
public class SimpleCalc {
        public int value;
        public void calculate( ) { value += 7; }
}
```

And:

```
Public class MultiCalc extends SimpleCalc {
        public void calculate( ) { value -= 3; }
        public void calculate( int multipier) {
                calculate( );
                super.calculate( );
                value *= multipier;
        }
        public static void main(String[] args) {
                MultiCalc calculator = new MultiCalc( );
                calculator.calculate(2);
                System.out.println(“ Value is: “ + calculator.value);
        }
}
```

What is the result?

A. Value is: 8

B. Compilation fails.

C. Value is: 12

D. Value is: -12

E. The code runs with no output.

Answer:

6. Given the following Java code:

```
public class Base {
        public static final String FOO = “foo”;
        public static void main(String[] args) {
                Base b = new Base( );
                Sub s = new Sub( );
                System.out.println(Base.FOO);
                System.out.println(Sub.FOO);
                System.out.println(b.FOO);
                System.out.println(s.FOO);
                System.out.println(((Base)s).FOO);
        }
}
class Sub extends Base {public static final String FOO=”bar”;}
```

What is the result?

A. foofoofoofoofoo

B. foobarfoobarbar

C. foobarfoofoofoo

D. foobarfoobarfoo

E. foofoofoobarbar

Answer:

7. Given the following Java code:

```
public class TestPoly {
        public static void main(String[] args) {
                Parent p = new Child( );
        }
}

class Parent {
        public Parent( ) {
                super();
                System.out.println(“instantiate a parent”);
        }
}

class Child extends Parent {
    public Child( ) {
        System.out.println(“instantiate a child”);
    }
}
```

What is the result?

A. instantiate a child

B. instantiate a parent

C. instantiate a child
instantiate a parent

D. instantiate a parent
instantiate a child

E. Compilation fails

Answer:

8. Given the following Java code:

```
public class TestPoly {
    public static void main(String[] args) {
        Parent p = new Child( );
    }
}

class Parent {
    public Parent( ) {
        super();
        System.out.println(“instantiate a parent”);
    }
}

class Child extends Parent {
    public Child( ) {
        System.out.println(“instantiate a child”);
        super( );
    }
}
```

What is the result?

A. instantiate a child

B. instantiate a parent

C. instantiate a child
instantiate a parent
D. instantiate a parent
instantiate a child
E. Compilation fails
Answer:

9. Given the following Java code:

```
class C {
        public static void main(String[] args) {
                A tmp = new B( );
                tmp.m1( );
                tmp.m2( );
                ((B)tmp).m1( );
                ((B)tmp).m2( );
        }
}
class A {public void m1( ) { System.out.println (“A”);}}
class B extends A {
        public void m1( ) { System.out.println (“B1”);}
        public void m2( ) { System.out.println (“B2”);}
        public void m3( ) { System.out.println (“B3”);}
        public void m4( ) { System.out.println (“B4”);}
}
```

What is the result?
A. AB2B1B2
B. B1B2B1B2
C. Compiler Error
D. Runtime Error
E. None of the above
Answer:

10. Given the following Java code:

```
public class Bootchy {
        int botch;
        String snootch;
```

```
    public Bootchy() {
        this(“snootchy”);
        System.out.print(“first ”);
    }
    public Bootchy(String snootch) {
        this(420, “snootchy”);
        System.out.print(“second ”);
    }
    public Bootchy(int bootch, String snootch) {
        this.bootch=botch;
        this.snootch = snootch;
        System.out.print(“third ”);
    }
    public static void main(String[] args){
        Bootchy b = new Bootchy();
        System.out.print(b.snootch +” “+ b.bootch);
    }
}
```

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 snootchy 420

(E) third first second snootchy 420

Answer:

11. Given the following Java code:

```
class A {
    private static int tmp = 1;
    static void m(int i) { tmp++; i++;}
    public void n(int i) { tmp = tmp + 2;}
    static void n() { tmp = tmp + 2;}
    public static void main(String[] args) {
        int tmp2 = 3;
        m(tmp2);
        System.out.println(tmp + “,” + tmp2);
    }
11. }
```

What is the result?

A. 1, 3

B. 2, 3

C. 1, 4

D. 2, 4

E. Compiler Error

Answer:

12. Which of the following are legal identifiers?

A. _3variable

B. 3_variable

C. this

D. super

E. *variable

Answer:

13. Which are not primitive types in Java?

A. float

B. Boolean

C. short

D. Double

E. long

Answer:

14. Given the following Java code:

```
1. interface Count {
2.          short counter = 0;
3.          void countUp( );
4. }
5. public class TestCount implements Count {
6.
7.          public static void main(String[] args) {
8.                   TestCount t = new TestCount( );
```

```
            t.countUp( );
    }
    public void countUp( ) {
            for (int x = 6; x > counter; x--, ++counter) {
            System.out.println(“ “ + counter);
            }
    }
}
```

What is the result?

A.0 1 2

B. 1 2 3

C. 0 1 2 3

D. 1 2 3 4

E. Compiler error

Answer:

15. Given the following Java code:

```
public class ConstOver {
    public ConstOver(int x, int y, int z) {
    }
}
```

Which two overload the ConstOver constructor?

A. ConstOver( ){}

B. Protected int ConstOver( ){}

C. Private ConstOver(int z, int y, int x){}

D. public Object ConstOver(int x, byte y, byte z){}

E. public void ConstOver(byte x, byte y, byte z){}

Answer:

16. Given the following Java code:

```
interface foo {
    int k = 0;
}
public class ExamA015 implements foo{
```

```
5.          public static void main(String[] args) {
6.                   int i;
7.                   ExamA015 test = new ExamA015( );
8.                   i = test.k;
9.                   i = ExamA015.k;
10.                  i = foo.k;
11.         }
12. }
```

What is the result?

A. Compilation succeeds.

B. An error at line 2 causes compilation to fail.

C. An error at line 9 causes compilation to fail.

D. An error at line 10 causes compilation to fail.

E. An error at line 11 causes compilation to fail.

Answer:

17. Given the following Java code:

```
1. public class foo {
2.          public static void main (String[] args) {
3.                   String s;
4.                   System.out.println(“s=” + s);
5.          }
6. }
```

What is the result?

A. The code compiles and “s=” is printed.

B. The code compiles and “s=null” is printed.

C. The code does not compile because string s is not initialized.

D. The code does not compile because string s cannot be referenced.

E. There is a runtime error.

Answer:

18. Which two statements 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.

Answer:

19. Which two statements are true? (choose two)
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.
Answer: