

1. How to import all package?

Answer : `import java.*`

2. Which of the following is correct about junit?

Answer: - `It is an open source framework.`
- `It provides Annotation to identify the test methods.`
- `It provides Assertions for testing Expected results`

3. Pattern which follows sequential order?

Answer : `Iterator`

4. The pattern involves a single class which is responsible to create an object while making sure that only single object gets created?

Answer: `Singleton`

5. 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";  
}  
}  
}.....
```

Animal animal = new Dog();

Cat cat= (Cat)animal;

SOP(cat.noise());

Answer : `Exception is thrown at the Runtime`

```
6. class output {  
    public static void main(String args[]){  
        String buffer s1 = new  
        StringBuffer("Hello world")  
        s1.insert(6,"Good");  
        SOP(s1);  
    }  
}
```

Answer: `Hello Goodworld`

7. SOP ('1'+new Integer (2) +3);

Answer: `123`

8. A team of programmer is involved in reviewing a proposed design for a new utility class after some discussion.....

Answer: `weak encapsulation`

9. Man dog best friend

Answer : `class man private dog best friend;`

10. Bluemix deploy cloud foundry

Answer: `cf push`

11. Class vector {

```
PSVM(String args[])  
Vector obj = new vector(4,2);  
Obj.add element(new Integer(3));  
Obj.add element(new Integer(2));  
Obj.add element(new Integer(5));  
Obj.removeAll(obj);  
SOP(obj.isEmpty());  
}
```

Answer : `True`

12. `<%@ taglib%>`

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

`<c:foreach item = "${j}" var = "item"
varstatus = "status">`

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

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

`</c:if>`

`</c:foreach>`

Answer: `1`

13. `StringBuffer s1 = new
StringBuffer("Hello");
StringBuffer s2 = reverse(s1);
SOP(s2);`
Answer : olleH
14. Chain...cf..which filter user?
Answer: `cf.doFilter()`
15. How to access the resultsetobject?
Answer : by using special getter method.....
16. X= something
While(X<10&&x>24)
Answer : condition false
17. `RunWith(Suite.class)
@SuiteClasses({
annotationTest.class,
AssertTest.class
CalculateTest.class,
EmployeeTest.class})
Public class SuiteTest{
//the class remains empty
//used only as a holder for the above
annotaiton
}
What is the sequence of junit test ?
Answer : AnnotationTest, AsserTest,
CalculateTest,EmployeeTest`
18. `Public static void main(String args[])
{
List<string> List = new
ArrayList<string>();
//add string
List.add("cricket");
List.add("football");
List.add("hockey");
Iterator it = List.iterator();
While(it.hasNext())
{
String s = it.next();
}
}
Which pattern
Answer : Iterator`
19. Java.util.enumeration is an example of which pattern?
Answer : Iterator
20. Aggregation represent in UML?
Answer : hollow diamond on the collection side
21. Composition represent in UML?
Answer : Filled diamond on the side of the collection
22. Generalization represent in UML?
Answer : hollow triangle shape on the superclass end of the line
23. Dependency represent in UML?
Answer : dotted line followed by
24. 1..* specify about relationship?
Answer : At least one instance
25. 0..1 Specify about relationship?
Answer: Zero or one instance. The notation n..m indicate n to m instances.
26. The relationship between the object and component parts in UML diagram are represented by -----.
Answer : Aggregation
27. Class x implements Runnable {
Public static void main(String args[])
{
/*missing code? */
}
Public void run() {
}
}
Which of the following line of code to start thread?
Answer : `x run = new x();
Thread t = new Thread(run);
t.start();`
28. `PSVM()
{
try{
int a = 5;
int b = 0;
int c = a/b;
SOP("World");
}
Catch(exception e)`

```
{
SOP("hello");
}
```

Answer : **hello**

29. Int a = 10;

Int b = 0;

Int c = a/b;

SOP("c");

Output :

Answer: **runtime exception not
arithmetic exception**

30. Which will contain the body of the thread?

Answer : **run()**

31. Which two of the following methods are defined in class thread?

1. Start() 2. Wait() 3. Notify() 4. Run()

Answer : **start() and run()**

32. <%@ %>

<c:foreach var = "item" begin ="0" end ="12" step = "3">

Answer: **0,3,6,9,12**

33. A user types on URL which method is called?

Answer : **get Method**

34. The method getWriter return an object of type PrintWriter?

Answer: **HttpServletResponse**

35. To send binary o/p?

Answer : **getOutputStream**

36. Unit testing?

Answer : **Single Entity**

37. Assume the following method is properly synchronized and called from a thread A on an object B:
wait(2000);

After calling this method, when will the thread A become a candidate to get another turn at the CPU?

A. **After thread A is notified, or after two seconds.**

B. After the lock on B is released, or after two seconds.

C. Two seconds after thread A is notified.

D. Two seconds after lock B is released.

38. Invoked only once?

Answer : **@BeforeClass**

39. Invoked before each test?

Answer : **@Before**

40. Invoked after each test?

Answer : **@After**

41. Only once invoked finishing all tests?

Answer : **@AfterClass**

42. Which of the following represents the XML equivalent of this statement

<%@ include file="a.jsp"%> . Select the one correct statement

a. <jsp:include file="a.jsp"/>

b. <jsp:include page="a.jsp"/>

c. **<jsp:directive.include
file="a.jsp"/>**

There is no XML equivalent of include directive.

43. What will be proper order of session?

Answer : **loading, instantiation, init,
service, destroy**

44. Proper order of access modifier

Answer : **private default protected
public**

45. If the options are iterator servlet etc answer will be **iterator**.

46. If the options will be like

New class foo{print(.....)},

Answer: **new foo(){print(.....)}**

47. Insert(6, "Good")

Answer : Hello Good Word

48. Class{

PSVM

String str = new String(".....");

}

Do{

str = **"Hello Stop World "**;

SOP(str);

```

}
While(str!=Strong);
{
.....
}

```

Answer : **Hello Stop World**

49. When we use command design pattern

Answer. **when history of requests is needed**

50. In use case diagram actor represents?

Answer: **An actor represent a roll that an outsider takes on when intracting with the business system for instance, an action can be a customer, a business partner , a supplier or another business system**

51. In use case diagram actor represents ____?

Answer: An actor represents a role that an outsider takes on when interacting with the business system.

For instance, an actor can be a customer, a business partner, a supplier, or another business system.

52. Common mechanisms for session tracking

Ans.

User authorization

Hidden fields

URL rewriting

Cookies

Session tracking API

53. The code snippet below is an example of which of the following?

Long myLong = 21L;

A Autoboxing

B Autounboxing

C Autocasting

D Autoinstanting

Answer: Autoboxing

54. which listener is used to log the...

Answer: ServletRequestListener

55. when you want to know attribute has added removed or replaced?

Answer: HttpSessionAttributeListener

56. when you want to know how many users are there?

Answer: HttpSessionListener

57. listener is noticed initialized and destroyed?

Answer: ServletContextListener

58. clear(2) in the string of index 2

Answer : 0134

59. Authentication mechanishm in deployment descriptor?

Answer: login-config.

60.

1. interface TestA { String toString(); }

2. public class Test {

3. public static void main(String[] args) {

4. System.out.println(new TestA() {

5. public String toString() { return "test";

}

6. });

7. }

8. }

What is the result?

A. test

B. null

C. An exception is thrown at runtime.

D. Compilation fails because of an error in line 1.

E. Compilation fails because of an error in line 4.

F. Compilation fails because of an error in line 5.

Answer: A

61. 2+3*5

Answer **17 integer**

17 byte

25 byte

25 integer

62. Identify the correct signature of the main method?

Answer : public static void main(String args[])

```

63. public static void main(String[] args)
{
    GenericTest<String> arr[] = new
    GenericTest[5];           //line 1
    arr[0] = new GenericTest("Java");
                               //line 2
    arr[1] = new GenericTest(1);
                               //line 3
    arr[2] = (GenericTest<String>)new
    GenericTest(1);           //line 4
    arr[3] = (GenericTest<String>)new
    GenericTest<Integer>(1);    //line 5
    for(GenericTest o:arr)
    {
        System.out.println(o);
    }
}

```

Output: Error in Line 1

64. <c:forToken item = "first, second,
third, last;" delims = "," var =
"current">
<c:out value = "\${current}" />=?
Answer: beforelast

65. Answer: AaBaaaCaaaa

66. Multiple exception catch block..
Answer : superclass exception can not
be caught first
Can we caught either super class or
subclass

67. Filter life cycle
Answer: init() dofilter() destroy()

68. /\${3+2-1}:\${3+2-1}
Answer: a. /\${3+2-1}:4
b. 4:4
c. /4:4

69. l.add(2)
l.add(5)
l.add(8)
l.add(1)
l.reverse();
Answer: 1852

70. getvalue(IDEAL)
SOP(word.substring(0,word.length()-
1)+word.charAt(word.length()-1))
Answer: DEAL

71. in javabean reference id define?
a. Classname
b. Package
c. Instance
d. None

72. Int x=6;
Int y = 12;
If(x>y)
{
X++;
y--;
}
SOP(....);
Answer: 2

73. Datatype of the no 9.6352
Answer: float
double
Float
Double

74. Answer: RS.CONCUR_UPDATABLE

75. How many thread one from exception
mainclass

Answer : is only 1 thread(main thread)

76. Start same thread two times

t1.start()
t1.start()

Answer: compilation fails, compile time
error

77. Behavior of UML diagram?
Answer: a. usecase
b. object
c. profile

78. Decouples in which design pattern?
Answer: Bridge

79. String s1 = abc;
String s2 = def;
String s3 = s1.concat(s2.UpperCase)
Answer:: abcdefabcDEF

80. HttpServlet supports

- a. http
- b. https
- c. both
- d. none

81. what is the output of this program

```
class output{
    public static void main(String args[])
    {
        Object obj = new object();
        System.out.print(obj.getClass());
    }
}
```

Answer: class java.lang.Object

82. declare array of string

```
string[]s;
string s[]
string []s;b
```

Answer: ABD

83. class A{

```
int i;
int j;
A(){
    I =1;
    J=2;
}
}
```

```
Class output{
    Public static void main(String args[])
    {
        A obj1 = new A();
        SOP(obj1.toString());
    }
}
```

Answer: String associated with obj1

And output will be A@1cde5f

84. public void divide(int a, int)

```
{
    Try{
        Int c = a/b;
    }
    Catch(Exception e)
    {
        SOP(Exception);
    }
    Finally{
```

```
SOP("finally")
}
}
```

Output :

Answer if a,b not initialized then error and if b = 0; then compile time error with finally will work.

85. Current thread

```
Public class MyRunnable implements
runnable {
    Public void run()
    {
        //some code
    }
}
```

Which of these will create and start this thread?

Answer:

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

86. When we use command design pattern?

Answer: when history of request is needed

87. Xyz-xyz instance (pattern)

Answer: Singleton

Abstract

Factory

88. To execute in the parameterized statement?

Answer: prepared statement

89. Design pattern to limit class instantiated to one object?

Answer: Singleton

90. Given

```
10 interface Foo{int bar();}
11 public class sprite{
12     public int fubar(Foo foo){return
foo.bar();}
13 public void testFoo(){
14     fubar(
15 //insert code here
16 );
17 }}
```

Answer: new Foo(){public int bar(){return 1;}}

91. A: unit testing is the testing of single entity

B: unit testing is the process of checking the functionality of the application whether it is working as per requirements

Options :

a. A is true

b. B is true

c. Both are true

d. None of the above

What will be the output of this -

92. String river = new String("TRAINING");
System.out.println(river.length());

answer-8

93. String getValue(String word)
{
if (word.length() == 1)
return "";
else
return getValue(word.substring(0,
word.length() - 1)) +
word.charAt(word.length() - 1);
}

What is the value of the string returned by getValue("DEMOS")

Answer: EMOS

94. public class Question {
public static void main(String args[]) {
String s1 = "uvw";
String s2 = "xyz";
String s3 = s1.concat(s2.toUpperCase()
);
System.out.println(s1+s2+s3);
}}
answer: uvwxyzuvwXYZ

95. int a = 9;
int b = 14;
while(a<b) {

System.out.println("In the loop");

a+=2;
b-=2;

}

answer- In the loop

In the loop

96. abstract class xyz

{

abstract abc (int a, int b) { }

}

- What sort of compilation error you will get ?

- How to fix the compilation error

Answer-- give return type void. and remove empty parenthesis.

97. interface ATOB {

String s = "yo";

public void meth();

}

interface BTOC { }

interface C extends ATOB, BTOC {

public void meth();

public void meth(int x);

}

- Will this compile ?

answer--- interface can also be extended (when inheriting from two interfaces.)

98. int i = -1;

int b = 10;

int val = b/ i;

answer- -10

interface Data { public void load(); }

abstract class Info { public abstract void load(); }

99. How Can i inherit both the interface and abstract class ?

answer= class xyz extends Info implements interface{ void load}

100. public class GenericTest<G> {

G g;

GenericTest(G g){

```

        this.g =g;
    }
    public static void main(String[] args)
    {
        GenericTest<String> arr[] = new
GenericTest[5];           //line 1
        arr[0] = new GenericTest("Java");
                        //line 2
        arr[1] = new GenericTest(1);
                        //line 3
        arr[2] = (GenericTest<String>)new
GenericTest(1);           //line 4
        arr[3] = (GenericTest<String>)new
GenericTest<Integer>(1);   //line 5
        for(GenericTest o:arr)
        {
            System.out.println(o);
        }
    }
}

```

Answer-- can not cast <Integer> to <String>
line5

and default value is null in an array

```

101.    class A {
        static    String    str    =
"STRING_IN_A";
    }
    class B extends A {
        static String str = "STRING_IN_B";
    }
}

```

//

in void main

B b1=new B();

sop b1.str

//

Answer--- will compile fine.bt str is being
overwritten in B class.

```

102.    int j=8;
while( (j < 10) && (j > 24)) {
    System.out.println(" Am here");
}

```

answer-while() condition is always false

```

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

```

answer-helloworld

104. are resultset updateable?

Answer: yes but only if you indicate a
concurrency strategy when executing the
statement and if the driver and database
support this option.

105. How can you start a database
transaction in the database

Answer: by setting the autocommit
property of the connection to false &
execute a statement in the database

106. The object that does not
contain the stored procedure itself
but contains only a call to the store
procedure?

Answer: callableStatement

107. To execute a stored procedure
"total stack" in a database server
which of the following code snippet is
used?

Answer: callablestatement clbstmnt =
con.preparecall("{call totoal stack}");
cs.executeQuery();

108. Jsp pages have access to
implicit objects that are exposed
automatically one such object that is
available is request that request
object is an instance of which class?

Answer: HttpServletRequest

109. Relative path

Answer: c:/test/***

110. In class diagram instance are
represented by using

Answer: Italics

111. Which type of driver converts jdbc calls into the Network protocol used by the datamanagement system?

Answer: Type 4

112. UML behavior diagram?

Answer: Attribute diagram

113. `SOP(index(i)+index(0)+lastIndex(i)+lastIndex(0));`

Answer: 6469

114. `<c:set var = "j" value = "4,3,2,1"/>`
`<c:forEach items = "${j}" var = "item" begin="1" end="2"`

Answer: 2 and 3

115. The `sendError` method defined in the `HttpServlet` class is equivalent to invoking the `setStatus` method with the following parameter. Select the one correct answer:

Answer: `SC_NOT_FOUND`

116. When a `select` statement returns only one record with a single column of integer datatype on invoking `resultSet.getInt(0)` causes _____

Answer:

117. A user types the url <http://www.javaprep.com/scwd/index.html> which http request gets generated select the one correct answer?

Answer: GET Method

118. Which listener class will `close()` method?

Answer: `WindowListener`

119.

```
Class exception_Handling{
    Public static void main(String args[]){
        Try{
            SOP("Hello"+" "+1/0);
        }
        Catch(ArithmeticException e)
        {
```

```
SOP("World");
}
}}
```

Answer: World

120. What get printed when the following JSTL code program 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>
```

Answer: 0,3,6,9,12

121. What is the output of this program?

```
Import java.util.*
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));
    SOP(obj.entrySet());
}
Answer: [A=1,B=2,C=3]
Public class genericstack<E>{
    Stack<E>{
        Stack<E>stk = new stack<E>();
    }
}
```

122. Which of the following describes the purpose of JUnit?

A It is a framework to help with writing unit tests for your code.

B It is a framework to help with writing code involving unit conversions.

C It is a framework to help with packaging your code into units for distribution.

D It is a framework to convert all days/dates in your code to June.

Answer: A

123. Which diagram in UML shows a complete or partial view of the structure of a modeled system at a specific time?

- a) Sequence Diagram
- b) Collaboration Diagram
- c) Class Diagram
- d) Object Diagram

Answer: d

124. Which of these keywords must be used to monitor for exceptions?

- a) try
- b) finally
- c) throw
- d) catch

View Answer

Answer: a

125. Treemap question and options will be like

(7,9)

(9,7)

Answer : 7,9

126. Testing Amit Gupta and Ravi Kumar wala question?

- a) Amit Gupta False
- b) Ravi Kumar False
- c) Amit Gupta False(Expected name as Ravi Kumar)
- d) Ravi Kumar False(Expected name as Amit Gupta)

use

127. Fill in the blank

```
Exception _____{  
    printstack e;  
}  
Exception _____{  
    printStack e;  
}
```

Answer class not found

Sql exception

128. Get value from resultset

Answer: getBoolean(),getClob()
getBlob()

```
129. class exception_handling {  
    public static void main(String args[]) {  
        try {  
            int a, b;  
            b = 0;  
            a = 5 / b;  
            System.out.print("A");  
        }  
        catch(ArithmeticException e) {  
            System.out.print("B");  
        }  
    }  
}
```

- a) A
- b) B
- c) Compilation Error
- d) Runtime Error

Ans:B

```
130. class A implements foo {  
    16 foo bar()  
    17 {  
    18 Return 1;  
    }  
    }  
    Class A implements foo  
    {  
    25 foobar()  
    26 {  
    27 return 2;  
    }  
    }
```

Answer: return 2;

If eliminate 16,17,18 return 2

If eliminate 25 26 27 return 1

131. Which of these handles the exception when no catch is used?

- a. Default handler**
- b. Finally
- c. Throw handler
- d. Java run time system

132. class exception_handling {
 public static void main(String args[]) {
 try {
 int a, b;
 b = 0;
 a = 5 / b;
 System.out.print("A");
 }
 catch(ArithmeticException e) {
 System.out.print("B");
 }
 finally {
 System.out.print("C");
 }
 }
 }

- a) A
- b) B
- c) AC
- d) BC

Answer:BC

133. public class Delta
 { static boolean foo(char c)
 {
 System.out.print(c);
 return true;
 }
 public static void main(String[] argv
)
 {
 int i = 0;
 for (foo('A'); foo('B') && (i < 2);
 foo('C'))
 {
 i++;
 foo('D');
 }
 }
 }

Answer: A.
 ABDCBDCB

134. Why beans are used in J2EE
 architecture in stead of writing all the
 code in JSPs ?

Answer: (a) Allows separation of roles
 between web developers and
 application developers

135. public class TreeSet {
 public static void main(String args[]){
 //Creating and adding
 elements
 TreeSet<Integer>
 al=new TreeSet<Integer>();
 al.add("7"); //line 5
 al.add(9);
 //Traversing elements
 Iterator<Integer>
 itr=al.iterator();
 while(itr.hasNext()){
 System.out.println(itr.next());
 }
 }
 }

Answer: compilation error at line 5

136. Which implementation of set
 would you choose if you want the
 iterator of set would give you object
 in the order it were inserted?

LinkedHashSet

TreeSet
 HashSet

137. Push(7);
 Push(2);
 Pop();
 Push(9);

Answer: print[7,9]

138. Which operator use to separate
 url and attribute?

Answer : ?

