1. In OOP, Reusability of code is a feature of ………………
   1. Polymorphism
   2. Abstract class
   3. Static data members
   4. Inheritance
2. Which of the following is not OOPS concept in Java?  
   a) Inheritance  
   b) Encapsulation  
   c) Polymorphism  
   d) Compilation
3. An object is an instance of a class and it demonstrates the attributes and …….. of the class.
   1. Object
   2. Behavior
   3. Data
   4. None of these
4. A class that inherits from another class is called...

1. superclass
2. subclass
3. instance class
4. none of the above
5. An object is
6. One instance of a class
7. Another word for a class
8. A class with static methods
9. None of the above
10. Using command prompt to compile and run Java programs, which command is used to execute the Java program?
11. javac
12. java
13. exec
14. execute
15. A subclass is also called as
16. inner class
17. nested class
18. derived class
19. hidden class
20. What will be the output of the following line

System.out.println(“The answer is:”+17+3);

1. The answer is:20
2. The answer is 173
3. The Answer is:17+3
4. Error
5. Which of these coding types is used for data type characters in Java?  
   a) ASCII  
   b) ISO-LATIN-1  
   c) UNICODE  
   d) None of the mentioned
6. What is the numerical range of a char data type in Java?  
   a) -128 to 127  
   b) 0 to 256  
   c) 0 to 32767  
   d) 0 to 65535
7. Which of these values can a boolean variable contain?  
   a) True & False  
   b) 0 & 1  
   c) Any integer value  
   d) true
8. Which one is a valid declaration of a boolean?  
   a) boolean b1 = 1;  
   b) boolean b2 = ‘false’;  
   c) boolean b3 = false;  
   d) boolean b4 = ‘true’
9. What will be the output of following code if value of ch is ‘A’?

switch(ch){

case ‘A’ : System.out.println(“Grade A”);

case ‘B’ : System.out.println(“Grade B”);

case ‘C’ : System.out.println(“Grade C”);

break;

case ‘D’ : System.out.println(“Grade D”);

default : System.out.println(“Grade F”);

}

1. Grade A

Grade B

1. Grade A
2. The code will give compilation error
3. Grade A

Grade B

Grade C

1. Which concept of Java is achieved by combining methods and attribute into a class?  
   a) Encapsulation  
   b) Inheritance  
   c) Polymorphism  
   d) Abstraction
2. A class relationship in which an object is exclusively owned by an aggregating object is called:

a) Aggregation  
b) Composition  
c) Encapsulation  
d) Association

1. Object Oriented Programming give importance to ……………
   1. Algorithm
   2. Procedure
   3. Data
   4. Functions
2. Encapsulation means ………………………
   1. Support for binding data and function members together.
   2. Support for function overloading
   3. Support for data abstraction
   4. None of the above
3. Inheritance is a process by which one object can inherit **private** data of another object.
   1. True
   2. False

-------------------------Day 2 --------------------------------------------------

1. Which will legally declare, construct, and initialize an array?
2. int [] myList = {"1", "2", "3"};
3. int [] myList = (5, 8, 2);
4. int myList [] [] = {4,9,7,0};
5. int myList [] = {4, 3, 7};
6. The constructor of a class must not have a return type.
7. true
8. false
9. Given the following code:

 public class Test {

…

}

Which of the following can be used to define a constructor for this class:

1. public void Test() {…}
2. public Test() {…}
3. public static Test() {…}
4. public static void Test() {…}
5. Memory deallocation in java is done by \_\_\_\_\_\_\_\_\_\_.
6. Programmer
7. Operating system
8. Garbage collector
9. None of the above
10. This method is invoked just before the garbage collector runs.
11. final()
12. finalize()
13. finish()
14. close()
15. What will be the output of the following Java program?

class mainclass {

public static void main(String args[])

{

boolean var1 = true;

boolean var2 = false;

if (var1)

System.out.println(var1);

else

System.out.println(var2);

}

}

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

1. Which of the following is a valid declaration of an object of class Box?  
   a) Box obj = new Box();  
   b) Box obj = new Box;  
   c) obj = new Box();  
   d) new Box obj;
2. Which of the following is correct:
3. String temp [] = new String {"j" "a" "z"};
4. String temp [] = { "j " " b" "c"};
5. String temp = {"a", "b", "c"};
6. String temp [] = {"a", "b", "c"};
7. Which of this method of class String is used to obtain a length of String object?
8. get()
9. sizeof()
10. lengthof()
11. length()
12. Which of these method of class String is used to extract a single character from a String object?
13. characterAtPosition()
14. charat()
15. charAt()
16. charPosition()
17. When is a constructor called?

1. Each time the constructor identifier is used in a program statement
2. During the instantiation of a new object
3. During the construction of a new class
4. At the beginning of any program execution
5. Garbage Collection of an object can be forced by calling :
6. System.gc( )
7. System.gc( obj )
8. Runtime.gc( )
9. Garbage collection cannot be forced.
10. A method named HelloWorld() is void and takes no arguments. If you have to define this method, which of the following is correct declaration
11. HelloWorld();
12. HelloWorld(void);
13. void HelloWorld();
14. void HelloWorld(void);
15. When access modifier is omitted from the definition of the member of a class.   
    The member has \_\_\_\_\_\_\_\_\_\_\_\_.
16. default access
17. public access
18. private access
19. protected access

**Day-3**

1. How many object are created in the following snippet?

double[] ann = new double[7];

double [] bob;

bob=ann;

1. 2
2. 7
3. 1
4. Compilation error
5. What will be the output of the following code snippet?

String s1 = new String( "hello" );  
String s2 = "hello";  
 if (s1==s2)

System.out.println( "Equals");  
 else

System.out.println( "Not Equal");

1. Equal
2. Not Equal
3. Compilation error
4. None of the above

When a class has the \_\_\_\_\_\_\_\_\_\_\_\_\_\_ access specification, it is visible to other classes in the same package as well as to its subclass outside the package .

a. protected

b. private

c. private-protected

d. friendly

|  |
| --- |
| How does java support run-time polymorphism?   1. Overloading 2. Overriding 3. Inheritance 4. Java does not support run-time polymorphism |
|  |

String in Java is a?  
a) class  
b) object  
c) variable  
d) character array

 Which of these keywords is used to refer to member of base class from a subclass?  
a) upper  
b) super  
c) this  
d) none of the mentioned

1. Consider the following class definition :

public class Test extends Base {  
 public Test(int j ) {   
 }  
 public Test( int j, int k ) {  
 super( j, k ) ;  
 }  
}

Which of the following forms of constructor must exist explicitly in the definition of the Base class ?

1. Base( ) { }
2. Base(int j ) { }
3. Base( int j, int K ) { }
4. Base( int j, int k, int l ) { }
5. 1
6. 1,2
7. 1,3
8. 1,2,3,4
9. What will happen when you attempt to compile and run the following code

class Base{

public void Base(){

System.out.println("Base");

}

}

public class In extends Base{

public static void main(String argv[]){

In i=new In();

}

}

1. Compile time error Base is a keyword
2. Compilation and no output at runtime
3. Output of Base
4. Runtime error Base has no valid constructor
5. Which keyword is used in a subclass to call the constructor of superclass?
6. super
7. this
8. extend
9. extends
10. Given:

Integer a = new Integer(2);

Integer b = new Integer(2);

What happens when you do if (a==b)?

1. true
2. false
3. compilation error
4. runtime Exception
5. Which feature of OOP is indicated by the following code?

class Student{

int marks;

};

class Graduate extends public Student{

String major;

Graduate(String major){

this.major=major;

}

};

1. Inheritance
2. Polymorphism
3. Inheritance and polymorphism
4. Encapsulation and Inheritance
5. The Scanner class is found in \_\_\_\_\_\_ package

java.lang

java.util

java.io

None of the above

1. Where in a constructor, can you place a call to a constructor defined in the super class?
2. Anywhere
3. The first statement in the constructor
4. The last statement in the constructor
5. You can't call super in a constructor
6. What is the correct ordering for the import, class and package declarations when found in a single file?
7. package, import, class
8. class, import, package
9. import, package, class
10. package, class, import
11. What is displayed when the following code is compiled and executed?

String s1 = new String("Test");

String s2 = new String("Test");

if (s1==s2)

System.out.println("Same");

if (s1.equals(s2))

System.out.println("Equals");

a. Same

Equals

b. Equals

c. Same

d. The code compiles, but nothing is displayed upon execution.

e. The code fails to compile.

1. A method name min() that needs two integer arguments is declared as
2. public void min();
3. public void min (int a, b);
4. public void min (int a, int b);
5. public int min (a, b);

20. For Strings, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ compares two object references to see whether they refer to the same instance.

1. equals()
2. equalsTo()
3. ==
4. =

**Day-4**

1. I have a function myFunc() which takes mandatory as well as variable arguments. Which of the following is the correct declaration for variable parameters to a function myFunc?
2. myFunc(int a, int b, ….)
3. myFunc(int x, int y, String …arr)
4. myFunc(String…arr, int z)
5. myFunc(String s…, String y)

10. If a class implementing an interface, does not override all the methods in that interface, it has to be declared as \_\_\_\_\_\_\_\_\_\_\_\_ class

a. final

b. static

c. abstract

d. protected

1. What is the use of final keyword in Java?
2. When a class is made final, a sublcass of it can not be created.
3. When a method is final, it can not be overridden.
4. When a variable is final, it can be assigned value only once.
5. All of the above
6. Given the code below:

final class A {

int i;

}

class B extends A {

int j;

System.out.println(j + " " + i);

}

class inheritance {

public static void main(String args[]){

B obj = new B();

obj.display();

}

}

What is the output of this program?

1. 2 2
2. 3 3
3. Runtime Error
4. Compilation Error
5. The qualifier used for declaring read-only variables is :
6. final
7. const
8. static
9. private

1. What is the correct declaration of an abstract method that is intended to be public:
2. public abstract void add();
3. public abstract void add() {}
4. public abstract add();
5. public virtual add();
6. The methods declared as ­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_ in base class must be implemented by the derived class.
7. final
8. static
9. abstract
10. private

|  |
| --- |
| 1. An interface can extend any number of interfaces, and can extend classes. 2. True 3. False |

**Day-5**

Which data structure is used to allocate primitive data types in java

1. stack
2. heap
3. linked list
4. arrays

1. Assuming a method contains code which may raise an Exception (but not a RuntimeException), what is the correct way for a method to indicate that it expects the caller to handle that exception:

1. throw Exception
2. throws Exception
3. new Exception
4. Don't need to specify anything

|  |
| --- |
| 2. Which is the super class of all the Exceptions in java?   1. Exception 2. Throwable 3. Error 4. Object |
| 3. Which type of exception is thrown by parseInt() if it gets illegal data?   1. RuntimeException 2. NumberFormatException 3. ArithmeticException 4. IOException |
| 4. Which method of Exception prints a list of methods that were called before an Exception was thrown   1. getMessage() 2. printStackTrace() 3. getStackTrace() 4. getException() |
|  |

5. What is the result of executing the following code, using the parameters 4 and 0:

public void divide(int a, int b) {

try {

int c = a / b;

} catch (Exception e) {

System.out.print("Exception ");

} finally {

System.out.println("Finally");

}

1. Prints out: Exception Finally
2. Prints out: Finally
3. Prints out: Exception
4. No output

7. Go thru the code below: What kind of exception is thrown by the code?

class DefaultDemo {

public static void main(String a[]) {

String str = null;

str.equals(“Hello”);

}

}

1. ArrayIndexOutOfBoundsException
2. IllegalAccessException
3. IOException
4. NullPointerException

8. If there are multiple catch clauses for a try block, then the most generic exception must be caught in the \_\_\_\_\_\_\_\_\_\_\_\_\_ block.

1. First
2. Second
3. Second-last
4. Last

9. The *throws* clause in a method signature can throw multiple exceptions

1. True
2. False

10. A java program can throw an exception explicitly using the \_\_\_\_\_\_\_\_\_\_\_\_\_ clause

1. throw
2. throws
3. throwing
4. hurl

11. Analyze the following code and pick up the best analysis from the ones below:

class A{

public static void main(String[] args){

method();

}

static void method() throws Exception{

try{

System.out.println("hello");

}

finally{

System.out.println("good-bye");

}

}

}

1. This code will compile and display “hello” and “good-bye”
2. This code will do everything in choice a, but Java will then halt the program and report that Exception was thrown but not handled.
3. This code will not compile.

12. What appears in the standard output if you run this program?

class A{

public static void main(String[] args){

method();

}

static void method(){

try{

System.out.println("hello");

}

finally{

System.out.println("good-bye");

}

}

}

1. “hello”
2. “good-bye”
3. “hello” followed by “good-bye”

|  |
| --- |
| **IO**   1. Which abstract class is the super class of all classes used for writing characters. Select the one correct answer. 2. Writer 3. FileWriter 4. CharWriter 5. OutputStream 6. FileOutputStream 7. The exception thrown by the read() method defined in InputStream class is 8. IOException 9. FileException 10. ReadException 11. InputException 12. The System.out is an object of type – 13. PrintStream class. 14. FileWriter class 15. DataOutputStream class 16. FileOutputStream class 17. Stream that is intended for general-purpose IO, not usually character data is called \_\_\_\_\_\_\_\_. 18. Reader 19. Byte Stream 20. Character Stream 21. None of the above |
| 1. Which of the following lines of code is valid for fetching a character from keyboard? 2. BufferedReader br=new BufferedReader(new InputStreamReader(System.in));   br.readLine();   1. BufferedReader br=new BufferedReader(new InputStreamReader(System.in));   br.readByte ();   1. BufferedReader br=new BufferedReader(new InputStreamReader(System.in));   br.readChar();   1. BufferedReader br=new BufferedReader(new InputStreamReader(System.in));   br.read(); |

**Collections:**

1. What is the name of the interface used to represent collections that maintain non-unique elements in order ?
2. Collection
3. Set
4. SortedSet
5. List
6. Which of the following statements are true?
7. At the root of the collection hierarchy is a class called Collection
8. The collection interface contains a method called enumerator
9. The interator method returns an instance of the Vector class
10. The Set interface is designed for unique elements
11. TreeMap class is used to implement which collection interface.
12. Set
13. SortedSet
14. List
15. Tree
16. SortedMap
17. Which of these are interfaces in the collection framework. Select the two correct answers.
18. HashMap
19. ArrayList
20. Collection
21. NavigableMap
22. TreeMap
23. These interfaces let us cycle through the contents of a collection \_\_\_\_\_\_\_\_\_\_ & \_\_\_\_\_\_\_\_\_\_\_.
24. The \_\_\_\_\_\_\_\_\_\_ method is used to insert key and values into a Hashtable:
25. add()
26. insert()
27. put()
28. append()
29. If you want the objects in a Treemap or a Treeset to be ordered in a different way, then you will need to implement the \_\_\_\_\_\_\_\_\_\_\_\_\_\_ method when you construct the set or map.
30. Comparable
31. Comparator
32. Sortable
33. Sorter
34. Orderable

**Database:**

1. Which class contains the transaction control methods setAutoCommit, commit, and rollback?
2. Connection
3. Statement
4. Resultset
5. All of the above
6. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ object contains an SQL statement that has been precompiled
7. PreparedStatement
8. CallableStatement
9. PrecompiledStatement
10. Statement
11. A ResultSet object maintains a cursor which is initially positioned
12. before the first row
13. on the first row
14. on the last row
15. any where in resultset
16. On creation, JDBC Connections are in an auto-commit mode
17. True
18. False
19. After getting a Connection object, the developer can set auto-commit to false with the
20. SetAutoCommit() method
21. SetAutoRollback() method
22. AutoCommit() method
23. setCommit() method
24. The Statement object’s executeUpdate() method returns
25. int
26. String
27. Resultset object
28. Integer object
29. Which of the statements allow drivers to be loaded?
30. Class.forName()
31. Class.loadDriver()
32. DriverManager.registerDriver()
33. DriverManager.loadDriver()
34. Which of the following methods is best suitable for executing a DDL statement?
35. executeQuery()
36. executeUpdate()
37. execute()
38. getResultSet()
39. Which of the following is the correct code snippet for establishing connection and executing the query?

Class.forName("com.mysql.jdbc.Driver");

Connection connection = DriverManager.getConnection("jdbc:mysql://localhost/test");

Statement stmt = connection.createStatement();

ResultSet rs = stmt.executeQuery("select \* from students where grade='A'");

Connection connection = DriverManager.getConnection("jdbc:mysql://localhost/test");

Class.forName("com.mysql.jdbc.Driver");

Statement stmt = Statement.createStatement(connection);

ResultSet rs = stmt.executeQuery("select \* from students where grade='A'");

Connection connection = DriverManager.getConnection(Class.forName("com.mysql.jdbc.Driver"));

Statement stmt = connection.createStatement(connection);

ResultSet rs = connection.executeQuery("stmt");

1. None of the Above

**Threads:**

|  |
| --- |
| 1. What is the name of the interface that can be used to define a class that can execute within its own thread? 2. Runnable 3. Run 4. Threadable 5. Thread 6. Executable |
| 1. Garbage Collector is a 2. Main Thread 3. Daemon Thread 4. High-priority Thread 5. Normal Thread |
| 1. Which of the following events will cause the thread to die ? 2. sleep( ) method is called 3. wait( ) method is called 4. Execution of run( ) method ends 5. Execution of thread constructor ends |
| 1. The static method \_\_\_\_\_\_\_\_\_\_\_in the Thread class can be used to obtain a reference to the Thread object associated with the current thread. 2. currentThread() 3. thisThread() 4. this() 5. current() |

|  |
| --- |
| 1. What will happen when you attempt to compile and run this code?   public class Runt implements Runnable{  public static void main(String argv[]){  Runt r = new Runt();  Thread t = new Thread(r);  t.start();  }  public void start(){  for(int i=0;i<100;i++)  System.out.println(i);  }  }   1. Compilation and output of count from 0 to 99 2. Compilation and no output 3. Compile time error: class Runt is an abstract class. It can't be instantiated. 4. Compile time error, method *start* cannot be called directly |
| 1. Given the code:   class Thread1 extends Thread{  Thread1(){  start();  }  public void run(){  Thread cur=Thread.currentThread();  // code goes here  int p=cur.getPriority();  System.out.println("Thread Priority :"+p);  }  public static void main(String args[]){  Thread1 m1=new Thread1();  }  }  Which of the following statements are valid to set the priority of the current thread to 5?  a. cur.setPriority(Thread.NORM\_PRIORITY);  b. cur.setPriority(5);  c. No need to write any code because the default priority of all the threads is 5.  d. Priority of user defined threads cannot be changed |

1. What is the name of the method used to schedule a thread for execution?
2. init();
3. start();
4. run();
5. resume();
6. Which of the following methods may cause a thread to stop executing?
7. sleep();
8. stop();
9. yield();
10. wait();
11. notify();
12. notifyAll()
13. synchronized()
14. Which of the following best describes the use of the synchronized keyword?
15. Allows two process to run in paralell but to communicate with each other
16. Ensures only one thread at a time may access a method or object
17. Ensures that two or more processes will start and end at the same time
18. Ensures that two or more Threads will start and end at the same time
19. What is the initial state of a thread when it is created and started?
20. Ready
21. Running
22. Waiting
23. Sleeping
24. Which of the following statements are true?
25. The sleep method takes parameters of the Thread and the number of seconds it should sleep
26. The sleep method takes a single parameter that indicates the number of seconds it should sleep
27. The sleep method takes a single parameter that indicates the number of milliseconds it should sleep
28. The sleep method is a static member of the Thread class
29. Which is the correct way to start a new thread? Select the one correct answer.
30. Just create a new Thread object. The thread will start automatically.
31. Create a new Thread object and call the method begin().
32. Create a new Thread object and call the method start().
33. Create a new Thread object and call the method run().
34. Create a new Thread object and call the method resume().
35. When extending the Thread class to provide a thread's behavior, which method should be overridden? Select the one correct answer.
36. begin()
37. start()
38. run()
39. resume()
40. behavior()
41. What will be the result of attempting to compile and run the following program?

public class MyClass extends Thread {

public MyClass(String s) { msg = s; }

String msg;

public void run() {

System.out.println(msg);

}

public static void main(String[] args) {

new MyClass("Hello");

new MyClass("World");

}

}

Select the one correct answer.

1. The program will fail to compile.
2. The program will compile without errors and will print Hello and World, in that order, every time the program is run.
3. The program will compile without errors and will print a never-ending stream of Hello and World.
4. The program will compile without errors and will print Hello and World when run, but the order is unpredictable.
5. The program will compile without errors and will simply terminate without any output when run.

