

JAVA 試験（入社用）

バージョン 1.0

2020 年 10 月 28 日

ニューエリート株式会社

1. 次のコードをコンパイル、実行したときの結果として、正しいものを選 びなさい。

(1つ選択)

1. public class Main {
2. public static void main(String[] args) {
3. for (int i = 0; i < 5; i++) {
4. for (i = 5; i < 10; i++) {
5. System.out.print(i);
6. }
7. }
8. }
9. }
10. 何も表示されない
11. 「56789」と1回表示される
12. 「56789」と5回表示される コンパイルエラーになる
13. コンパイルエラーになる
14. 実行時に例外が発生する

2. 次のSampleクラスを継承したサブクラスを定義するときに、サブクラ スに定義したメソッドのうち、Sampleクラスのメソッドを正しくオーパーライトしているものを選びなさい。 (2つ選択)

1. public class Sample {
2. void methodA() {}
3. void methods(int a) {}
4. int methodC(int a, int b) {
5. return 0;
6. }
7. int methodD(int a) {
8. return 0;
9. }
10. }
11. public void methodA() {}
12. public void methodB(long a) {}
13. public int methodC(char a, int b) {  
     return 0;  
    }
14. public int methodD (int i) {  
     return 1;  
    }

3. 次のコードをコンバイル、 実行したときの結果として、正しいものを選 びなさい。

(1つ選択)

1. public class Train {
2. static String name = "none";
3. public Train(String name) {
4. this.name = name;
5. }
6. public static void main(String[] args) {
7. Train t1 = new Train();
8. Train t2 = new Trainl("aline");
9. System.out.print(t1.name + " " + t2.name)
10. }
11. }
12. 「none aline」と表示される
13. 「null aline」と表示される
14. 「aline aline」と表示される
15. コンパイルエラーになる
16. 実行時に例外が発生する

4. ある企業は、GUIベースのアプリケーション開発を希望しており、将来的な拡張として、Webベースのアプリケーションへの移行を予定している。このアプリケーションを作成するには、 どのJavaテクノロジを使用するのがよいか。 正しいものを選びなさい。

(1つ選択)

1. Java SE
2. Java EE
3. Java ME
4. Java DB

5. 次のプログラムをコンパイル、実行したときの結果として、 正しいものを選びなさい。(1つ選択)

1. public interface Sample {
2. void test();
3. }
4. public class A implements Sample {
5. public void test () {
6. System.out.println("A");
7. }
8. }
9. public class B extends A {
10. public void test () {
11. System.out.println("B");
12. }
13. }
14. public class Main {
15. public static void main(String[] args) {
16. Sample[] samples = {new A(), new B()};
17. for (Sample s : samples) {
18. s.test();
19. }
20. }
21. }
22. 「A」「B」の順に表示される
23. 「B」「A」の順に表示される
24. 「A」「A」の順に表示される
25. 「B」「B」の順に表示される
26. Bクラスでコンパイルエラーが発生する
27. Mainクラスでコンパイルエラーが発生する

6. 以下の中から、アクセス修飾子privateで修飾できるものを選びなさい。(3つ選択)

1. インタフェースのフィールド
2. クラスのコンストラクタ
3. クラスのフィールド
4. クラスの抽象メソッド
5. インタフェースのメソッド
6. クラスの具象メソッド

7. クラスの宣言として有効なものを選びなさい。(3つ選択)

1. public class Test extends java.lang.\* {}
2. public class Test extends java.lang.Object {}
3. final class Test {}
4. public class Test {}
5. public class Test implements Object {}

8. 次のコードをコンバイル、実行したときの結果として、正しいものを選 びなさい。

(1つ選択)

1. public class Main {
2. public static void main(String[] args) {
3. for (int i = 0; ; i++) {
4. int j = 0;
5. while(j <= 3)
6. System.out.print(j++);
7. }
8. }
9. }
10. 「123」と1回表示される
11. 「0123」と1回表示される
12. 「1234」と1回表示される
13. 「1234」が無限に表示される
14. 「0123」が無限に表示される
15. コンパイルエラーとなる

9. 次のコードをコンバイル、 実行したときの結果として、正しいものを選 びなさい。

(1つ選択)

1. public class Main {
2. public static void main(String[] args) {
3. for (int i = 0; ++i < 5;)
4. System.out.print(i + " ");
5. }
6. }
7. 何も表示されない
8. 「0 1 2 3 4」と表示される
9. 「0 1 2 3 4 5」と表示される
10. 「1 2 3 4」と表示される
11. 「1 2 3 4 5」と表示される
12. コンパイルエラーになる

10. 次のインタフェースのコンパイルを成功させるには、どのコードを2行目の「// insert code here」に入れればよいか。正しいものを選びなさい。(2つ選択)

1. public interface Sample {
2. // insert code here
3. }
4. public void setVal(String val);
5. private void setVal(String val);
6. String val;
7. void setVal(String val);
8. public static void setVal(String val);

11. 次のコードをコンパイル、実行したときの結果として、正しいものを選 びなさい。

(1つ選択)

