

jsp, servlet, exceptions, syntaxes

```
statement = connection.createStatement(  
    java.sql.ResultSet.TYPE_FORWARD_ONLY,  
    java.sql.ResultSet.CONCUR_READ_ONLY  
);  
statement.setFetchSize(Integer.MIN_VALUE);
```

Which isolation level prevents dirty read in JDBC, connection class.

- (A) TRANSACTION_READ_COMMITTED
- (B) TRANSACTION_UNREAD_COMMITTED

Isolation Level	Table-Level Locking	Row-Level Locking
-----------------	---------------------	-------------------

TRANSACTION_READ_UNCOMMITTED	Dirty reads, nonrepeatable reads, and phantom reads possible	Dirty reads, nonrepeatable reads, and phantom reads possible
TRANSACTION_READ_COMMITTED	Nonrepeatable reads and phantom reads possible	Nonrepeatable reads and phantom reads possible
TRANSACTION_REPEATABLE_READ	Phantom reads not possible because entire table is locked	Phantom reads possible
TRANSACTION_SERIALIZABLE	None	None

TYPE_SCROLL_INSENSITIVE:

The result can be scrolled; its cursor can move both forward and backward relative to the current position, and it can move to an absolute position. The result set is insensitive to changes made to the underlying data source while it is open. It contains the rows that satisfy the query at either the time the query is executed or as the rows are retrieved.

TYPE_SCROLL_SENSITIVE:

The result can be scrolled; its cursor can move both forward and backward relative to the current position, and it can move to an absolute position. The result set reflects changes made to the underlying data source while the result set remains open.

What is the effect of executing the following JSP statement, assuming a class with name Employee exists in classes package.

```
<%@ page import = "classes.Employee" %> <jsp:useBean id="employee" class="classes.Employee" scope="session"/> <jsp:setProperty name="employee" property="*" />
```

The code does not compile as there is no property attribute of setProperty tag.

The code does not compile as property attribute cannot take * as a value.

The code sets value of all properties of employee bean to "*".

Ans) The code sets the values of all properties of employee bean to matching parameters in request object.

bookmark is in get.

```
1. class Animal {
    public String noise() {
        return "peep";
    }
}
class Dog extends Animal {
    public String noise() {
        return "bark";
    }
}
class Cat extends Animal {
    public String noise() {
```

```

        return "meow";
    }
}
...
30. Animal animal = new Dog();
31. Cat cat = (Cat)animal;
32. System.out.println(cat.noise());

```

What is the result?

- A. peep
- B. bark
- C. meow
- D. Compilation fails.
- E. An exception is thrown at runtime.

Answer: E

```

1. class A {
2. void foo() throws Exception {
3. throw new Exception();
4. }
5. }
6.
7. class SubB2 extends A {
8. void foo() {
9. System.out.println("B ");
10. }
11. }
12. class Tester {
13. public static void main(String[] args) {
14. A a = new SubB2();
15. a.foo();
16. }
17. }

```

What is the result?

- A. B
- B. B, followed by an Exception.

- C. Compilation fails due to an error on line 9.
- D. Compilation fails due to an error on line 15.
- E. An Exception is thrown with no other output

Answer: D

Section: All

Explanation/Reference:

Unhandled exception type Exception

2. legal `s3=s2+s1`

3. when we are sending data in URL in get method how to separate

parameters? ANs) &

when we are sending data in URL in get method how to separate

parameter and separate path? ANs) ?

```
4. class multithreaded_programing{
    public static void main(String args[]){
        Thread t = Thread.currentThread();
        t.setName("new thread");
        System.out.println(t);
    }
}
```

ans- `thread[new thread,5,main]`

5. current thread

```
public class MyRunnable implements Runnable{
    public void run(){
        //some code
    }
}
```

which of these will create and start this thread?

Ans) `new Thread(new MyRunnable()).start();`

Description: Because the class implements Runnable, an instance of it has to be passed to the Thread co

nstructor

and then the instance of the Thread has to be started.

```
6. class Mammal { }
```

```
    class Raccoon extends Mammal {  
        Mammal m = new Mammal();  
    }
```

```
class BabyRaccoon extends Mammal { }
```

Which four statements are true? (Choose four.)

- A. Raccoon is-a Mammal.
- B. Raccoon has-a Mammal.
- C. BabyRaccoon is-a Mammal.
- D. BabyRaccoon is-a Raccoon.
- E. BabyRaccoon has-a Mammal.
- F. BabyRaccoon is-a BabyRaccoon.

Answer: A,B,C,F

Explanation :Check is-A and Has-A relationship between both classes.

7. What is the output of this program?

```
import java.io.*;  
  
public class filesinputoutput {  
    public static void main(String[] args)
```

```

{
    String obj  = "abc";

    byte b[] = obj.getBytes();

    ByteArrayInputStream obj1 = new Byte
ArrayInputStream(b);

    for (int i = 0; i < 2; ++ i) {

        int c;

        while ((c = obj1.read()) != -1)
        {

            if (i == 0) {

                System.out.print(Character.toUpperCase((char)c)); }}}}
        ans) ABC

```

```

8.  class output {

    public static void main(String args[])

    {

        StringBuffer s1 = new StringBuffer("
Hello World");

        s1.insert(6 , "Good ");

        System.out.println(s1);

    }

}

```

- a) HelloGoodWorld
- b) HellGoodoWorld
- c) HellGood oWorld

d) Hello Good World

Answer: d

Explanation: The insert() method inserts one string into another. It is overloaded to accept values of all simple types, plus String and Objects. Sting is inserted into invoking object at

specified position. "Good " is inserted in "Hello World" T index 6 giving "Hello Good World".

output:

```
$ javac output.java
```

```
$ java output
```

```
Hello Good World
```

9. string index(i)

10. Which of these keywords cannot be used for a class which has been declared final?

a) abstract

b) extends

c) abstract and extends

d) None of the mentioned

View Answer

Answer: a

Explanation: A abstract class is incomplete by itself and relies upon its subclasses to provide complete

implementation. If we declare a class final then no class can inherit that class, an abstract class needs its subclasses

hence both final and abstract cannot be used for a same class.

11. import all- import .*

12. abstract class abc{
abstract ...(int a, int b, int c)

Q= method declared right?

13. sos('1'+new integer(2)+3);

ANS-123

```
14.int a[]=new int[]{1,2,6,4,5};
    int b[]=new int[5];
    b[2]=a[2];
    a[2]=10;
    System.out.println(b[2]);
    ans: 6
```

15. GENERICS STRING TAG QUESTION

16. What gets printed when the following JSTL code fragment is executed? Select the one correct answer.

```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core"
prefix="c" %>
<c:set var="item" value="2"/>
<c:if test="${var==1}" var="result" scope="session"
>
<c:out value="${result}"/>
</c:if>
```

A.The JSTL code does not compile as attribute f or if tag are not correct.

B.true

C. false

Ans D. Nothing gets printed.

17. ideal
removin loop
-operations
a)idea
b)deal

18.The sendError method defined in the HttpServlet class is
equivalent to invoking the setStatus method with th
e following
parameter. Select the one correct answer.

4.

A. SC_OK

B. SC_MOVED_TEMPORARILY

- C. SC_NOT_FOUND Ans) C
- D. SC_INTERNAL_SERVER_ERROR
- E. ESC_BAD_REQUEST
- 5. The sendRedirect method defined

The sendRedirect method defined in the HttpServlet class is equivalent to invoking the setStatus method with the following parameter and a Location header in the URL. Select the one correct answer.

- A. SC_OK
- B. SC_MOVED_TEMPORARILY Ans) B
- C. SC_NOT_FOUND
- D. SC_INTERNAL_SERVER_ERROR
- E. ESC_BAD_REQUEST

19.doget is in which? a)request.httpresponse ? b)response.http?

20.jsp-attributes: differentiate a), b)& c)?

21. independently vary=bridge

22. factory set

23. A team of programmers is involved in reviewing a proposed design for a new utility class. After some discussion, they realize that the current design allows other classes to access methods in the utility class that should be accessible only to methods within the utility class itself.

What design issue has the team discovered?

- A. Tight coupling
- B. Low cohesion
- C. High cohesion
- D. Loose coupling
- E. Weak encapsulation
- F. Strong encapsulation

Answer: E

24. instanceof

25. man dog best friend `ans: class man { private dog best friend; }`

26. `push` pop stack

27. blue mmix deploy cloud foundry `ans) cf push`

28. `1..*` At least one instance

29. `window if class=underline`

30. `abstract class-` italics

31. `slant line for delay`

32. hollow diamond on collection side for aggregation, filled for composition and hollow triangle for generalization.