**HTML:**

1. Consider the following HTML document:  
     
   <TABLE COLS=2 BORDER=1>  
    <TR>  
    <TD>One</TD>  
    </TR>  
    <TR>  
    <TD>One</TD><TD>Two</TD><TD>Three</TD>  
    </TR>  
   </TABLE>  
     
   How will it show up in the browser?
2. As a table with two rows with one column each.
3. As a table with two rows and two columns each
4. As a table with two rows and three columns each
5. As a table with two rows, the first row having one column and the second row having three columns
6. <OL> tag should enclose which of the following tags?

a. <IL>

b. <LI>

c. <UL>

d. None of above

1. In HTML, the \_\_\_\_\_\_\_ attribute gives the address of the script/url that will process the form contents.

A: method

B: action

C: type

D: encType

1. HTML document start and end with which tag pairs?

a. <HEAD>….</HEAD>  
b. <BODY>….</BODY>  
c. <HTML>….</HTML>  
d. <WEB>….</WEB>

1. How do you add a row in a HTML table?

A: <tablerow>

B: <row>

C: <tr>

D: <td>

Choose the correct option to create a HTML hyperlink

A: <a href=”http://[www.someserver.com](http://www.someserver.com)”>Go to Some server</a>

B: <a name=”http://[www.someserver.com](http://www.someserver.com)”>Go to Some server</a>

C: <url=<http://www.someserver.com>>Go to Some server</url>

D: <href=<http://www.someserver.com>> Go to Some server>/href>

1. Which is the correct way to define a multi-line text input control in a HTML form?

A: <input type=”text” multiline=”true”></input>

B: <textarea rows="4" cols="50"> text goes here</textarea>

C: <input type=”textarea”> text goes here </input>

D: <multitext rows=”4” cols=”50”> text goes here </multitext>

1. Which is the correct CSS syntax?

A: {body:color=blue(body}

B: body:color=blue

C: {body;color:blue}

D: body {color: blue}

1. What is the correct HTML for inserting an image?
2. Top of Form
3. <img src="image.gif" alt="MyImage">
4. <image src="image.gif" alt="MyImage">
5. <img alt="MyImage">image.gif</img>
6. <img href="image.gif" alt="MyImage">
7. Which is the correct way to bring in an external script say “myscript.js” into out HTML page?

A: <script source="myscript.js">

B: <include script="myscript.js">

C: <script name="myscript.js">

D: <script src="myscript.js">

1. Javascript code can be written in (choose the most correct option):

A: <head> section only

B: <body> section only

C: <body> section and external (.js) file

D: All of the above

1. Given the following Javascript code:

var billStatus = false;

console.log(typeof(billStatus));

What will be displayed on console?

A: false

B: string

C: boolean

D: 0

1. In Javascript, the ­\_\_\_\_\_\_\_\_\_\_\_ function returns the portion of the string, starting at the specified index and extending for a given number of characters afterward.

A: slice()

B: split()

C: substring()

D: search()

1. Given this Javascript code:

var arr = [1,2,3]

arr.push(4,5)

console.log(arr.length)

What will be the output?

A: 3

B: 4

C: 5

D: incorrect function push()

1. This Javascript Document method returns an element object representing the element whose [id](https://developer.mozilla.org/en-US/docs/Web/API/Element/id) property matches the specified string

A: getElementsById()

B: getElementById()

C: getElementsByTagName()

D: getElementsByName()

1. Given the following Javascript code:

<p id=“p1">Click the button to change the text.</p>

<button onclick="f1()">Try it</button>

Choose the correct code that will change the contents of the paragraph p1:

A:

function f1(){

document.getElementByTagName(“p").innerHTML="New Content";

}

B:

function f1(){

document.getElementById(“p1").value="New Content ";

}

C:

function f1(){

document.getElementById(“p1").innerHTML="New Content ";

}

D:

function f1(){

document.getElementById(“p1").textValue="New Content ";

}

1. Given the following Javascript code to perform form-level validation:

<script >

function validate(){

//code to do validation of all form elements

}

</script>

What is the correct code to put into <form> tag to perform this validation and if validation fails, do not allow form to be submitted to server-side?

A: <form id="form1" onsubmit=“validate()”>

B: <form id="form1" onclick=“validate()”>

C:

<form id="form1" >

…

<input type=”submit” onclick=”validate()”>

</form>

D: <form id="form1" onsubmit=“return validate()”>

------------------------------ servlets / JSP -----------------------------------------------

1. What is the superclass of HttpServlet?
2. GenericServlet
3. Servlet
4. ServletConfig
5. Object
6. Which of the following method calls can retrieve the "email" value sent from the browser?

<form action="/myapp/servlet/EmailCatcherServlet">  
 Please enter your email: <input type="text" name="email">  
 <input type="submit">  
 </form>

1. getParameter("email") of ServletRequest
2. getField("email") of HttpServletRequest
3. getFormValue("email") of HttpServletRequest
4. getParameters("email") of HttpServlet
5. Which of the following is true about servlets?
6. Servlets execute within the address space of web server
7. Servlets are platform-independent because they are written in java
8. Servlets can use the full functionality of the Java class libraries
9. All of the above
10. When a servlet accepts a service call from a client, it receives two objects, \_\_\_and \_\_\_\_\_
11. ServletRequest , ServletResponse
12. Request , Parameters
13. HTTP Header , Body
14. None of the above
15. The getParameter() method belongs to \_\_\_\_\_\_\_
16. Servlet
17. ServletRequest
18. ServletResponse
19. All of the above
20. The method getSession() is defined in one of the following interfaces.
21. HttpServletRequest
22. ServletRequest
23. HttpServletSession
24. HttpSession
25. If you want to declare class variable in JSP, within what type of tags must you declare the variable?
26. <%@ … %>
27. <%!.... %>
28. <%.....%>
29. <%-- --%>
30. The init() and destroy() methods belong to \_\_\_\_\_\_\_\_
31. Service Interface
32. GenericServlets Interface
33. ServletContext Interface
34. None of the above
35. Servlets can run on \_\_\_\_\_\_\_\_\_\_ web server
36. Any
37. Any JEE compliant
38. JBOSS
39. All of the above
40. The \_\_\_\_\_\_\_\_\_ class of status code indicates an error in the client
41. 2xx
42. 3xx
43. 4xx
44. 5xx
45. Which of the following code is used to retrieve value of an attribute in a HTTP Session object in servlets?