1. public class Bridge {
2. String name;
3. public static void main(String[] args) {
4. Bridge b = new Bridge();
5. if(b.name == "")
6. b.name = "Brooklyn";
7. System.out.println(b.name);
8. }
9. }
10. 何も表示されない
11. 「null」と表示される
12. 「Brooklyn」と表示される
13. コンパイルエラーにな
14. 実行時に例外が発生する

12. 次のコードをコンバイル、実行したときの結果として、正しいものを選びなさい。

(1つ選択)

1. public class Test {
2. private static int a;
3. private int b;
4. public static int countUpA() {
5. return ++a;
6. }
7. public int doMethod() {
8. return countUpA();
9. }
10. public static void main(String[] args) {
11. Test test = new Test();
12. System.out.print(test.doMethod());
13. System.out.print(" " + test.countUpA());
14. }
15. }
16. 「0 1」と表示される
17. 「1 2」と表示される
18. 9行目でコンパイルエラーになる
19. 15行目でコンパイルエラーになる
20. 実行時に例外が発生する

13. 以下の中から、privateメソッドにアクセスできるものを選びなさい。(3つ選択)

1. サブクラスのpublicメソッド
2. 同じクラスのprivateメソッド
3. super()を使用するサブクラスのコンストラクタ
4. オーバーロードされたメソッド
5. 同じクラスのメソッド内に定義した自インスタンスを参照するthis変数

14. 次の配列の要素をすべて出力するコードとして正しいものを選びなさい。(1つ選択)

int[] array = {1, 2, 3, 4, 5};

1. for(int i = 1; i < array.length; 1++) {  
    System.out.println(array[i]);  
   }
2. while(int i = 0; i < array.length) {  
    System.out.println(array[i]);   
    i++;  
   }
3. for (int i = 0; i < array.length; i++) {  
    System.out.println(array[i]);  
   }
4. while(int i = 1; i < array.length) {  
    System.out.println(array[i]);  
    ++i;  
   }

15. 以下の中から、ポリモーフィズムに関係が深いものを選びなさい。(2つ選択)

1. インタフェースの継承
2. インタフェースの実装
3. メソッドのオーバーロード
4. メソッドのオーバーライド
5. アクセス修飾子protectedの利用

16. 次のコードをコンパイル、実行したときの結果として、正しいものを選びなさい。

(1つ選択)

1. public class Counter {
2. static int count = 0;
3. Counter () {
4. ++count;
5. }
6. public static void main(String[] args) {
7. Counter c = new Counter ();
8. c = new Counter();
9. System.out.println(c.count);
10. }
11. }
12. 0が表示される
13. 1が表示される
14. 2が表示される
15. コンパイルエラーになる
16. 実行時に例外が発生する

17. 次のようなItemクラスとBookクラスが定義されている場合、Itemクラスのインスタンスが生成されるコードとして正しいものを選びなさい。 (2つ選択)

1. class Item {}
2. class Book extends Item {}
3. Item item = null;
4. Item item;
5. Item item = new Book();
6. Item item = new Item();
7. new Item();

18. 次のプログラムをコンパイル、実行したときの結果として正しいものを選びなさい。

(1つ選択)

1. public class SuperClass {
2. private int num = 10;
3. void test() {
4. System.out.println(num);
5. }
6. }
7. public class SubClass extends SuperClass {
8. private int num = 20;
9. public static void main(String[] args) {
10. SubClass sub = new SubClass();
11. sub.test();
12. }
13. }
14. コンパイルエラーになる
15. 実行時に例外がスローされる
16. 10が表示される
17. 20が表示される

19. 次のコードをコンパイル、 実行したときの結果として、 正しいものを選びなさい。

(1つ選択)

1. public class Main {
2. public static void main (String[] args) {
3. int x = 3;
4. int y = 5;
5. if(x != 3)
6. System.out.println("A");
7. else if(y > x)
8. System.out.println("B");
9. else
10. System.out.println("C");
11. }
12. }
13. 「A」と表示される
14. 「B」と表示される
15. 「C」と表示される
16. 「B C」と表示される
17. コンパイルエラーになる

20. 次のコードをコンパイル、実行したときの結果として、正しいものを選びなさい。

(1つ選択)

1. public class Main {
2. public static void main(String[] args) {
3. int x = 12 / 2;
4. int y = 2 \* 3;
5. if(x > y)
6. System.out.println("A");
7. else if(x < y)
8. System.out.println("B");
9. else if(x = y)
10. System.out.println("C");
11. else
12. System.out.println("D");
13. }
14. }
15. 「A」と表示される。
16. 「B」と表示される
17. 「C」と表示される
18. 「D」と表示される
19. コンパイルエラーになる
20. 実行時に例外が発生する

21. 次のコードをコンパイル、実行したときの結果として、正しいものを選びなさい。

(1つ選択)

