**1. System.out.println(‘5’ + 2);的输出结果应该是（ D ）。**

A、52 B、7 C、2 D、55

**2.下面程序的运行结果是什么？（ D ）**

|  |
| --- |
| **public class** Demo {  **static** String *x* = **"1"**;  **static int** *y* = 1;  **public static void** main (String[] args) {  **static int** z= 2;  System.***out***.println(*x*+*y*+z);  } } |

A. 3   
B. 112   
C. 13   
D. 程序有编译错误

**3.下面方法编译错误的是（ AD ）**

|  |
| --- |
| **public class** Demo{  **float** func1(){  **int** i=1;  **return**;  }  **float** func2(){  **short** i=2;  **return** i;  }  **float** func3(){  **long** i=3;  **return** i;  }  **float** func4(){  **double** i=4;  **return** i;  } } |

A. func1   
B. func2   
C. func3   
D. func4

**4.在不考虑极值的情况下，下面哪一个循环会导致死循环？（ C ）**

A、for (int k = 0; k < 0; k++)

B、for (int k = 10; k > 0; k--)

C、for (int k = 0; k < 10; k--)

D、for (int k = 0; k > 0; k++)

**5下面字段声明中哪一个在interface主体内是合法的? （ B ）**

1. private final static int answer = 42;
2. public static int answer = 42;
3. final static answer = 42;
4. int answer;

单个技术点掌握的可以

综合起来就不会

**急于求成**

**5-6：从零基础学到项目----》就业**

**公司里面会有各种各样你没见过的**技术

**6.指出正确的表达式（ BCD ）**

A、 byte b=128;

B、 Boolean b =null;

C、 long l=0xfffL;

D、 double d =0.9239d;

**7.下面的方法，当输入i为2的时候返回值是多少?（ D ）**

|  |
| --- |
| public static int getValue(int i) {  int result = 0;  switch(i){  case 1:  result = result + i;  case 2:  result = result + i \* 2;  case 3:  result = result + i \* 3;  }  return result;  } |

A）0 B）2 C）4 D）10

**8.下面程序的运行结果是什么？**

|  |
| --- |
| **public class** Inc {  **public static void** main(String args[]){  Inc inc = **new** Inc();  **int** i = 0;  inc.fermin(i);  i = i++;  System.***out***.println(i);  }  **void** fermin(**int** i){  i++;  } } |

A. 0   
B. 1   
C. 2   
D. 3

**9.下面的代码段中，执行之后i 和j 的值是什么? (    )**

int i = 1; int j;

j = i++\*2 + 3\*--i;

 A．1, 2          B．1, 5      C． 2, 1         D． 2, 2

**10.给定如下java代码，编译运行之后，将会输出。**

|  |
| --- |
| class Test{  public static void main(String args[]){  int a=5;  System.out.println(a%2 == 1? a/2 :(a+1) /2);  }  } |

结果输出：（ B ）

 A）1        B）2        C）2.5        D）3

**11. 给定如下java代码，编译运行时，结果是（ ）。**

|  |
| --- |
| class Test{  public static void main (String args[]) {  for (int i=0;i<3; i++) {  System.out.print(i) ;  }  System.out.print(i);  }  } |

A) 编译时报错

B) 正确运行，输出012

C) 正确运行，输出123

D) 正确运行，输出0123

**12. 关于下列代码说法正确的是：（ ）**

|  |
| --- |
| **public class** Foo {  **public static int** add(**int** a, **int** b) {  **return** a + b;  }  **public static void** main(String[] args) {  Foo foo = **null**;  System.***out***.println(foo.add(10, 20));  } } |

A.编译错误

B.正常运行，但无结果输出

C.运行输出：30

D.运行时抛出NullPointerException异常

**13 Java表达式"4 & 5"的结果是什么?（ ）**

A. 0

B. 5

C. 4

D. 1

**14 下面程序运行的结果正确的是（ ）**

|  |
| --- |
| **public** **class** Demo1 {  **public** **static** **void** main(String[] args) {  **int** n = 100;  **int** m = 20;  **if** (m++ <= 20 && n == n++)  System.***out***.println(n);  System.***out***.println(m);  }  } |