33. jstl

34. arrow towards mammal

35. iterator question

36. `hasnext()`, `traverse`

37. junit testing in which order

38. scroll insensitive

39. What is the output of this program?

```
import java.util.*;

class vector {

    public static void main(String args[])
    {

        Vector obj = new Vector(4,2);

        obj.addElement(new Integer(3));

        obj.addElement(new Integer(2));

        obj.addElement(new Integer(5));

        obj.removeAll(obj);

        System.out.println(obj.isEmpty());
    }
}
```

```
    }  
}
```

- a) 0
- b) 1
- c) true
- d) false

View Answer

Answer: c

Explanation: firstly elements 3, 2, 5 are entered in the vector obj, but when `obj.removeAll(obj);` is executed all the elements are deleted and vector is empty, hence `obj.isEmpty()` returns true.

Output:

```
$ javac vector.java  
$ java vector  
true
```

```
40.      import java.util.*;  
  
      class Array {  
  
          public static void main(String args[])  
{  
  
              int array[] = new int [5];  
              for (int i = 5; i > 0; i--)  
                  array[5 - i] = i;  
  
              Arrays.sort(array);  
  
              for (int i = 0; i < 5; ++i)  
                  System.out.print(array[i]);  
  
          }  
  
      }
```

} Ans) 12345

41. What is the output of this program?

```

import java.util.*;

class Maps {

    public static void main(String args[])
    {

        TreeMap obj = new TreeMap();

        obj.put("A", new Integer(1));

        obj.put("B", new Integer(2));

        obj.put("C", new Integer(3));

        System.out.println(obj.entrySet());

    }

}

```

- a) [A, B, C]
- b) [1, 2, 3]
- c) {A=1, B=2, C=3}
- d) [A=1, B=2, C=3]

Answer: d

Explanation: obj.entrySet() method is used to obtain a set that contains the entries in the map. This method provides set view of the invoking map.

Output:

```

$ javac Maps.java
$ java Maps
[A=1, B=2, C=3]

```

42. What is the output of this program?

```

import java.util.*;

class Bitset {

```

```

        public static void main(String args[])
        {
            BitSet obj = new BitSet(5);
            for (int i = 0; i < 5; ++i)
                obj.set(i);
            obj.clear(2);
            System.out.print(obj);
        }
    }

```

- a) {0, 1, 3, 4}
- b) {0, 1, 2, 4}
- c) {0, 1, 2, 3, 4}
- d) {0, 0, 0, 3, 4}

View Answer

Answer: a

Explanation: None.

Output:

\$ javac Bitset.java

\$ java Bitset

{0, 1, 3, 4}

43. What is the output of this program?

```

import java.util.*;

class Array {
    public static void main(String args[])
    {
        int array[] = new int [5];
        for (int i = 5; i > 0; i--)

```

```

        array[5-i] = i;

        Arrays.fill(array, 1, 4, 8);

        for (int i = 0; i < 5 ; i++)

            System.out.print(array[i]);

    }

}

```

advertisements

- a) 12885
- b) 12845
- c) 58881
- d) 54881

View Answer

Answer: c

Explanation: array was containing 5,4,3,2,1 but when method `Arrays.fill(array, 1, 4, 8)` is called it fills the index location starting with 1 to 4 by value 8 hence array becomes 5,8,8,8,1.

Output:

```

$ javac Array.java
$ java Array
58881

```

44. prefix uri

45. last; or;last

46. What is the output of this program?

```

import java.util.*;

class Collection_Algos {

    public static void main(String args[])
    {

        LinkedList list = new LinkedList();
    }
}

```

```

        list.add(new Integer(2));
        list.add(new Integer(8));
        list.add(new Integer(5));
        list.add(new Integer(1));
        Iterator i = list.iterator();
        Collections.reverse(list);
        Collections.shuffle(list);
        while(i.hasNext())
            System.out.print(i.next() + " ");
    }
}

```

- a) 2 8 5 1
- b) 1 5 8 2
- c) 1 2 5 8
- d) Any random order

Answer: d

Explanation: shuffle - randomizes all the elements in a list.

Output:

```
$ javac Collection_Algos.java
```

```
$ java Collection_Algos
```

```
1 5 2 8
```

(output will be different on your system)

47.What is the output of this program?

```

import java.util.*;

class Collection_Algos {

```

```

        public static void main(String args[])
        {
            LinkedList list = new LinkedList();

            list.add(new Integer(2));
            list.add(new Integer(8));
            list.add(new Integer(5));
            list.add(new Integer(1));

            Iterator i = list.iterator();

            Collections.reverse(list);

            Collections.sort(list);

            while(i.hasNext())

                System.out.print(i.next() + " ");

        }
    }

```

- a) 2 8 5 1
- b) 1 5 8 2
- c) 1 2 5 8
- d) 2 1 8 5

Answer: c

Explanation: Collections.sort(list) sorts the given list, the list was 2->8->5->1 after sorting it became 1->2->5->8.

