

100 JAVA Questions

Please choose the correct answer

1. Which following Java statement will generate the output of “Hello World!”?

- A. `System.output.println("Hello World!");`
- B. `cout<<“Hello World!”<<endl;`
- C. `echo (“Hello World!”);`
- D. `System.out.println(“Hello World!”);`

2. For public class Hello { }, which following statement is correct?

- A. The Java’s file name should be Hello.javac.
- B. The Java’s file name should be Hello.javadoc.
- C. The Java’s file name should be Hello.java.
- D. The Java’s file name should be Hello.class.

3. Which following statement is not correct?

- A. The value range of byte is -128~127
- B. The value range of short is -32768~32767
- C. The value range of int is -3276800~3276700
- D. The value range of char is 0~65535

4. What will be the output of the following code if it is started from the command line using “java test app boy cat dog”?

```
public class test {  
    public static void main (String[ ] args){  
        System.out.println(args[3]);  
    }  
}
```

What is the output?

- A. app
- B. boy
- C. cat
- D. dog

5. About Java comments, which following is correct?

- A. // is used in multiple line comments
- B. /* */ are used in single line comments
- C. Both // and /* */ can be used in single line or multiple comments.
- D. // is used in single line comments and /* */ are used in multiple comments.

6. \n will force a new line break in the output, \t will make a tab spacing in the output, \" will _____?

- A. allow quotation marks to be used inside strings.

- B. allow backspace to be used inside strings.
- C. allow return to be used inside strings.
- D. allow back slash to be used inside strings.

7. To create a variable, and assign a value, use ____?

- A. `variableName=value;`
- B. `var variableName=value;`
- C. `dataType variableName=value;`
- D. `modifier variableName=value;`

8. Which following is **not** a valid variable name?

- A. `myVar`
- B. `10Var`
- C. `_myVar`
- D. `$myVar`

9. Which following can be used as a variable name?

- A. `char`
- B. `float`
- C. `try`
- D. `nonVir`

10. Which following keyword description is **not** correct?

- A. "char" is used for a single Unicode character.
- B. "float" is used for a floating-point number with a decimal point.
- C. "integer" is used for an integer number from -2.14 billion to 2.14 billion.
- D. "boolean" is used for a boolean value of true or false.

11. Which following operator description is **not** correct?

- A. + is used in addition
- B. - is used in subtraction
- C. * is used in multiplication
- D. % is used in division

12. Which following express is **not** correct?

- A. $a += b$ means $a = (a + b)$
- B. $a *= b$ means $a = (a * b)$
- C. $a != b$ means $a = (a ! b)$
- D. $a \% = b$ means $a = (a \% b)$

13. Which following line is **not** correct?

```
public static void main (String args[ ]){  
String str="good";  
if ((str!="good")&&(str.length() <100)) // line1  
{System.out.print("compile succeed");} // line2
```

```
else if ((str!="good")&(str.length<100)) // line3
{System.out.print("compile failure");} // line4
else {System.out.print("end");}
}
```

- A. line1
- B. line2
- C. line3
- D. line4

14. Which following statement **cannot** create an array?

- A. `int[] myArray1 = [1,2,3];`
- B. `int[] myArray2 = new int[3];`
- C. `int[] myArray3 = {1,2,3};`
- D. `int[] myArray4 = new int[]{1,2,3};`

15. Which following code can count the total number of elements in an array?

- A. `arrayName.size`
- B. `arrayName.length`
- C. `arrayName.count`
- D. `arrayName.length()`

16. Which following statement of Math method is **not** correct?

- A. `ceil()` returns an integer that is greater than or equals to its argument.
- B. `log()` returns a natural logarithm of a number
- C. `sqrt()` returns a square root of a number
- D. `floor()` returns a closest value equal to an integer.

17. What is the result of `math.ceil(7.5)` & `math.floor(7.5)`?

- A. 7 8
- B. 7 7
- C. 8 8
- D. 8 7

18. What is the result of `math.ceil(-7.5)` & `math.floor(-7.5)`?

- A. -7 -8
- B. -7 -7
- C. -8 -8
- D. -8 -7

19. What is the result of `math.round(7.5)`, `math.round(-7.5)`?

- A. 8 -8
- B. 8 -7
- C. 7 -8
- D. 7 -7

20. What is the output?

```
int index=1;
```

```
int arr[ ]=new int[5];
```

```
int b=arr[index];
```

```
int a=b+ index;
```

A. a value is 0

B. a value is 1

C. a value is 2

D. compile failure

21. Which following can create an array?

A. array a=new Array(3);

B. int a[]=new int(3);

C. int a[]=new int[3];

D. int a[3]={1,2,3};

22. What is the value of b of the following code?

```
int a=3;
```

```
int b=10;
```

```
b=a++-1;
```

A. 2

B. 3

C. 4

D. 5

23. What is the value of b of the following code?

```
int a=3;
```

```
int b=10;
```

```
b=++a-1;
```

A. 2

B. 3

C. 4

D. 5

24. Which of the following is **not** an operator in java?

A. >>

B. <<

C. >>>

D. <<<

25. What is the output in following code?

```
public static void main(String args[ ]){
```

```
int val=-10;
```

```
System.out.println((val>"Negative"?10:"Negative");
```

```
}
```

A. 10

- B. Negative
- C. Positive
- D. compile failure

26. What is the output of the following code?

```
boolean a = true;  
boolean b = false;  
if( (!b) || (a));  
boolean c=(false?a:b);  
System.out.print(c);
```

- A. true
- B. false
- C. 0
- D. compile failure

27. Which following is a correct statement?

- A. if () { } else { }
- B. if () then { } else { }
- C. if () then { } then else { }
- D. if () then { } if else { }

28. What is the output?

```
int result = 0;
```

```
for( int n=1; n<=100; n++)  
result+=n;  
System.out.println(result);
```

What is the output?

- A. 100
- B. 1000
- C. 5000
- D. 5050**

29. What is the output?

```
int i=0;  
while(i< 10)  
{  
++i;  
if(i==3) break;  
System.out.println("The number is"+i);  
}
```

What is the output?

- A. The number is 1
- B. The number is 1 The number is 2**
- C. The number is 1 The number is 2 The number is 3
- D. The number is 1,2,3

30. What is the output?

```
int i=0;
while (i<4)
{
++i;
if ( i ==2) continue;
System.out.println("The number is"+i);
}
```

What is the result?

- A. The number is 1
- B. The number is 1 The number is 2
- C. The number is 1 The number is 3 The number is 4
- D. The number is 1,2,3

31. What will be the output?

```
char num = 1;
switch (num) {
case 1:
System.out.println ("one");
case 2:
System.out.println ("two");
case 3:
System.out.println ("three"); break;
default:
System.out.println ("four"); break;
}
```

- A. one
- B. one two
- C. one two three
- D. one two three four

32. What is the output?

```
public class Test {  
    public static void main (String args[ ]){  
        int i=1;  
        while (i){  
            if (i==3){  
                break;  
            }  
            ++i;  
            System.out.print(i);  
        }  
    }  
}
```

- A. 1
- B. 2
- C. 3
- D. compile failure

33. What is the output?

```
int x=5;
int y=6;
if (x=y){
System.out.println("5");}
else
System.out.println("6");
```

- A. 5
- B. 6
- C. 0
- D. compile failure

34. What is the output in following code?

```
boolean test=true;
if (test=false) {System.out.println("One");}
else if (test) {System.out.println("Two");}
else {System.out.println("Three");}
```

- A. One
- B. Two
- C. Three
- D. Compile failure

35. What is the output in following code?

```
char num='b';
switch (num) {
```

```
case 'a' : System.out.print('A');  
case 'b' : System.out.print('B');  
case 'c' : System.out.print('C');  
default:  System.out.print('D'); break;  
}
```

- A. B
- B. BD
- C. BCD
- D. Compile failure

36. Which following line is **not** correct?

```
import java.util.Date; // line1  
package myPackage;  
class MyClass1 extends Date {...} // line2  
class MyClass2 extends Date {...} // line3  
class MyClass3 extends Date {...} //line4
```

- A. line 1
- B. line 2
- C. line 3
- D. line 4

37. Which following statement is **not** correct?

```
int a=1;  
A. do{ break;} while ( a>0 );
```

B. `if(a>0) { break;}`

C. `switch (a) { default : break;}`

D. `for(;a>0;) break;`

38. What is the output in the following code?

```
public class Test {  
    public static void main (String args[ ]){  
        int num=10;  
        if(num>0)  
            if(num>100)  
                if(num>1000)  
                    System.out.println("1000");  
            else  
                System.out.println("100");  
        else  
            System.out.println("10");  
    }  
}
```

A. compile failure

B. 1000

C. 100

D. 10

39. For `if (expression)`, which following type is correct?

- A. if(integer-expression) ...
- B. if(char-expression) ...
- C. if(float-expression)...
- D. if(boolean-expression)...

40. Given code:

```
System.out.print(a[n]);
```

Which following code can display all elements in array int a[]?

- A. for(int n=0;n<a.length-1; n++)
- B. for(int n=0;n<a.length+1; n++)
- C. for(int n=0;n<a.length; n++)
- D. for(int n=0;n<a.length(); n++)

41. What is the output of the following code?

```
int num=2;
```

```
if (++num==num++)
```

```
System.out.print("A");
```

```
else
```

```
System.out.print("B");
```

- A. A
- B. B
- C. 2
- D. compile failure

42. What is the output of the following code?

```
String x="A";  
boolean[ ] y=new boolean[10];  
if (y[1]) { x="B";}  
else { System.out.print(x); }
```

A. A

B. B

C. “ “

D. null

43. What is the output of the following code?

```
int num=100;  
switch(num){  
default: System.out.println("default");  
case 0: System.out.println("A"); break;  
case 1: System.out.println("B"); break;  
case 2: System.out.println("C"); break;
```

A. A

B. default

C. default A

D. compile failure

44. What is the output in the following code?

```
for (int n=1; n<20; n++){  
    if(n%9==0)  
        break;  
    if(n%3==1)  
        continue;  
    System.out.print(n);  
}
```

- A. 24689
- B. 21678
- C. 26137
- D. 23568**

45. What is the output of the following code?

```
int n=0;  
while(true){  
    if(n++>10) break;  
}  
System.out.println(n);
```

- A. 11
- B. 12**
- C. 13
- D. compile failure

46. Which following statement can prevent creating sub class?

- A. static class test{ }
- B. public class test{ }
- C. abstract class test{ }
- D. final class test{ }

47. Which following line is **not** correct?

```
class Test extends SuperClass{  
int x=1; int y=2; // line1  
void fun( ){ // line2  
int y=0; // line3  
for (int y=10; y<100; y++){...} // line4  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

48. Which following line is **not** correct?

```
class test {  
int x=1; // line1  
int y=2; // line2  
}  
public class HelloWorld004{
```

```
public static void main(String args[ ]){  
test t=new test(1,2); // line3  
System.out.println(t.x+t.y); } // line4  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

49. Which following line is **not** correct?

```
class test {    // line1  
int myMethod( ){ return 10; }  
void myMethod( ){ int num =10; } // line2  
}  
  
public class HelloWorld004 {  
public static void main(String args[]){  
test t=new test( ); // line3  
t.myMethod( ); // line4  
}}
```

- A. line1
- B. line2
- C. line3
- D. line4

50. Which following line is **not** correct?

```
class test{ // line1
private int x=1, y=2; // line2
}
class subtest extends test{
int m=10; // line3
int n=m+x+y; // line4
}
```

- A. line1
- B. line2
- C. line3
- D. line4

51. Which following line is **not** correct?

```
class test{
private int x;
private int y;
test(int x, int y ){
this.x=x; // line1
this.y=y; // line2
}
}
class subtest extends test{
subtest(int x, int y) { //line3
```

```
int s = x + y;  
super(x, y); // line4  
}  
}  
.....
```

- A. line1
- B. line2
- C. line3
- D. line4

52. Which following statement is correct?

- A. Overriding occurs only in one class.
- B. Overloading occurs only in one class.
- C. Overriding occurs only between super class and sub class.
- D. Overloading occurs only between super class and sub class.

53. Which following line is **not** correct?

```
class test{ //line 1  
static int num=10; // line2  
static int myMethod( ){ // line3  
return this.num; // line4  
}  
}
```

- A. line1

- B. line2
- C. line3
- D. line4

54. Which following line is **not** correct?

```
class Test{  
void myMethod( ){  
static int num=10; // line1  
System.out.print(num); // line2  
}  
}  
  
public class HelloWorld004{  
public static void main (String args[ ]){  
Test obj=new Test( ); // line3  
obj.myMethod( ); // line4  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

55. Which following line is **not** correct?

```
class test{           // line1  
final int m=10, n =100; //line2  
public void myMethod( ){ //line3
```

```
System.out.print(m=m+n); //line4
}
}
```

- A. line1
- B. line2
- C. line3
- D. line4

56. Which following line is **not** correct?

```
class test{
final void myMethod(int num){ // line1
System.out.println(num); } // line2
}
class subtest extends test{
void myMethod(int num){ // line3
System.out.println(num+1); // line4
}
}
```

- A. line1
- B. line2
- C. line3
- D. line4

57. Which following line is **not** correct?


```
class test{  
private int x;    // line1  
private int y;  
test(int x, int y ){ // line2  
this.x=x;  
this.y=y;  
}  
}  
class subtest extends test{ // line3  
final subtest(int m, int n){ // line 4  
super(m, n);  
int sum=m+n;  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

58. Which following line is **not** correct?

```
abstract class car{  
abstract void drive( ); // line1  
}  
class redCar extends car{ // line2  
void driving( ){ // line3
```

```
System.out.println("Driving red car."); // line4  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

59. Which following line is **not** correct?

```
abstract class car{  
    abstract void drive(){ }; // line1  
}  
class redCar extends car{ // line2  
    void drive(){ // line3  
        System.out.println("Driving red car."); // line4  
    }  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

60. Which following statement is **not** correct?

- A. abstract and final cannot be used at the same time to modify a class.
- B. abstract method can be placed inside or outside the abstract class.
- C. abstract class cannot contain private member.
- D. abstract and static cannot be used at the same time to modify a method.

61. Which following line is **not** correct?

```
interface test{  
int num; // line1  
int myMethod( ); // line2  
}  
class newTest implements test{  
public int myMethod( ){ // line3  
return 100; // line4  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

62. Which following line is **not** correct?

```
interface test{
double PI=3.14159; // line1
int myMethod( ); // line2
}
class newTest implements test{
int myMethod( ){ // line3
return 100; // line4
}
}
```

- A. line1
- B. line2
- C. line3**
- D. line4

63. Which following line is **not** correct?

```
class test{
private test( ){ } // line1
static test myMethod( ){return new test( ); } // line2
}
public class HelloWorld004{
public static void main (String args[ ]){
test t=new test( ); // line3
t.myMethod( ); // line4
}
}
```

- A. line1
- B. line2
- C. line3
- D. line4

64. Which following line is **not** correct?

```
class test{  
public int num=10; // line1  
}  
protected class newTest extends test{ // line2  
public test t=new test( ); // line3  
public void myMethod( ){ // line4  
System.out.println("OK!");  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

65. Which following statement is **not** correct?

- A. The protected member can be accessed by all members in the same class.

- B. The protected member can be accessed by all members in the same package.
- C. The protected member can be accessed by all members in the sub class.
- D. The protected member can be accessed by all members in the different package.

66. Which following line is **not** correct?

```
public class test{  
static int num; // line1  
num=10; // line2  
public static void main (String args[ ]){ // line3  
System.out.print(num); // line4  
}  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

67. What is the output of the following code?

```
public class test{  
public static void main (String args[ ]){  
String s=new String("true");
```

```
Boolean b=new Boolean(true);  
if(s.equals (b)){  
System.out.print("true");  
}  
}  
}
```

- A. true
- B. false
- C. no output
- D. compile failure

68. Which following declaration is **not** correct?

- A. final class test
- B. public class test
- C. abstract class test
- D. private class test

69. Which following line is **not** correct?

```
class superClass { ... }  
class subClass extends superClass { ... }  
...  
superClass a=new superClass( ); // line1  
subClass b=new subclass( ); // line2  
.....
```

a=b; // line3

b=a; // line4

A. line1

B. line2

C. line3

D. line4

70. Which following line is **not** correct?

interface A{ // line 1

double PI=3.14159; // line 2

void myMethod(); // line 3

}

class B implements A{ } // line 4

A. line1

B. line2

C. line3

D. line4

71. Which following line is **not** correct?

final class test{

void myMethod () { // line1

int num=10; // line2

System.out.print(num); // line3

}

}

class subtest extends test{... } // line4

- A. line1
- B. line2
- C. line3
- D. line4

72. Which following statement is correct?

- A. The name of the constructor method can be different from the name of the class.
- B. Constructor method cannot have the return value, so it needs a void return type before its name.
- C. To call a constructor method in the super class, you can use "super" statement in sub class.
- D. Any class must define a constructor method explicitly, so that it initializes all members in the class.

73. Which following statement is **not** correct?

- A. The "super" statement must place in the first line in the sub class's constructor method.
- B. The "super" statement can be used to the static method.
- C. Abstract method cannot have a method body, and it must be placed inside the abstract class.
- D. Abstract class cannot use "new" statement to create an object.

74. What is the output of the following code?

```
class superClass{
superClass( ){System.out.print("A");}
}
public class subClass extends superClass{
subClass( ){System.out.print("B");}

public static void main (String args[ ]){
superClass a=new superClass( );
subClass b=new subClass( );
}
}
```

A. A

B. B

C. AB

D. AAB

75. Which line of code is **not** correct for overloading or overriding?

```
class superClass{
public int myMethod( ) { return 10; }
}
public class subclass extends superClass{
/*The following codes will be inserted here for overloading or overriding.
*/
```

}

- A. public int myMehtod(){ }
- B. public double myMehtod(double d){ }
- C. public void myMehtod(){ }
- D. public float myMehtod(float f){ }

76. Which following code is correct?

- A. abstract class test {abstract void myMethod();}
- B. class abstract test {abstract void myMethod();}
- C. abstract test {abstract void myMethod();}
- D. abstract class test {abstract void myMethod(){...};}

77. Which following line is **not** correct?

```
class test{  
int x=10; // line1  
static int y=100; // line2  
static void method1(int c ){  
x = c; // line3  
}  
public void method2( ){  
y=200; // line4  
}  
}
```

- A. line1

- B. line2
- C. line3
- D. line4

78. Which following line is **not** correct?

```
public class test{ // line1
public int num=10; // line2
public static void main (String args[ ]){ // line3
System.out.print("num="+num); // line4
}
}
```

- A. line1
- B. line2
- C. line3
- D. line4

79. Given a code:

```
test obj=new test( );
```

If you want to free the memory space of variable obj, which following code is suitable?

- A. obj="" ; System.finalize();
- B. obj="" ; System.gc();
- C. obj=null; System.finalize();
- D. obj=null; System.gc();

80. If you want to restrict the access of member to current class from another class, which following modifier should you choose?

- A. default
- B. private**
- C. public
- D. protected

81. Which following statement is **not** correct?

- A. final variable cannot be modified.
- B. final method cannot be overridden.
- C. final object cannot be accessed.**
- D. final class cannot be extended.

82. Which following line is **not** correct?

```
class ABC{  
private int a=10;  
protected int b=20;  
int c=30;  
void d ( ) {  
System.out.print("ok");  
}  
}
```

```
public class test{  
    public static void main (String args[ ]){  
        ABC obj=new ABC ( );  
        obj.a=100; // line1  
        obj.b=200; // line2  
        obj.c=300; // line3  
        obj.d( ); // line4  
    }  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

83. What is the output of the following code?

```
public class test{  
    public test(){  
        System.out.print("A");  
    }  
    public static void main (String args[ ]){  
        test t=new test( );  
        t.test( );  
        System.out.print("B");  
    }  
}
```

- A. AB
- B. AAB
- C. AABB
- D. compile failure

84. What is the output of the following code?

```
class test{  
    int a=10;  
    String b="str1";  
    test(){  
        a=20;  
        b="str2";  
    }  
    public static void main (String args[ ]){  
        test obj=new test( );  
        System.out.print(obj.a+obj.b);  
    }  
}
```

- A. 10str1
- B. 10str2
- C. 20str1
- D. 20str2

85. Which following line is **not** correct?

```
abstract class abstractTest{ // line1
abstract int myMethod( ); // line2
}
public class test extends abstractTest{ // line3
private int myMethod( ) {return 10;} // line4
}
```

- A. line1
- B. line2
- C. line3
- D. line4**

86. What is the output in the following code?

```
package myPackage;
public class HelloWorld {
public static void main (String args[ ]){
String s1=new String("ok");
String s2=new String("ok");
if (s1==s2){
System.out.print("equal");
}
else{
System.out.print("not equal");
}}}
```

- A. no output
- B. equal

C. not equal

D. compile failure

87. Which following line is **not** correct?

```
public class Test {  
    public static void main (String args[ ]){  
        int a[ ]= {0,1,2,3}; // line1  
        System.out.print(a.length); // line2  
        String str="Java"; // line3  
        System.out.print(str.length); // line4  
    }  
}
```

A. line1

B. line2

C. line3

D. line4

88. Which following statement is correct?

```
public String substring (int m, int n)
```

A. return substring from m to n.

B. return substring from m-1 to n-1.

C. return substring from m-1 to n.

D. return substring from m to n-1.

89. Which following statement is **not** correct when comparing String object?

- A. compareTo() is used to compare two objects' value by dictionary order. Returning zero means equal.
- B. compareObjects() is used to compare two String objects by its value.
- C. equals() is used to compare two String objects by its value.
- D. == is used to compare two String objects to identify if they are the same object.

90. Which following statement is **not** correct when connect two strings?

- A. operator "+" can connect two StringBuffer objects.
- B. operator "+" can connect two String objects.
- C. append() can connect two StringBuffer objects.
- D. concat() can connect two String objects.

91. Which following code can create an array with 3 empty strings?

- A. String arr [3];
- B. String arr []={null, null, null};
- C. String arr []={"", "", ""};
- D. String arr []=new String [3];
for (int i=0; i<3; arr[i++]=null);

92. Which following statement is **not** correct?

- A. String class is a final class. It cannot have sub class.
- B. String class is included in the package java.lang.
- C. "ok".equals("ok") will return true.
- D. String a="ok", b="ok"; a.compareTo(b) will return true.

93. What is the output in the following code?

```
public static void main (String args[ ]){  
    String str1="AB";  
    String str2="CD";  
    str2.toLowerCase( );  
    System.out.print(str1+str2);  
}
```

- A. ABcd
- B. ABCD
- C. abcd
- D. compile failure

94. Which following line is **not** correct?

```
public static void main (String args[ ]){  
    String s1;    //line1  
    StringBuffer s2 = new StringBuffer("String2"); //line2  
    s1=s2.concat(" String1").toString(); //line 3  
    System.out.println(s1); //line4
```

}

- A. line1
- B. line2
- C. line3
- D. line4

95. Which following line is **not** correct?

StringBuffer str1, str2; // line1

str1="OK"; // line2

str2=new StringBuffer("OK"); // line3

System.out.println(str2); // line4

- A. line1
- B. line2
- C. line3
- D. line4

96. Which following line is **no** good practice in programming?

int a=100, b=0, // line1

c=a/b; // line2

try{ // line3

a=200/b; // line4

}

catch(ArithmeticException e){

System.out.println("Divided by zero.");

}

- A. line1
- B. line2
- C. line3
- D. line4

97. Which following line is **not** correct?

```
try{  
int a, b, // line1  
c=a/b; // line2  
}  
catch(RuntimeException e)  
{  
System.out.println("Divided by zero.");  
}  
catch(ArithmeticException e) //line 3  
{  
System.out.println("Divided by zero"); //line 4  
}
```

- A. line1
- B. line2
- C. line3
- D. line4

98. The common super class of Error and Exception is ___?

- A. Throwable
- B. AWTError
- C. AWTException
- D. ErrorException

99. Which following statement is correct?

- A. An error is an Exception.
- B. IOException is a sub class of EOFException.
- C. Any codes that maybe throw an error should be placed inside the try{ } block.
- D. Any codes that maybe throw an exception should be placed inside the try{ } block.

100. Which following operation may **not** throw exception?

- A. Want to open a file that does not exist.
- B. A float or integer divided by zero.
- C. When the index of an array is out of bounds
- D. Return -1 when use indexOf() to search a substring.

101. Which following line will throw exception?

```
try{  
int a=10, b=20, c=0;  
if(a==b && a==b/c) // line 1
```

```
System.out.print("A");    // line 2
else if ( a==b || a==b/c ) // line 3
System.out.print("B");    // line 4
}
catch(ArithmeticException e)
{
System.out.println("Divided by zero");
}
```

- A. line1
- B. line2
- C. line3
- D. line4

102. Which following code is **not** correct?

- A. public void myMethod() throws Exception
- B. public void myMethod() throw Exception
- C. public void myMethod() throws RuntimeException
- D. public void myMethod() throws ArithmeticException

103. Which exception should be thrown so as to pass compile?

```
class testException extends test{
public void myMethod( ) throws TimeoutException{ }}
public void myTest( ) throws _____{
}
```

```
{  
myMethod( );  
}  
private void myMethod() {  
}  
}
```

A. Exception

B. EOFException

C. AWTException

D. IOException

104. What is the output of the following code?

```
try  
{  
System.out.print("A");  
return;  
}  
catch(RuntimeException e)  
{  
System.out.print("B");  
}  
finally  
{  
System.out.print("Finally");  
}
```



```
}  
}
```

- A. A
- B. B
- C. AB
- D. **AFinally**

105. Which following object can use “throws” when exception occurred?

- A. Event
- B. Object
- C. String
- D. **EOFException**

106. java.io Package does **not** include _____.