[A.](javascript:%20void(0)) session.getAttribute(String name)

[B.](javascript:%20void(0)) session.retrieveAttribute(String name)

[C.](javascript:%20void(0)) session.attribute(String name)

[D.](javascript:%20void(0)) session.setAttribute(String name)

1. Which of the following lines would you use to include the output of DataServlet into any other servlet?

a. RequestDispatcher rd = request.getRequestDispatcher("/servlet/DataServlet");

rd.include(request, response);

b. RequestDispatcher rd = request.getRequestDispatcher("/servlet/DataServlet");

rd.include(response);

c. RequestDispatcher rd = request.getRequestDispatcher();

rd.include("/servlet/DataServlet", request, response);

d. RequestDispatcher rd = request.getRequestDispatcher();

rd.include("/servlet/DataServlet", response);

1. Consider the following servlet snippet:

protected void doGet(HttpServletRequest req, HttpServletResponse res)  
 throws ServletException, IOException  
 {  
 \_\_\_\_\_\_\_\_\_\_\_\_\_(1)\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 \_\_\_\_\_\_\_\_\_\_\_\_\_\_(2)\_\_\_\_\_\_\_\_\_\_\_\_\_  
 }

What must replace (1) to set the content type of the response?  
What must replace (2) to request a PrintWriter object to write text to the response message?

a.

(1) req.setContentType("text/html");  
(2) PrintWriter out = HttpServletResponse.getOutputStream();

b.

(1) res.setContentType("text/html");  
(2) PrintWriter out = res.getWriter();

c.

(1) HttpServletResponse.setContentType("text/html");  
(2) PrintWriter out = res. getOutputStream();

d.

(1) request.setContentType("text/html");  
(2) PrintWriter out = HttpServletResponse. getWriter();

1. We have following methods declared in our servlet:  
     
   public void service(HttpServletRequest request, HttpServletResponse response)   
   { System.out.println("Service Method"); }  
     
   public void doGet(HttpServletRequest request, HttpServletResponse response)   
   { System.out.println("Get Method"); }  
     
   public void doPost(HttpServletRequest request, HttpServletResponse response)   
   { System.out.println("Post Method"); }

This servlet is invoked by an HTML in which we have specified method="post"

What output will it show on the console?

1. Service Method  
   Get Method  
   Post Method
2. Service Method  
   Post Method
3. Service Method
4. Post Method
5. If you want to declare class/instance variable in JSP, within what type of tags must you declare the variable?
6. <%@ … %>
7. <%!.... %>
8. <%.....%>
9. <%-- --%>
10. Which of the following is the proper way to include java.util package in your jsp page?
11. <%= import java.util.\*; %>
12. <%@ page import="java.util.\*" %>
13. <%@ import class="java.util.\*" %>
14. <page:import package="java.util">
15. The My.jsp is loaded and executed twice, then server is restarted and again My.jsp is requested. What processing will be done with respect to My.jsp?

1) JSP translated to Servlet and compiled

2) Servlet loaded into memory

3) jspInit called

4) jspService called

1. Only 4
2. Only 3,4
3. 2,3,4
4. 1,2,3,4
5. In EL(Expression Language), expressions are invoked through the construct \_\_\_\_\_\_\_\_\_
6. #{expression}
7. $(expression)
8. <%= expression %>
9. ${expression}
10. \_\_\_\_\_\_\_ provides a set of reusable standard tags.
    1. JSP actions
    2. JSP custom tags
    3. JSP directives
    4. JSTL
11. <c:set var="city" value="${param.city}" /> What does the above code implement ?
12. Variable city is assigned value of user bean’s property
13. Variable called ‘city’ is set with the value accepted as parameter
14. Bean property address is set to ‘city’
15. Variable city is declared.

Spring

1. Spring is based on the \_\_\_\_\_\_\_\_\_\_ pattern.

A.SessionFactory

B. Observer

C. Inversion of Control/Dependency Injection

D. None of the above

1. By default the beans in Bean factories are treated as :

A. Non Transient

B. Multiple instance generator

C. Singleton

D.All of the above

1. What kind of requirements are applied to beans so that they could be managed by Spring?
2. Bean object ***must*** implement some business interface
3. Bean object ***must*** extend org.springframework.Bean class
4. Bean object ***must*** have public no-arg constructor
5. To be a spring bean object doesn't have to extend or implement anything.
6. Given the following statements:

A) ClasspathXmlApplicationContext **is a** ApplicationContext

B) ApplicationContext **is a** BeanFactory

1. Both A and B are true
2. Only A alone is true
3. Only B alone is true
4. Both A and B are false
5. When a bean is instantiated, some initialization can be performed. This can be done using the \_\_\_\_\_\_\_\_\_\_\_ attribute of the <bean> to invoke a setup operation.
6. init-method
7. initialize-method
8. setup
9. setup-method
10. Fill in the blanks to complete the given code

Resource res = new ClassPathResource("currencyconverter.xml");  
 BeanFactory factory = new XmlBeanFactory(res);  
 CurrencyConverter curr = (CurrencyConverter) factory.\_\_\_\_\_\_\_\_\_\_\_\_("currencyConverter");

1. retrieveBean
2. getBean
3. get
4. getFactoryBean
5. You may want all properties of a JavaBean to be set just once, when the bean is created and become immutable after that point.