Output:

when reverse 1 5 8 2

but sorting 1 2 5 8 ans is C

48. URL from a remote location protocol http is used. a)http b)https c)both d)none [Ans. c]

49. `<%@ taglib uri="http://java.sun.com/jsp/jstl/core"`

`prefix="c" %>`

`<c:set var="item" value="2"/>`

`<c:forEach var="item" begin="0" end="0" step="2">`

`<c:out value="${item}" default="abc"/>`

`</c:forEach>`

The JSTL code does not compile as an attribute for
forEach

tag is not correct. Ans A

A. 0

B. 2

C. ABC

D. Nothing gets printed as c.out statement does not
get executed.

50. `<%@ taglib uri="http://java.sun.com/jsp/jstl/core"`

`prefix="c" %>`

`<c:set var="j" value="4,3,2,1"/>`

`<c:forEach items="${j}" var="item" varStatus="status"`

`<c:if test="${status.first}">`

`<c:out value="${status.index}" default="abc"/>`

`</c:if>`

`</c:forEach>`

A. 1

B. 2

C. 3

D. 4

E. abc

F. The program does not compile

51. What is the output of this program?

```
import java.util.*;
```

```
public class genericstack <E> {
```

```

        Stack <E> stk = new Stack <E>();

        public void push(E obj) {

            stk.push(obj);

        }

        public E pop() {

            E obj = stk.pop();

            return obj;

        }

    }

    class Output {

        public static void main(String args[])

        {

            genericstack <String> gs = new genericstack<String>();

            gs.push("Hello");

            System.out.print(gs.pop() + " ");

            genericstack <Integer> gs = new genericstack<Integer>();

            gs.push(36);

            System.out.println(gs.pop());

        }

    }

```

a) Error

- b) Hello
- c) 36
- d) Hello 36

Answer: d

Explanation: None.

Output:

```
$ javac Output.java
```

```
$ java Output
```

```
Hello 36
```

```
52. class output {  
    public static void main(String args[])  
    {  
        StringBuffer c = new StringBuffer(  
"Hello");  
        StringBuffer c1 = new StringBuffer  
(" World");  
        c.append(c1);  
        System.out.println(c);  
    }  
}
```

- a) Hello
- b) World
- c) Helloworld
- d) Hello World Ans-d.

```
53.class output {  
    public static void main(String args[])  
    {
```

```

        StringBuffer s1 = new StringBuffer("
Hello");

        s1.setCharAt(1, 'x');

        System.out.println(s1);

    }

}

```

a) xello

b) xxxxx

c) Hxlllo

d) Hexlo

View Answer

Answer: c

Explanation: None.

Output:

\$ javac output.java

\$ java output

Hxlllo

54. switch case

55. array of string

56. "welcome.trim()"= when compiled ans) equals

57. builder pattern

58. Question on this:

```
Calendar calendar = new GregorianCalendar();
```

```
int year      = calendar.get(Calendar.YEAR);
```

```
int month     = calendar.get(Calendar.MONTH);
```

```
int dayOfMonth = calendar.get(Calendar.DAY_OF_MONTH); // Jan = 0, not 1
```

```
int dayOfWeek  = calendar.get(Calendar.DAY_OF_WEEK);
;
```

```
int weekOfYear = calendar.get(Calendar.WEEK_OF_YEAR);
;
```

```
int weekOfMonth= calendar.get(Calendar.WEEK_OF_MONTH);
;
```

```

int hour          = calendar.get(Calendar.HOUR);
    // 12 hour clock
int hourOfDay     = calendar.get(Calendar.HOUR_OF_DAY)
; // 24 hour clock
int minute        = calendar.get(Calendar.MINUTE);
int second        = calendar.get(Calendar.SECOND);
int millisecond= calendar.get(Calendar.MILLISECOND)
;

```

```

59. class output {

        public static void main(String args[])

        {

                String s1 = "one";

                String s2 = s1 + " two";

                System.out.println(s2);

        }

}

```

- a) one
- b) two
- c) one two
- d) compilation error

View Answer

Answer: c

60. Total number of users using listners - ans. httpsessionlistner

```

61. class exception_handling {

        public static void main(String args[])

        {

```

```

        try {
            System.out.print("Hello" + " "
+ 1 / 0);

        }

        catch(ArithmeticException e) {
            System.out.print("World");

        }

    }

}

```

- a) Hello
- b) World
- c) HelloWorld
- d) Hello World

Answer: b

```

62. class output {

    public static void main(String args[])

    {

        char c[]={'a', '1', 'b' , ' ' , 'A' ,

'0'};

        for (int i = 0; i < 5; ++i)

        {

            if(Character.isDigit(c[i]))

                System.out.println(c[i]+

```

```

" is a digit");

                                if(Character.isWhitespace(c[
i]))

                                System.out.println(c[i]+
" is a Whitespace character");

                                if(Character.isUpperCase(c[i
]))

                                System.out.println(c[i]+
" is an Upper case Letter");

                                if(Character.isLowerCase(c[i
]))

                                System.out.println(c[i]+
" is a lower case Letter");

                                i=i+3;

                                }

                                }

                                }

```

a) a is a lower case Letter
is White space character
b) b is a lower case Letter
is White space character
c) a is a lower case Letter
A is a upper case Letter
d) a is a lower case Letter
0 is a digit
View Answer
Answer:c

62. Two questions on these.
Interfaces for Event Listeners-
Session Serialization--HttpSession

Request Lifecycle Event--ServletRequestListener

Session Attribute Binding--HttpSessionAttribute
Listener

Session LifecycleEvent--HttpsessionListener

Web Application Lifecycle Event--ServletContext
Listener

63. Which of the following describes the correct sequence of the steps involved in making a connection with a database.

1. Loading the driver
2. Process the results.
3. Making the connection with the database.
4. Executing the SQL statements.

(A) 1,2,3,4

(B) 1,3,4,2

(C) 2,1,3,4

(D) 4,1,2,3

Answer : 1,3,4,2

64. `import java.io.*;`

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

```
    {  
        InputStream obj = new FileInputStream(  
am("inputoutput.java");
```

```
        System.out.print(obj.available());
```

```
    }  
}
```

Note: inputoutput.java is stored in the disk.

a) true

b) false

c) prints number of bytes in file

d) prints number of characters in the file

View Answer

Answer: c

Explanation: `obj.available()` returns the number of bytes.

Output:

```
$ javac filesinputoutput.java
```



```
$ java filesinputoutput
```

```
1422
```

(Output will be different in your case)

```
65.import java.io.*;
    public class filesinputoutput {
        public static void main(String[] args)
        {
            String obj = "abc";
            byte b[] = obj.getBytes();
            ByteArrayInputStream obj1 = new Byte
ArrayInputStream(b);
            for (int i = 0; i < 2; ++ i) {
                int c;
                while((c = obj1.read()) != -1) {
                    if(i == 0) {
                        System.out.print(Character.toUpperCase((char)c));
                        obj2.write(1);
                    }
                }
                System.out.print(obj2);
            }
        }
    }
```

- a) AaBaCa
- b) ABCaaa
- c) AaaBaaCaa
- d) AaBaaCaaa

Answer: d

Explanation: None.

Output:

```
$ javac filesinputoutput.java
```

```
$ java filesinputoutput
```

```
AaBaaCaaa
```

66. Which of these methods are used to read in from file?

- a) get()
- b) read()

- c) scan()
- d) readfileInput()

View Answer

Answer: b

67. toString question

68. SC_OK

69. Which of the following allows non repeatable read in JDBC Connection?

- a. TRANSACTION_READ_UNCOMMITTED
- b. TRANSACTION_READ_COMMITTED
- c. TRANSACTION_SERIALIZABLE
- d. TRANSACTION_REPEATABLE_READ Ans. D

70. dirty unread- ans)read.

71.package pkg;

```
    class output {  
        public static void main(String args[])  
        {  
            StringBuffer s1 = new StringBuffer(  
"Hello");  
  
            s1.setCharAt(1, x);  
  
            System.out.println(s1);  
        }  
    }
```

- a) xello
- b) xxxxx
- c) Hxlllo
- d) Hexlo

Answer: c

```

72. public class Test8 {
    public static void main (String args[]) {
        Number numberRef = new Integer(0);
        Double doubleRef = (Double)numberRef;
    }
}

```

It shows me exception at runtime:

```

Exception in thread "main" java.lang.ClassCastException:
    java.lang.Integer cannot be cast to java.lang.Double

```

73. command design pattern ans)d (history...)