139. Which operator is used to
 separate parameters or attributes?

Answer: &

140. Which method is used to url rewriting?

Answer doGet method

141. Delay show in usecase by "Slant line"

Answer Slant line for delay

142. (first,two,three,four;,five;) delimits(,)

Answer: firsttwothreefour;five;

143. Which packages contain the JDBC classes?

(a) java.jdbc and javax.jdbc

(b) java.jdbc and java.jdbc.sql

(c) java.sql and javax.sql

(d) java.rdb and javax.rdb

144. Which type of driver provides JDBC access via one or more ODBC drivers?

(a) Type 1 driver

(b) Type 2 driver

(c) Type 3 driver

(d) Type 4 driver

145. Which type of driver converts JDBC calls into the network protocol used by the database management system directly?

(a) Type 1 driver

(b) Type 2 driver

(c) Type 3 driver

(d) Type 4 driver

146. How can you retrieve information from a ResultSet?

(a) By invoking the method get(..., String type) on the ResultSet, where type is the database type

(b) By invoking the method get(..., Type type) on the ResultSet, where Type is an object which represents a database type

(c) By invoking the method getValue(...), and cast the result to the desired Java type.

(d) By invoking the special getter methods on the ResultSet: getString(...), getBoolean (...), getClob(...),...

147. How can you execute DML statements (i.e. insert, delete, update) in the database?

(a) By making use of the InsertStatement, DeleteStatement or UpdateStatement classes

(b) By invoking the execute(...) or executeUpdate(...) method of a normal Statement object or a sub-interface object thereof

(c) By invoking the executeInsert(...), executeDelete(...) or executeUpdate(...) methods of the DataModificationStatement object

(d) By making use of the execute(...) statement of the DataModificationStatement object

148. How do you know in your Java program that a SQL warning is generated as a result of executing a SQL statement in the database?

(a) You must catch the checked SQLException which is thrown by the method which executes the statement

(b) You must catch the unchecked SQLWarningException which is thrown by the method which executes the statement

(c) You must invoke the getWarnings() method on the Statement object (or a sub interface thereof)

(d) You must query the ResultSet object about possible warnings generated by the database

149. What is, in terms of JDBC, a DataSource?

(a) A DataSource is the basic service for managing a set of JDBC drivers

(b) A DataSource is the Java representation of a physical data source

(c) A DataSource is a registry point for JNDI-services

(d) A DataSource is a factory of connections to a physical data source

150. What is the meaning of ResultSet.TYPE_SCROLL_INSENSITIVE

(a) This means that the ResultSet is insensitive to scrolling

(b) This means that the Resultset is sensitive to scrolling, but insensitive to updates, i.e. not updateable

(c) This means that the ResultSet is sensitive to scrolling, but insensitive to changes made by others

(d) The meaning depends on the type of data source, and the type and version of the driver you use with this data source

151. Are ResultSets updateable?

(a) Yes, but only if you call the method openCursor() on the ResultSet, and if the driver and database support this option

(b) Yes, but only if you indicate a concurrency strategy when executing the statement, and if the driver and database support this option

(c) Yes, but only if the ResultSet is an object of class UpdateableResultSet, and if the driver and database support this option

(d) No, ResultSets are never updateable. You must explicitly execute DML statements (i.e. insert, delete and update) to change the data in the underlying database

152. How can you start a database transaction in the database?

(a) By asking a Transaction object to your Connection, and calling the method begin() on it

(b) By asking a Transaction object to your Connection, and setting the autoCommit property of the Transaction to false

(c) By calling the method beginTransaction() on the Connection object

(d) By setting the autoCommit property of the Connection to false, and execute a statement in the database

153. What is the meaning of the transaction isolation level TRANSACTION_REPEATABLE_READ

(a) Dirty reads, non-repeatable reads and phantom reads can occur

(b) Dirty reads are prevented; non-repeatable reads and phantom reads can occur

(c) Dirty reads and non-repeatable reads are prevented; phantom reads can occur

(d) Dirty reads, non-repeatable reads and phantom reads are prevented

154. What statements are correct about positioned updates (i.e. cursor updates) in ResultSets? (2 correct answers)

[a] Using the cursor technique is currently the only possible way to change the data in the current row of a ResultSet

[b] Insert statements are only supported when using scrollable cursors.

[c] Only scrollable updateable ResultSets can use this approach to change the data in the current row of a ResultSet

[d] The name of the cursor is specified by the setCursorName(String name) method of the Statement object.

155. How can you execute a stored procedure in the database?

- (a) Call method `execute()` on a `CallableStatement` object
- (b) Call method `executeProcedure()` on a `Statement` object
- (c) Call method `execute()` on a `StoredProcedure` object
- (d) Call method `run()` on a `ProcedureCommand` object

156. What happens if you call the method `close()` on a `ResultSet` object?

- (a) the method `close()` does not exist for a `ResultSet`. Only `Connections` can be closed.
- (b) the database and JDBC resources are released
- (c) you will get a `SQLException`, because only `Statement` objects can close `ResultSets`
- (d) the `ResultSet`, together with the `Statement` which created it and the `Connection` from which the `Statement` was retrieved, will be closed and release all database and JDBC resources

157. What happens if you call `deleteRow()` on a `ResultSet` object?

- (a) The row you are positioned on is deleted from the `ResultSet`, but not from the database.
- (b) The row you are positioned on is deleted from the `ResultSet` and from the database
- (c) The result depends on whether the property `synchronizeWithDataSource` is set to true or false
- (d) You will get a compile error: the method does not exist because you can not delete rows from a `ResultSet`

158. What statements are correct about batched insert and updates? (2 answers)

[a] To create a batch of insert and update statements, you create an object of type `Batch`, and call the method `addStatement(String statement)` for each statement you want to execute in the batch

[b] Batch insert and updates are only possible when making use of parameterized queries.

