1．Java语言的类间的继承关系是( B )。

A) 多重的 B) 单重的 C) 线程的 D) 不能继承

2．下列选项中，用于在定义子类时声明父类名的关键字是( C )。

A）interface B) package C) extends D) class

3．定义类头时能使用的修饰符是( C )。

A) private B) static C) abstract D) protected

4. 下列类定义中，不正确的是( C )。

A) class x { .... }

B) class x extends y { .... }

C) static class x implements y1,y2 { .... }

D) public class x extends Applet { .... }

5. 设 x、 y为已定义的类名，下列声明x类的对象x1的语句中正确的是(C )。

A) static x x1; B) public x x1=new x(int 123);

C) x x1; D) x x1=x( );

6. 设 i、j、k 为类 x 中定义的 int 型变量名，下列类 x 的构造方法中不正确的是( B )。

A) x( int m){ ... } B) void x( int m){ ... }

C) x( int m, int n){ ... } D) x( int h,int m,int n){ ... }

7. 下列方法定义中，正确的是( A )。

A) int x( ){ char ch='a'; return (int)ch; }

B) void x( ){ ...return true; }

C) int x( ){ ...return true; }

D) int x( int a, b){ return a+b; }

8. 用于定义类成员的访问控制权的一组关键字是( D )。

A) class, float, double, public

B) float, boolean, int, long

C) char, extends, float, double

D) public, private, protected

9. 由于Java不支持多继承，而有可能某个类或对象要使用分别在几个类或对象里面的方法或属性，现有的单继承机制就不能满足要求需要用到 接口 。

10、以下叙述正确的是：（ B ）  
 A．构造方法必须是public方法  
 B．main方法必须是public方法  
 C．Java应用程序的文件名可以是任意的  
 D．构造方法应该声明为void类型

11、关于类和对象的叙述正确的是：（ A ）  
 A．Java的类分为两大部分：系统定义的类和用户自定义的类  
 B．类的静态属性和全局变量的概念完全一样，只是表达形式不同  
 C．类的成员至少有一个属性和一个方法  
 D．类是对象的实例化  
12、以下有关构造方法的说法，正确的是：（ A ）  
 A．一个类的构造方法可以有多个  
 B．构造方法在类定义时被调用  
 C．构造方法只能由对象中的其它方法调用  
 D．构造方法可以和类同名，也可以和类名不同  
13、以下有关类的继承的叙述中，正确的是：（ D ）  
 A．子类能直接继承父类所有的非私有属性，也可通过接口继承父类的私有属性  
 B．子类只能继承父类的方法，不能继承父类的属性  
 C．子类只能继承父类的非私有属性，不能继承父类的方法  
 D．子类不能继承父类的私有属性

14、void 的含义：（ C ）  
 A．方法体为空  
 B．定义的方法没有形参  
 C．定义的方法没有返回值  
 D．方法的返回值不能参加算术运算

15. 在// point x处的哪些声明是合法的（ BCDE ）

class Person {

private int a;

public int change(int m){ return m; }

}

public class Teacher extends Person {

public int b;

public static void main(String arg[]){

Person p = new Person();

Teacher t = new Teacher();

int i;

// point x

}

}

A. i = m;

B. i = b;

C. i = p.a;

D. i = p.change(30);

E. i = t.b;

16. main()方法的返回类型是什么？ [ B ]

A. int

B. void

C. boolean

D. static

17. 给出下面的不完整的类代码

class Person {

String name, department;

int age;

public Person(String n){ name = n; }

public Person(String n, int a){ name = n; age = a; }

public Person(String n, String d, int a) {

// 完成Person(String n, int a)的逻辑

department = d;

}

}

下面的哪些表达式可以加到构造方法中的注释处? [ C ]

A. Person(n,a);

B. this(Person(n,a)); //构造函数的名字不能当方法名调用;

C. this(n,a);

D. this(name,age); //构造方法的参数也不能直接传实例变量名

18. 类的设计要求它的某个成员变量不能被外部类直接访问。应该使用下面的哪些修饰符获得需要的访问控制? [ D ]