A 什么也不输出

B 20

C 101 20

D 101 21

**15.下面说法正确的是（ ）**

|  |
| --- |
| **public class** MyClass {  **long var**;  **public void** MyClass(**long** param) {  **var** = param;  }*//(1)* **public static void** main(String[] args) {  MyClass a, b;  a = **new** MyClass();*//(2)* b = **new** MyClass(5);*//(3)* } } |

A．编译错误将发生在（1），因为构造函数不能指定返回值

B．编译错误将发生在（2），因为该类没有默认构造函数

C．编译错误将在（3）处发生，因为该类没有构造函数，该构造函数接受一个int类型的参数

D．该程序将正确编译和执行

**17、 不能用在类(非内部类)上的修饰符是（  ）**

A) final

B) public

C) abstract

D) protected

E) private

**18、 当编译并运行下面程序时会出现什么结果（  ）**

public class MyAr{

     public static void main(String argv[]){

 int[] i = new int[5];

          System.out.println(i[5]);

}

}

A) 编译出错

B) 运行出错

C) 输出0

D) 输出“null”

**19、 已定义两个接口B和C，要定义一个类同时实现这两个接口，正确的是（   ）**

A) interface A extends B,C

B) interface A implements B,C

C) class A implements B,C

D) class A implements B,implements C

**20、 下面关于继承的叙述正确的是（ ）**

A. 在JAVA里只允许单一继承

B. 在JAVA里一个类只能实现一个接口

C. JAVA可以多继承

D. JAVA的类单一继承避免了菱形继承的危险

**21、 给定JAVA代码如下，关于super的用法，以下描述正确的是（   ）**

class C extends B {

    public C()     {

         super();

}

}

A、用来调用类B中定义的super()方法

B、用来调用类C中定义的super()方法

C、用来调用类B中的无参构造方法

D、用来调用类B中第一个出现的构造方法

**22、以下描述不正确的是？（ ）**

A. final class不能被继承

B. final变量不能被修改

C. final属性不可以在构造方法中赋值

D. final方法不能被覆盖(override)

E. final 类属性可以在构造器中赋值

**23、下面关于java语言中方法的说法错误的是： （ ）**

A.所有的具体类中方法必须有方法体

B.参数传递是值传递的方式

C.如果方法没有返回值必须声明返回为void

D.如果方法定义为返回void，则方法中不能出现return语句

**24. 给定Java代码如下所示，在横线处新增下列（ ）方法，是对cal方法的重载。**

public class Test{

  public void cal(int x, int y, int z) {}

\_\_\_\_\_\_\_\_\_\_\_\_\_

}

A. public int cal(int x, int y, float z){ return 0; }

B. public int cal(int x, int y, int z){ return 0; }

C. public void cal(int x, int z){ }

D. public void cal(int z, int y, int x){ }

**25. 在Java中，下面对于构造函数的描述正确的是（   ）。**

A. 一个类可以定义多个构造函数

B. 构造函数的返回类型是void

C. 构造函数和类有相同的名称，并且不能带任何参数

D. 抽象类中不能定义构造函数

**26. 下面Java代码的运行结果是（   ）**

|  |
| --- |
| **class** Penguin {  **private** String name = **null**; // 名字  **private** **int** health = 0; // 健康值  **private** String sex = **null**; // 性别  **public** **void** Penguin() {  health = 10;  sex = "雄";  System.*out*.println("执行构造方法。");  }  **public** **void** print() {  System.*out*.println("企鹅的名字是" + name + "，健康值是" + health + "，性别是" + sex  + "。");  }  **public** **static** **void** main(String[] args) {  Penguin pgn = **new** Penguin();  pgn.print();  }  } |

A. 企鹅的名字是null，健康值是10，性别是雄。

B. 执行构造方法。  企鹅的名字是null，健康值是0，性别是null。

C. 企鹅的名字是null，健康值是0，性别是null。

D. 执行构造方法。 企鹅的名字是null，健康值是10，性别是雄。

**27.给定如下一个Java源文件Child.java，编译并运行Child.java，以下结果正确的是（  ）。**

|  |
| --- |
| **class** Parent1 {  Parent1(String s){  System.***out***.println(s);  } } **class** Parent2 **extends** Parent1{  Parent2(){  System.***out***.println(**"parent2"**);  } } **public class** Child **extends** Parent2 {  **public static void** main(String[] args) {  Child child = **new** Child();  } } |