[c] To do a batched update/insert, you call `addBatch(String statement)` on a `Statement` object for each statement you want to execute in the batch

[d] To execute a batched update/insert, you call the `executeBatch()` method on a `Statement` object

159. What is correct about DDL statements (`create`, `grant`,...)?

(a) DDL statements are treated as normal SQL statements, and are executed by calling the `execute()` method on a `Statement` (or a sub interface thereof) object

(b) To execute DDL statements, you have to install additional support files

(c) DDL statements can not be executed by making use of JDBC, you should use the native database tools for this.

(d) Support for DDL statements will be a feature of a future release of JDBC

160. The JDBC-ODBC Bridge supports multiple concurrent open statements per connection?

- a. True
- b. False

162. 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`

163. Which of the following statements is false as far as different type of statements is concern in JDBC?

- a. Regular Statement
- b. Prepared Statement
- c. Callable Statement
- d. Interim Statement

164. Which of the following methods are needed for loading a database driver in JDBC?

- a. registerDriver() method
- b. Class.forName()
- c. Both A and B
- d. getConnection()

165. Which of the following is false as far as type 4 driver is concern?

- a. Type 4 driver is "native protocol, pure java" driver
- b. Type 4 drivers are 100% Java compatible
- c. Type 4 drivers uses Socket class to connect to the database
- d. Type 4 drivers can not be used with Netscape

166. To execute a stored procedure "totalStock" in a database server, which of the following code snippet is used?

- a. `Statement stmt = connection.createStatement(); stmt.execute("totalStock()");`
- b. `CallableStatement clbstmnt = con.prepareCall("{call totalStock}"); cs.executeQuery();`
- c. `StoreProcedureStatement stmt=connection.createStoreProcedure("totalStock()"); spstmt.executeQuery();`
- d. `PreparedStatement pstmt = connection.prepareStatement("totalStock()"); pstmt.execute();`

167. Which driver is efficient and always preferable for using JDBC applications?

- a. Type – 4
- b. Type – 1
- c. Type – 3
- d. Type – 2

168. JDBC facilitates to store the java objects by using which of the methods of PreparedStatement

- setObject () 2. setBlob() 3. setClob()
- a. 1, 2
- b. 1,2,3
- c. 1,3
- d. 2,3

169. Which statement is static and synchronized in JDBC API?

- a. executeQuery()
- b. executeUpdate()
- c. getConnection()
- d. prepareCall()

170. The JDBC-ODBC bridge is

- a. Three tiered
- b. Multithreaded
- c. Best for any platform
- d. All of the above

171. All raw data types (including binary documents or images) should be read and uploaded to the database as an array of

- a. byte
- b. int
- c. boolean
- d. char

172. The class java.sql.Timestamp has its super class as

- a. java.sql.Time
- b. java.util.Date
- c. java.util.Time
- d. None of the above

173. BLOB, CLOB, ARRAY and REF type columns can be updated in

- a. JDBC 1.0
- b. JDBC 4.0
- c. JDBC 2.0
- d. JDBC 3.0

174. Which of the following methods finds the maximum number of connections that a specific driver can obtain?

- a. Database.getMaxConnections
- b. Connection.getMaxConnections
- c. DatabaseMetaData.getMaxConnections
- d. ResultSetMetaData.getMaxConnections

175. Are prepared statements actually compiled?

- a. Yes, they compiled
- b. No, they are bound by the JDBC driver

176. When the message "No Suitable Driver" occurs?

- a. When the driver is not registered by Class.forName() method
- b. When the user name, password and the database does not match
- c. When the JDBC database URL passed is not constructed properly
- d. When the type 4 driver is used

177. Which driver is called as thin-driver in JDBC?

- a. Type-4 driver
- b. Type-1 driver
- c. Type-3 driver
- d. Type-2 driver

178. How many transaction isolation levels are defined in java.sql.Connection interface?

- a. 4
- b. 3

- c. 5
- d. 2

179. Which method is used to perform DML statements in JDBC?

- a. execute()
- b. executeQuery()
- c. executeUpdate()
- d. executeResult()

180. What is the disadvantage of Type-4 Native-Protocol Driver?

- a. At client side, a separate driver is needed for each database.
- b. Type-4 driver is entirely written in Java
- c. The driver converts JDBC calls into vendor-specific database protocol
- d. It does not support to read MySQL data.

```
181. import java.util.*;
class TestHashMaps{
    public static void main(String args[]) {
        HashMap<Integer,String> hm= new
        HashMap<Integer,String> ();
        hm.put(100, "John");
        hm.put(101, "Paul");
        hm.put(102, "George");
        hm.put(103, "Ringo");
        for (Map.Entry m: hm.entrySet()) {
            System.out.println(m.getKey() + " " +
            m.getValue());
        }
    }
}
```

Answer:

100 John
101 Paul
102 George
103 Ringo

```
182. import java.util.Map;
import java.util.TreeMap;
public class TestTreeMap {
```



```

public static void main(String args[]) {
    TreeMap< Integer, String > hm= new
    TreeMap< Integer, String > ();
    hm.put(100, "John");
    hm.put(102, "Paul");
    hm.put(101, "George");
    hm.put(103, "Ringo");
    for (Map.Entry m: hm.entrySet()) {
        System.out.println(m.getKey() + " " +
        m.getValue());
    }
}

```

Answer

```

100 John
101 George
102 Paul
103 Ringo

```

183. Given:

```

5. import java.util.*;
6. public class SortOf {
7.     public static void main(String[] args) {
8.         ArrayList<Integer> a = new
        ArrayList<Integer>();
9.         a.add(1); a.add(5); a.add(3);
11.        Collections.sort(a);
12.        a.add(2);
13.        Collections.reverse(a);
14.        System.out.println(a);
15.    }
16. }

```

What is the result?

- A. [1, 2, 3, 5]
 - B. [2, 1, 3, 5]
 - C. [2, 5, 3, 1]
 - D. [5, 3, 2, 1]
 - E. [1, 3, 5, 2]
 - F. Compilation fails.
 - G. An exception is thrown at runtime.
- Answer: C

184. 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

185. Which Man class properly represents the relationship "Man has a best friend who is a Dog"?

- A. class Man extends Dog { }
- B. class Man implements Dog { }
- C. class Man { private BestFriend dog; }
- D. class Man { private Dog bestFriend; }
- E. class Man { private Dog<bestFriend>; }
- F. class Man { private BestFriend<dog>; }

Answer: D

186. Given:

```

11. class Alpha {
12.     public void foo() {
13.         System.out.print("Afoo "); }
14. }
15. public class Beta extends Alpha {
16.     public void foo() {
17.         System.out.print("Bfoo "); }
18. }
19. public static void main(String[] args) {
20.     Alpha a = new Beta();
21.     Beta b = (Beta)a;
22.     a.foo();
23.     b.foo();
24. }

```

What is the result?

- A. Afoo Afoo
- B. Afoo Bfoo
- C. Bfoo Afoo
- D. Bfoo Bfoo
- E. Compilation fails.

F. An exception is thrown at runtime.

Answer: D

187. Given:

5. import java.util.Date;

6. import java.text.DateFormat;

21. DateFormat df;

22. Date date = new Date();

23. // insert code here

24. String s = df.format(date);

Which code fragment, inserted at line 23, allows the code to compile?

A. df = new DateFormat();

B. df = Date.getFormat();

C. df = date.getFormat();

D. df = DateFormat.getFormat();

E. df = DateFormat.getInstance();

Answer: E

188. Given:

1. public class Base {

2. public static final String FOO = "foo";

3. public static void main(String[] args) {

4. Base b = new Base();

5. Sub s = new Sub();

6. System.out.print(Base.FOO);

7. System.out.print(Sub.FOO);

8. System.out.print(b.FOO);

9. System.out.print(s.FOO);

10. System.out.print(((Base)s).FOO);

11. } }

12. class Sub extends Base {public static final String FOO="bar";}

What is the result?

A. foofoofoofoofoo

B. foobarfoobarbar

C. foobarfoofoofoo

D. foobarfoobarfoo

E. barbarbarbarbar

F. foofoofoobarbar

G. foofoofoobarfoo

Answer: D

189 A company has a business application that provides its users with many different reports:

receivables reports, payables reports,

revenue projects, and so on. The

company has just

purchased some new, state-of-the-art,

wireless printers, and a programmer has

been assigned the

task of enhancing all of the reports to use

not only the company's old printers, but

the new

wireless printers as well. When the

programmer starts looking into the

application, the programmer

discovers that because of the design of

the application, it is necessary to make

changes to each

report to support the new printers. Which

two design concepts most likely explain

this situation?

(Choose two.)

A. Inheritance

B. Low cohesion

C. Tight coupling

D. High cohesion

E. Loose coupling

F. Object immutability

Answer: B,C

190.A team of programmers is reviewing

a proposed API for a new utility class.

After some discussion,

they realize that they can reduce the

number of methods in the API without

losing any

functionality. If they implement the new

design, which two OO principles will they

be promoting?

A. Looser coupling

B. Tighter coupling

C. Lower cohesion

D. Higher cohesion

E. Weaker encapsulation

F. Stronger encapsulation

Answer: A

191. 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

192. Given:

```
1. public class TestString1 {  
2. public static void main(String[] args) {  
3. String str = "420";  
4. str += 42;  
5. System.out.print(str);  
6. }  
7. }
```

What is the output?

- A. 42
- B. 420
- C. 462
- D. 42042
- E. Compilation fails.
- F. An exception is thrown at runtime.

Answer: D

193. Given

```
11. public interface Status {  
12. /* insert code here */ int MY_VALUE =  
13. } Which three are valid on line  
12?  
(Choose three.)
```

A. final

B. static

C. native

D. public

E. private

F. abstract

G. protected

Answer: A,B,D

194. Given:

```
10. interface Foo { int bar(); }  
11. public class Sprite {  
12. public int fubar( Foo foo ) { return  
foo.bar(); }  
13. public void testFoo() {  
14. fubar(  
15. // insert code here  
16. );  
17. }  
18. }
```

Which code, inserted at line 15, allows the class Sprite to compile?