A. public

B. 不加修饰符

C. protected

D. private

19. 下面的哪些java源文件代码片断是对的? [ ACD ]

A. package testpackage;

public class Test{//do something...}

B. import java.io.\*;

package testpackage;

public class Test{// do something...}

C. import java.io.\*;

class Person{// do something...}

public class Test{// do something...}

D. import java.io.\*;

import java.awt.\*;

public class Test{// do something..}

20. 给出下面的代码输出。

class Parent {

String one, two;

public Parent(String a, String b){

one = a;

two = b;

}

public void print(){ System.out.println(one); }

}

public class Child extends Parent {

public Child(String a, String b){

super(a,b);

}

public void print(){

System.out.println(one + " to " + two);

}

public static void main(String args[]){

Parent p = new Parent("south", "north");

Parent t = new Child("east", "west");

p.print();

t.print();

}

}

south

east to west

21. 类Teacher和Student都是类Person的子类

Person p;

Teacher t;

Student s;

若p,t和s都是非空值

if(t instanceof Person) { s = (Student)t; }

这个语句导致的结果是什么? [C] // 默认new时声明的都是本类;

A. 将构造一个Student对象

B. 表达式合法

C. 编译时非法

D. 编译时合法而在运行时可能非法

22.编译和运行下面的应用程序，并在命令行界面输入12345，则回车后屏幕输出的结果是( C ).

public class A {

public static void main(String args[]) throws IOException{

BufferedReader buf=new BufferedReader(

new InputStreamReader(System.in));

String str=buf.readLine();

int x=Integer.parseInt(str);

System.out.println(x/100);

}

}

A) 45 B)5 C) 123 D) 12345

23.下面main()方法的定义哪些是正确的( ACD )？

A)public static void main(String args[]) {} B)public static void main(String[]) {}

C)public static void main(String[] args) {} D)public static void main(String [] x) {}

24.下面关于类的说法正确的是( ABCD )。

A)类是 Java 语言中的一种复合数据类型。

B)类中包含数据变量和方法。

C)类是对所有具有一定共性的对象的抽象。

D) Java 语言的类只支持单继承。

25. 编译并运行下面的程序，运行结果为( BD ).

public class T1 {

public static void main (String[] args){

T1 a=new T1();

a.method(8);

a.method(1.2f);

}

void method(float i) {

System.out.println("float: "+i);

}

void method(long i) {

System.out.println("long: "+i);

}

}

A)程序有编译错误，因为两个method()方法必须定义为静态（static）的。

B)程序可以编译运行，输出结果为：

long: 8

float: 1.2

C)程序可以编译运行，输出结果为：

float: 1.2

long: 8

D) 程序中定义的两个参数不同的method方法称为方法重栽。

26.下面哪个说法正确(ACD )？

A)如果某公共的接口被声明在一个源文件中，则接口必须与源文件具有相同的名字。

B)如果源文件包含import语句，则该语句必须是除空行和注释行外的第一个语句行。

C)如果某源文件包含package语句，则该语句必须是第一个非空、非注释行。

D)接口中的方法必定是抽象方法，但可以不用关键字abstract修饰。

27．接口中的成员只有静态常量和\_静态方法\_\_\_\_\_。

28. 在子类中使用关键字\_\_super\_\_\_\_\_做前缀可调用被子类覆盖的父类中的方法。

29. 关键字\_\_\_import\_\_\_用于导入包中的类到程序中，供程序中使用。

30、在编写Java Application程序时，若需要使用到标准输入输出语句，必须在程序的开头写上( C )语句。

A. import java.awt.\* ;

B. import java.applet.Applet ;

C. import java.io.\* ;

D. import java.awt.Graphics ;

31、有关Java语言的特点，哪种是错误的（ D ）

A、多态 B、可移植 C、多线程 D、手动垃圾回收

32、以下不属于构造方法特征的是： （ B ）

A、构造方法名与类名相同 B、构造方法能够继承

C、构造方法在创建对象时自动执行。 D、每一个类可以有多个构造方法

33、设类B是类C的父类，下面声明对象x1的语句不正确的是（ D ）。

A、 B x1 = new B() B、B x1 = new C()