- A. InputStream Class
- B. OutputStream Class
- C. File Class
- D. **FileDescription Class**

107. File class is **not** used to _____.

- A. check if file exists.
- B. **read files or write files.**
- C. check if file is readable or is writable.
- D. return the length of a file.

108. Which following statement is **not** correct?

- A. FileInputStream Class is the sub class of InputStream Class.
- B. FileOutputStream Class is the sub class of OutputStream Class.
- C. FileInputStream and FileOutputStream can write or read byte data from a file.
- D. FileInputStream and FileOutputStream can write or read byte, character, integer, and one line of data from a data stream.

109. Which following statement is **not** correct?

- A. DataInputStream Class is the sub class of InputStream Class.
- B. DataOutputStream Class is the sub class of OutputStream Class.
- C. DataInputStream and DataOutputStream only can write or read byte data from a file, but cannot write or read various types of data.
- D. DataInputuStream and DataOutputStream can write or read byte, character, integer and one line of data from a data stream.

110. What is the output of the following code?

```
FileInputStream in=new FileInputStream("myfile.java");  
byte b[ ]=new byte[10];  
int d = 0;
```

```
try {  
    d = in.read(b);  
} catch (IOException e) {  
    e.printStackTrace();  
}  
System.out.print(d);
```

A. 10

B. 20

C. exception occurs

D. compile failure.

111. If the size of a file myFile.txt is 100 bytes, then after running
File f=new File("myFile.txt");
System.out.print(f);

The output is_____.

A. 100.

B. myFile.txt.

C. 0.

D. Compile failure.

112. When using read() in java.io.InputStream or write() in
java.io.OutputStream, you may handle an exception which is _____.

A. java.io.IOException

- B. `java.io.OutputException`
- C. `java.io.InputOutputException`
- D. `java.io.IOException`

113. Which following statement is **not** correct?

- A. There are two kinds of main streams: one is an input stream, another is an output stream.
- B. `InputStream` and `OutputStream` are a super class stream of all other class stream.
- C. In order to support input/output device, Java defines two stream objects. One is `System.in` object. Another is `System.out` object.
- D. `PushbackInputStream` class has the function of rejecting data stream.

114. Which following stream can output character?

- A. `java.io.OutputStreamWriter`
- B. `java.io.OutputStream`
- C. `java.io.BufferedOutputStream`
- D. `java.io.FileOutputStream`

115. Which following stream can input character?

- A. `java.io.InputStreamReader`
- B. `java.io.InputStream`

- C. java.io.BufferedInputStream
- D. java.io.FileInputStream

116. What is the output of the following code?

```
String str1="Java 100 Tests.";
String str2=str1;
String str3=new String("Java 100 Tests.");
if (str3. equals (str2))
{System.out.print("Equal Value!");}
else if(str3==str2)
{System.out.print("Different Object!");}
else
{System.out.print("The End. Thank you!");}
```

- A. Equal Value!
- B. Different Object!
- C. Compile Failure!
- D. The End. Thank you!

100 JAVA Answers

001.D	026.B	051.D	076.A	101.C
002.C	027.A	052.C	077.C	102.B
003.C	028.D	053.D	078.D	103.A
004.D	029.B	054.A	079.D	104.D
005.D	030.C	055.D	080.B	105.D
006.A	031.C	056.C	081.C	106.D
007.C	032.D	057.D	082.A	107.B
008.B	033.D	058.C	083.D	108.D
009.D	034.C	059.A	084.D	109.C
010.C	035.C	060.B	085.D	110.C
011.D	036.A	061.A	086.C	111.B
012.C	037.B	062.C	087.D	112.D
013.C	038.D	063.C	088.D	113.D
014.A	039.D	064.B	089.B	114.A
015.B	040.C	065.D	090.A	115.A
016.D	041.A	066.B	091.C	116.A
017.D	042.A	067.C	092.D	
018.A	043.C	068.D	093.B	
019.B	044.D	069.D	094.C	
020.B	045.B	070.D	095.B	
021.C	046.D	071.D	096.B	
022.A	047.D	072.C	097.C	
023.B	048.C	073.B	098.A	
024.D	049.B	074.D	099.D	
025.D	050.D	075.C	100.D	