Which of the given may be best choice for this?

1. Interface injection
2. Setter injection
3. Constructor injection
4. None of the above
5. Examine the below class:

public class Employee  
{  
 public Employee(int empId, int age){  
 ....  
 }  
 ...  
}  
Consider Spring Configuration code snippet for the above class:  
<bean id="employee" class="Employee">  
<constructor-arg \_\_\_\_\_\_\_\_\_\_\_ value="2001"/>  
<constructor-arg \_\_\_\_\_\_\_\_\_\_\_ value="24"/>  
</bean>  
Select correct option(s) to fill the blank lines in the above code for completing the configuration of the bean to inject values for the properties through constructor injection.

a)

type="int"  
type="double"

b)

index="0"  
index="1"

c)

index="1"  
index="2"

d)

order="0"  
order="1"

1. What can be considered as a realization of the Front Controller design pattern in the case of Spring MVC?
2. HandlerMapping
3. ViewResolver
4. DispatcherServlet
5. ModelAndView
6. What are beans in the concept of Spring or what are spring beans?
7. Controller classes
8. Service classes
9. Repository
10. Any class that is managed by the container.
11. What is the attribute of java bean to specify scope of bean to have single instance per Spring IOC?
12. Prototype
13. Session
14. Request
15. Singleton
16. Match the following:

1. @Component   
2. @Controller  
3. @Service   
4. @Repository  
  
a. Business logic implementation class will be marked with this annotation  
b. Represent a data access logic class  
c. Classes annotated with this is referred as spring beans  
d. Presentation component in Spring MVC will be annotated with this.

Options are:

1. 1 - b, 2 - d, 3 - a, 4 – c
2. 1 - c, 2 - a, 3 - d, 4 – b
3. 1 - c, 2 - d, 3 - b, 4 – a
4. 1 - c, 2 - d, 3 - a, 4 - b
5. Spring promotes loose coupling through a technique known as :
6. Aspect Oriented Programming
7. Inversion of Control (IoC)
8. JNDI
9. Spring does not support loose coupling
10. In IOC the objects give their dependencies instead of creating or looking for dependent objects
11. TRUE
12. FALSE
13. Examine the below code in Spring configuration file:

<beans>  
<bean id="testBean" class="com.igate.beans.Test" >  
 <property name="testId" value="101"></property>  
 <property name="testName" value="Java"></property>  
 <property name="invigilators">  
 <list>  
 <ref bean="staff1"/>  
 <ref bean="staff2"/>  
 </list>  
 </property>  
</bean>  
<bean id="staff1" class="com.igate.beans.Staff">  
 <property name="staffId" value="4001"></property>  
 <property name="name" value="Rama"></property>  
</bean>  
  
<bean id="staff2" class="com.igate.beans.Staff">  
 <property name="staffId" value="4002"></property>  
 <property name="name" value="Sita"></property>  
</bean>  
</beans>  
Choose the appropriate spring beans code which will match with the given configuration code.

A.

package com.igate.beans;  
public class Staff {  
 private int staffId;  
 private String name;  
 //getter and setter methods for these properties  
}

b.

package com.igate.beans;  
public class Staff {  
 private int staffId;  
 private String name;  
 private List<Test> tests;  
 //getter and setter methods for these properties  
}

c.

public class Test {  
 private int testId;  
 private String testName;  
 private List<Staff> invigilators;  
 //getter and setter methods for these properties  
}

d.

public class Test {  
 private int testId;  
 private String testName;  
 private Staff invigilators;  
 //getter and setter methods for these properties  
}

1. Savitha wants to delete purchase details from database through spring Application. She has used the below partial code for deletion.  
     
   1. String sqlquery="DELETE FROM purchase WHERE purchaseId=101";  
   2. int count = jdbcTemplate.\_\_\_\_\_\_\_\_\_\_\_(sqlquery);  
     
   Which need to be inserted at line 2 to complete this implementation?
2. queryForList
3. deleteFrom
4. update
5. updateFrom
6. Which of the following statements are true of the JdbcTemplate class?
7. It’s an abstract class
8. It needs to be sub classed
9. Executes the core JDBC workflow like statement creation and execution
10. Executes SQL queries, update statements or stored procedure calls
11. Given the following code:  
      
    jt.query ( "select ename, job from emp", new EmployeeRowMapper());  
    ------------------------------------------------------------------------------  
    class EmployeeRowMapper implements RowMapper {  
     . . .  
    }   
      