A. 编译错误：没有找到构造器Child()

B. 编译错误：没有找到构造器Parent1()

C. 正确运行，没有输出值

D. 正确运行，输出结果为：parent2

**28. 分析如下所示的Java代码，则选项中的说法正确的是（  ）。**

|  |
| --- |
| **class** Parent{  **public** String **name**;  **public** Parent(String pName){  **this**.**name** = pName;  } } **public class** Test **extends** Parent { *//1* **public** Test(String Name){ *//2* **name**=**"hello"**; *//3* **super**(**"kitty"**); *//4* } } |

A. 第2行错误，Test类的构造函数中参数名称应与其父类构造函数中的参数名相同

B. 第3行错误，应使用super关键字调用父类的name属性，改为super.name="hello";

C. 第4行错误，调用父类构造方法的语句必须放在子类构造方法中的第一行

D. 程序编译通过，无错误

**29. 在Java接口中定义常量，下面语法错误的是（ ）。**

A. static int MALE = 1;

B. final int MALE = 1;

C. int MALE = 1;

D. private int MALE = 1;

31、请看下列代码

|  |
| --- |
| **class** ClassA {} **class** ClassB **extends** ClassA {} **class** ClassC **extends** ClassA {} **public class** Test{  **public static void** main(String[] args) {  ClassA p0 = **new** ClassA();  ClassB p1 = **new** ClassB();  ClassC p2 = **new** ClassC();  ClassA p3 = **new** ClassB();  ClassA p4 = **new** ClassC();  <插入代码>  } } |

下列选项中放置在<插入代码>处，使程序编译正确的是：  （ ）

A.p0 = p1;

B.p1 =p2;

C.p2 = p4;

D.p2 = (ClassC)p1;

**32、程序输出的结果是 （ ）**

|  |
| --- |
| **class** One {  **public** One() {  System.***out***.print(1);  } } **class** Two **extends** One {  **public** Two() {  System.***out***.print(2);  } } **class** Three **extends** Two {  **public** Three() {  System.***out***.print(3);  } } **public class** Numbers {  **public static void** main(String[] argv) {  **new** Three();  } } |

A. 1

B. 3

C. 123

D. 321

**33、下面代码运行结果是： （ ）**

|  |
| --- |
| **public class** Animal {  **public** String noise() {  **return "peep"**;  }  **public static void** main(String[] args) {  Animal animal = **new** Dog();  Cat cat = (Cat)animal;  System.***out***.println(cat.noise());  } } **class** Dog **extends** Animal {  **public** String noise() {  **return "bark"**;  } } **class** Cat **extends** Animal {  **public** String noise() {  **return "meow"**;  } } |

A.peep

B.bark

C.meow

D.抛出运行时异常

**34、下面代码运行结果是： （ ）**

|  |
| --- |
| **class** SimpleCalc {  **public int value**;  **public void** calculate() {  **value** += 7;  } } **public class** MultiCalc **extends** SimpleCalc {  **public void** calculate() {  **value** -= 3;  }  **public void** calculate(**int** multiplier) {  calculate();  **super**.calculate();  **value** \*= multiplier;  }  **public static void** main(String[] args) {  MultiCalc calculator = **new** MultiCalc();  calculator.calculate(2);  System.***out***.println(**"Value is: "** + calculator.**value**);  } } |

 A.Value is: 8

 B.Value is: -8

C.Value is: 12

D.Value is: -12

35、程序执行的结果是：（ ）

|  |
| --- |
| **package** com.rl;  **class** Foo {  **public** **int** a;  **public** Foo() {  a = 3;  }  **public** **void** addFive() {  a += 5;  }  }  **class** Bar **extends** Foo {  **public** **int** a;  **public** Bar() {  a = 8;  }  **public** **void** addFive() {  **this**.a += 5;  }  }  **public** **class** TestFoo {  **public** **static** **void** main(String[] args) {  Foo foo = **new** Bar();  foo.addFive();  System.*out*.println("Value: " + foo.a);  }  } |

A.Value: 3

B.Value: 8

 C.Value: 13

D.Value: 18