74. Which of these is not a valid value for dispatcher element of filtermapping.
Select the one correct answer.

- A. FORWARD
- B. RESPONSE Ans) B
- C. INCLUDE
- D. ERROR

75. Which of the following is legal JSP syntax to print the value of i.
Select the one correct answer

- A. <%int i = 1;%>
 <%= i; %>
- B. <%int i = 1;
 i; %>
- C. <%int i = 1%>
 <%= i %>
- D. <%int i = 1;%>
 <%= i %>

E. `<%int i = 1%>`
`<%= i; %>`

Ans) D.

76. `<% int sum = 0;`
`for(j = 0; j < 10; j++) { %>`
`// XXX --- Add j to sum`
`<% } %>`
`// YY --- Display ths sum. What must be added at`
`the location YY to print the sum of ten numbers.`
Select the one correct statement

- A. `<% sum %>`
- B. `<% sum; %>`
- C. `<%= sum %>`
- D. `<%= sum; %>` ANS) C.

77. `<%@ taglib uri="http://java.sun.com/jsp/jstl/core"`
`prefix="c" %>`
`<c:set var="j" value="4,3,2,1"/>`
`<c:forEach items="{j}" var="item" begin="1" end="2"`
`>`
`<c:out value="{item}" default="abc"/>`
`</c:forEach>`

- A. 1
 - B. 2
 - C. 3
 - D. 4
 - E. abc
 - F. The program does not compile.
- Which numbers gets pri

Ans) 2,3

78. Which number gets printed when the following JS
TL code
fragment is executed? Select the one correct answer
s.
`<%@ taglib uri="http://java.sun.com/jsp/jstl/core"`
`prefix="c" %>`

```

<c:set var="j" value="4,3,2,1"/>
<c:forEach items="${j}" var="item" varStatus="status">
<c:if test="${status.first}">
<c:out value="${status.index}" default="abc"/>
</c:if>
</c:forEach>

```

Ans) A

- A. 1
- B. 2
- C. 3
- D. 4
- E. abc
- F. The program does not compile.

79. Which of the following is not a valid HTTP/1.1 method. Select the one correct answer.

15.

- A. CONNECT method
- B. COMPARE method
- C. OPTIONS method
- D. TRACE method

Name the http method used

answer: B

80. Name the location of compiled class files within a war file? Select the one correct answer.

19.

- A. /META-INF/classes
- B. /classes
- C. /WEB-INF/classes
- D. /root/classes

Answers- web inf

81. What get printed when the following JSTL code fragment is executed?

```

<%@ taglib uri="http://java.sun.com/jsp/jstl/core"
prefix="c" %>

```

```
<c:forEach var="item" begin="0" end="12" step="3">
    ${item}
</c:forEach>
```

options
A) 0, 3, 6, 9, 12
B) 3, 6, 9, 12
C) 0, 6, 9, 12
D) 0, 3, 6, 9

<http://www.sanfoundry.com/java-questions-answers-freshers-experienced/>

The four techniques used in session tracking are:

Java Language Keywords

Here is a list of keywords in the Java programming language. You cannot use any of the following as identifiers in your programs. The keywords `const` and `goto` are reserved, even though they are not currently used. `true`, `false`, and `null` might seem like keywords, but they are actually literals; you can not use them as identifiers in your programs.

<code>abstract</code>	<code>continue</code>	<code>for</code>	<code>new</code>	<code>switch</code>
<code>assert***</code>	<code>default</code>	<code>goto*</code>	<code>package</code>	<code>synchronized</code>
<code>boolean</code>	<code>do</code>	<code>if</code>	<code>private</code>	<code>this</code>
<code>break</code>	<code>double</code>	<code>implements</code>	<code>protected</code>	<code>throw</code>
<code>byte</code>	<code>else</code>	<code>import</code>	<code>public</code>	<code>throws</code>
<code>case</code>	<code>enum****</code>	<code>instanceof</code>	<code>return</code>	<code>transient</code>
<code>catch</code>	<code>extends</code>	<code>int</code>	<code>short</code>	<code>try</code>
<code>char</code>	<code>final</code>	<code>interface</code>	<code>static</code>	<code>void</code>

```
class      finally      long      strictfp**  volatile
const*    float    native    super    while
*          not used
**         added in 1.2
***        added in 1.4
****       added in 5.0
```