1. public class Main (
2. public static void main(String[] args) {
3. byte a = -120;
4. short b = 90000;
5. int c = -20000000:
6. long d = 920000000L;
7. }
8. }
9. コンパイルは成功する。
10. 複数行でコンパイルエラーになる
11. 3行目でコンパイルエラーになる
12. 4行目でコンパイルエラーになる
13. 5行目でコンパイルエラーになる
14. 6行目でコンパイルエラーになる

22. 次のコードをコンパイル、実行したときの結果として、正しいものを選びなさい。

(1つ選択)

1. public class Calculator {
2. int beforeTaxes(int price) {
3. return (int)(price \* 1.08);
4. }
5. double beforeTaxes (int price) {
6. return price \* 1.08;
7. }
8. public static void main(String[] args) {
9. int p = 40;
10. Calculator calc = new Calculator();
11. System.out.println("payment = " + calc.beforeTaxes(p));
12. }
13. }
14. 「payment = 40」と表示される
15. 「payment = 43」 と表示される
16. 「payment = 43.2」 と表示される
17. コンパイルエラーになる

23. 次のプログラムをコンバイル、実行したときの結果として、正しいものを選びなさい。(1つ選択)

1. public class SuperClass {
2. public void test() {
3. System.out.println("super");
4. }
5. }
6. public class SubClass extends SuperClass {
7. public void test() {
8. System.out.println("sub");
9. }
10. }
11. public class Sample {
12. public static void main(String[] args) {
13. SuperClass s = new SubClass();
14. s.test();
15. }
16. }
17. 「super」と表示される
18. 「sub」と表示される。
19. コンパイルエラーになる
20. 実行時に例外がスローされる

24. 抽象クラスに関する説明として、正しいものを選びなさい。(1つ選択)

1. 抽象クラスに定義されるメソッドは、暗黙的にpublic abstractである
2. 抽象クラスに定義されるフィールドは、暗黙的にstatic finalである
3. 抽象クラスはサブクラス化できない
4. 抽象クラスはインスタンス化できない
5. 抽象クラスは抽象メソッドを持たなければいけない

25. 次のプログラムをコンパイル、実行したときの結果として、正しいものを選びなさい。(1つ選択)

1. public class SuperClass {
2. public void print() {
3. System.out. printin("super");
4. }
5. }
6. public class SubClass extends SuperClass {
7. public void print() {
8. System.out.println("sub");
9. }
10. public static void main(String[] args) {
11. SuperClass s = new SuperClass();
12. SubClass s2 = (SubClass) s;
13. s2. print();
14. }
15. }
16. 「super」と表示される
17. 「sub」と表示される
18. コンパイルエラーになる
19. 実行時に例外がスローされる

26. 次のプログラムを実行し、「sub,super」と表示されるようにしたい。 「insert code here」にあてはまるコードを選びなさい。 (1つ選択)

1. public class SuperClass {
2. String val = "super";
3. }
4. public class SubClass extends SuperClass {
5. String val = "sub";
6. public void test() {
7. System.out.println(val + "," + ｢insert code here｣);
8. }
9. public static void main(String[] args) {
10. SubClass sub = new SubClass();
11. sub. test();
12. }
13. }
14. super (val)
15. this.val
16. super.val
17. this(val)
18. super().val
19. SuperClass.val

27. 配列の宣言と配列インスタンスの生成として有効なものを選びなさい。(3つ選択)

1. int array = new int[3];
2. int[] array = new int(3);
3. int[] array;  
   array = new int[3];
4. int array[3];
5. int[] array = {3, 6, 2};
6. int[] array = new int()[3];
7. int[] array = new int[3];

28. 次のコードが正常にコンパイルされるようにしたい。 Sampleimplクラ スの2行目「insert code here」に入るコードとして、正しいものを びなさい。(1つ選択)

1. abstract class Sample {
2. void print () {
3. System. out.print In ("sample");
4. }
5. public abstract void test ();
6. }
7. public class SampleImpl extends Sample {
8. //insert code here
9. }
10. void test() {   
     System.out.println("impl");  
    }
11. public abstract void test() {  
     System.out.println("impl");  
    }
12. public void test() {  
     System.out.println("impl");  
    }
13. public void print() {  
     System.out.println("imp1");  
    }
14. void print() {  
     System.out.println("impl");  
    }

29、次のコードをコンパイル、実行したときの結果として、正しいものを選 びなさい。

(1つ選択)

1. public class Point {
2. public static void main(String args) {
3. int x = 5;
4. Point p = new Point ();
5. p.move (x);
6. }
7. public void move (int y) {
8. System.out.println(x++ + "" + --y);
9. }
10. }
11. 「0 5」と表示される
12. 「1 5」と表示される
13. 「0 4」と表示される
14. コンパイルエラーになる
15. 実行時に例外が発生する

30. コンストラクタに関する説明として、正しいものを選びなさい。(2つ選択)

1. サブクラスはスーパークラスのコンストラクタを引き継がない
2. コンストラクタの戻り値型はvoidにすることができる
3. コンストラクタはprivateで修飾できる
4. コンストラクタは、 クラス内のすべてのフィールドを初期化しなければならない