C、C x1 = new C() D、C x1 = new B()

34、下列（ B ）修饰符所定义的方法必须被子类所覆盖

A、final B、abstract C、 static D、volatile

35、this关键字的含义是 ( B )。

A. 本类; B. 本类对象 ;

C. 这个类; D.父类对象;

36、public class Test{

double x,y;

Test (){x=1.0;y=2.0;}

Test (double x,double y){this.x=x;this.y=y;}

public static void main(String args[]){

Test p1=new Test ();

Test p2=new Test (10.0,20.0);

System.out.println(p1.x+p2.x);

}

}

11

37、class Employee

{

String name; //姓名

char sex; //性别

Employee(String n, char s)

{

name = n;

sex = s;

}

public String getName()

{ return name; }

public char getSex()

{ return sex;}

}

class Worker extends Employee

{

char category;

boolean dressAllowance;

Worker(String n,char s,char c,boolean d)

{

super(n,s);

category = c;

dressAllowance = d;

}

public char getCategory()

{return category;}

public boolean getAll()

{ return dressAllowance; }

}

class InheDemo

{

public static void main(String args[])

{

Worker w = new Worker("M.John",'W','B',true);

System.out.println("工人信息");

System.out.println("姓名 ："+w.getName());

If (w.getSex()==’M’)

System.out.println("性别 ：男 ");

Else

System.out.println("性别 ：女 ");

System.out.println("类别 ："+w.getCategory());

if(w.getAll())

System.out.println("提供服装津贴");

else

System.out.println("未提供服装津贴");

}

}

工人信息

姓名 ：M.John

性别 ：女

类别 ：B

提供服装津贴

38．读程序，写出程序执行结果。

public class TestStatic{

static int num=10;

public static void test( ){ TestStatic. num++; }

public static void main(String[ ] args){

System.out.println(TestStatic.num);

TestStatic x=new TestStatic ();

x. num++;

System.out.println(x.num);

TestStatic y=new TestStatic ();

y.test();

System.out.println(x.num);

} }

程序执行结果是：10

11

12

39．一个Point类定义如下：

public class Point {

public int x = 0;

public int y = 0;

public Point(int x, int y) {

this.x = x;

this.y = y;

} }

若要创建一个Point类的对象origin\_one，要使用的语句是（ C ）

A、Point origin\_one = Point(23, 94);

B、Point origin\_one(23, 94);

C、Point origin\_one = new Point(23, 94);

D、 new Point(23, 94);

40、按以下要求编写程序

(1) 创建一个Rectangle类，添加width和height两个成员变量

(2) 在Rectangle中添加两种方法分别计算矩形的周长和面积

(3) 编程利用Rectangle输出一个矩形的周长和面积

解答：

public class Rectangle {

float width, height;

public Rectangle(float width, float height) {

this.width = width;

this.height = height;

}

public float getLength(){

return (this.width + this.height) \* 2;

}

public float getArea(){

return this.width \* this.height;

}

public static void main(String [] args) {

Rectangle rect = new Rectangle(10, 20);

System.*out*.println("周长是：" + rect.getLength());

System.*out*.println("面积是：" + rect.getArea());

}

}

41、按以下要求编写程序

(1) 编写Animal接口，接口中声明run() 方法

(2) 定义Bird类和Fish类实现Animal接口

(3) 编写Bird类和Fish类的测试程序，并调用其中的run()方法

解答：

public interface Animal {

void run();

}

class Bird implements Animal {

public void run() {

System.*out*.println("鸟儿在飞...");

}

}

class Fish implements Animal {

public void run() {

System.*out*.println("鱼儿在游...");

}

}

public class TestAnimal {

public static void main(String[] args) {

Bird bird = new Bird();

bird.run();

Fish fish = new Fish();

fish.run();

}

}