- A. Foo { public int bar() { return 1; } }
- B. new Foo { public int bar() { return 1; } }
- C. new Foo() { public int bar() { return 1; } }
- D. new class Foo { public int bar() { return 1; } }

Answer: C

195. Given:

```
11. class Animal { public String noise() {  
return "peep"; } }  
12. class Dog extends Animal {  
13. public String noise() { return "bark"; }  
14. }  
15. class Cat extends Animal {  
16. public String noise() { return "meow";  
}  
17. } ...
```

```
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

196. A programmer has an algorithm that requires a `java.util.List` that provides an efficient implementation of `add(0, object)`, but does NOT need to support quick random access. What supports these requirements?

- A. `java.util.Queue`
- B. `java.util.ArrayList`
- C. `java.util.LinearList`
- D. `java.util.LinkedList`

Answer: D

197. Which `Man` class properly represents the relationship "Man has a best friend who is a Dog"?

- A. `class Man extends Dog { }`
- B. `class Man implements Dog { }`
- C. `class Man { private BestFriend dog; }`
- D. `class Man { private Dog bestFriend; }`
- E. `class Man { private Dog<bestFriend>; }`
- F. `class Man { private BestFriend<dog>; }`

Answer: D

199. A team of programmers is reviewing a proposed API for a new utility class. After some discussion, they realize that they can reduce the number of methods in the API without losing any functionality. If they implement the new design, which two OO principles will they be promoting?

- A. Looser coupling
- B. Tighter coupling
- C. Lower cohesion
- D. Higher cohesion
- E. Weaker encapsulation
- F. Stronger encapsulation

Answer: A

200. Given:

```
10. interface Jumper { public void jump();  
    } ...  
20. class Animal { } ...  
30. class Dog extends Animal {  
31. Tail tail; 32. } ...  
40. class Beagle extends Dog implements  
    Jumper{  
41. public void jump() {}  
42. } ...  
50. class Cat implements Jumper{  
51. public void jump() {}  
52. }
```

Which three are true?(Choose Three)

- A. Cat is-a Animal
- B. Cat is-a Jumper
- C. Dog is-a Animal
- D. Dog is-a Jumper
- E. Cat has-a Animal
- F. Beagle has-a Tail
- G. Beagle has-a Jumper

Answer: B,C,F

201. Given:

```
10. interface Data { public void load(); }  
11. abstract class Info { public abstract  
    void load(); }
```

Which class correctly uses the `Data` interface and `Info` class?

- A. `public class Employee extends Info implements Data {
 public void load() { /*do something*/ }
}`
- B. `public class Employee implements Info extends Data {
 public void load() { /*do something*/ }
}`
- C. `public class Employee extends Info implements Data {
 public void load(){ /*do something*/ }
 public void Info.load(){ /*do something*/ }
}`

```

}
D. public class Employee implements Info
extends Data {
public void Data.load(){ /*do something*/
}
public void load(){ /*do something*/ }
}
E. public class Employee implements Info
extends Data {
public void load(){ /*do something*/ }
public void Info.load(){ /*do something*/
}
}
F. public class Employee extends Info
implements Data{
public void Data.load() { /*do
something*/ }
public void Info.load() { /*do something*/
}
}

```

Answer: A

202. Given:

```

1. class TestA {
2.     public void start() {
System.out.println("TestA"); }
3. }
4. public class TestB extends TestA {
5.     public void start() {
System.out.println("TestB"); }
6. public static void main(String[] args) {
7.     ((TestA)new TestB()).start();
8. }
9. }

```

What is the result?

- A. TestA
- B. TestB
- C. Compilation fails.
- D. An exception is thrown at runtime.

Answer: B

```

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

```

```

File obj = new File("/java/system");
System.out.print(obj.getName());
}
}

```

- a) java
- b) system
- c) java/system
- d) /java/system

Answer:B

```

204. import java.io.*;
class files {
public static void main(String args[]) {
File obj = new File("/java/system");
System.out.print(obj.getAbsolutePath());
}
}

```

Note: file is made in c drive.

- a) java
- b) system
- c) java/system
- d) /java/system

View Answer

Answer: d

Explanation: None.

Output:

```

$ javac files.java
$ java files
\java\system

```

205. What is the output of this program?

```

import java.io.*;
class files {
public static void main(String args[]) {
File obj = new File("/java/system");
System.out.
}
}

```

Note: file is made in c drive.

- a) java
- b) system
- c) java/system
- d) /java/system

Answer:D

```

206. class exception_handling {
public static void main(String args[]) {
try {
int a = args.length;
int b = 10 / a;
System.out.print(a);
try {
if (a == 1)
a = a / a - a;
if (a == 2) {
int c = {1};
c[8] = 9;
}
}
catch (ArrayIndexOutOfBoundsException
e) {
System.out.println("TypeA");
}
catch (ArithmeticException e) {
System.out.println("TypeB");
}}}}
a) TypeA
b) TypeB
c) 0TypeA

```

Important link:

<http://javabysri.blogspot.in/2012/06/jdbc-objective-questions.html>