    What method must EmployeeRowMapper class implement to convert ResultSet records to Domain Objects?
12. rowMap()
13. mapRow()
14. mapResultset()
15. resultsetMap()
16. What can be considered as a realization of the Front Controller design pattern in the case of Spring MVC?
17. HandlerMapping
18. ViewResolver
19. DispatcherServlet
20. ModelAndView
21. The view resolver’s job is to :
22. Map URL patterns to Controller objects.
23. Return a ModelAndView object to the DispatcherServlet
24. Take the view name returned in the ModelAndView and map it to a view
25. None of the above
26. org.springframework.jdbc.core.RowMapper is an interface. Classes that implement this interface must implement the \_\_\_\_\_\_\_\_\_\_\_\_ method to map each row of data in the ResultSet
27. rowMap(ResultSet rs)
28. mapRow(ResultSet rs)
29. mapResultset(ResultSet rs)
30. resultsetMap(ResultSet rs)
31. "prefix" and "suffix" are properties of the:
32. InternalResourceViewResolver class
33. JstlView class
34. XMLViewResolver class
35. ViewResolver class
36. In Spring, the\_\_\_\_\_\_\_\_\_\_\_, implemented as a bean factory post-processor, is used to externalize some property values from a BeanFactory definition, into another separate file in Java Properties format.
37. PropertyPlaceholder
38. PlaceholderConfigurer
39. PropertyPlaceholderConfigurer
40. PropertyPlaceConfigurer
41. Which of the following statement(s) is/are true about @RequestMapping annotation?
42. @RequestMapping annotation can be used only at Method Level.
43. @RequestMapping annotation can be used at Class Level only.
44. @RequestMapping annotation can be used at both class and method Level.
45. @RequestMapping can also be used to specify the acceptable request method.
46. Where do you expect to find the servlet specific Spring config to the example servlet in the web.xml below?

<servlet>

   <servlet-name>example</servlet-name>

   <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

   <load-on-startup>1</load-on-startup>

</servlet>

1. /WEB-INF/example-servlet.xml
2. /WEB-INF/servlet-config.xml
3. /WEB-INF/default-config.xml
4. /WEB-INF/example-servlet-config.xml

ViewResolvers are configured in

1. servlet-name.xml
2. servletname-servlet.xml
3. controller-servlet.xml
4. springmvc-config.xml

SpringBoot Post assessment

1. **Class used to have full access to the Spring context’s life cycle machinery and dependency injection.**
2. DelegatingFilterProxy
3. WebApplicationContextUtils.getRequiredWeb()
4. WebApplicationUtils.getRequiredWebApplicationContext()
5. None of the mentioned
6. **In JAX-WS, Which of the following annotation you will use to mark a method as a Web Service method:**
7. WebServiceMethod
8. ServiceMethod
9. WebMethod
10. WebService
11. **In RESTful web services, ideally, the @Produces annotation can be used with methods that are annotated with:**
12. @GET
13. @POST
14. @PUT
15. @DELETE
16. **What are the possible mechanisms provided by Spring Security to store user details? Select one or more correct answers.**
17. Database
18. JAAS
19. LDAP
20. Properties file
21. **What is the name of the default environment configuration file of Spring Boot?**
22. configuration.spring
23. configuration.xml
24. application.properties
25. application.json
26. **Which among these is not an applications server provided by Spring Boot?**
27. Jetty
28. Tomcat
29. Undertow
30. EclipseLink
31. **Spring Boot reduces the need to write lots of configuration?**
32. True
33. False
34. **Why is it possible to get started with minimum effort on Spring Boot?**
35. Because it has easy to use codes
36. Because it is enabled by the Spring framework
37. Because it has an opinionated view on Spring platform
38. Because it is well explained
39. **Which of the following features are enabled using @SpringBootApplication annotation?**

A) @Component  
B) @EnableAutoConfiguration  
C) @ComponentScan  
D) @Bean  
E) @Transactional  
F) @Configuration

1. A,B,C,D,E,F
2. A,B,C
3. A,B,C,F
4. B,C,F
5. **What annotation is used to map a method for PUT request? Select the most correct answer**
6. @RequestMapping
7. @Put
8. @PutMapping
9. @PostMapping
10. **What annotation is used to map value to the method argument in** [**http://localhost/factorial/{value}**](http://localhost/factorial/%7bvalue%7d)**?**
11. @Map
12. @Param
13. @RequestParam
14. @PathVariable
15. **What is the property used to change the port to 9090 in a Spring Boot application?**
16. server=9090
17. port=9090
18. server.port=9090
19. Cannot be changed
20. **In Spring Web App, which of the following acts as a Front Controller?**
21. @RestController
22. DispatcherServlet
23. Models
24. Views
25. **\_\_\_\_\_\_\_\_\_\_\_\_ is one approach to create Spring Boot project.  It is a web tool which is provided by Spring on official site. You can create Spring Boot project by providing project details.**
26. Spring CLI
27. Spring Maven
28. Spring Initializr
29. Spring Web
30. **For building a RESTful web service, we need to add:**
31. spring-boot-starter-web
32. spring-boot-starter-rest
33. spring-boot-starter-webmvc
34. spring-boot-starter



1. Which tag informs the spring container about the use of AspectJ annotation?  
a) aop:aspectj-autowire  
b) aop:aspectj-name  
c) aop:aspectj-autoproxy  
d) none of the mentioned

2. Which of the following is advice supported by Aspect Annotation?  
a) @Before  
b) @After  
c) @AfterReturning  
d) All of the mentioned

5. Which advice is executed only when joint point returns or throws an exception?  
a) @Before  
b) @After  
c) @AfterReturning  
d) @AfterThrowing

Which annotation is not Spring Boot Annotation?

@Controller

@RestController

@Service

@Data

What is the correct syntax for annotation wiring?

<annotation-context:config /> to bean configuration.

<annotation-config /> to bean configuration.

<annotation-context-config /> to bean configuration.

<context:annotation-config/> to bean configuration.

.The following are valid techniques to offer configuration metadata to spring container, except \_\_\_\_\_\_\_\_

XML-based configuration file

Annotation-based configuration

Java-based configuration

JSON-based configuration file