Top of Form



|  |
| --- |
| QuestionID  :  1182         Subject Name  Core Java |
| Q1. Which of the following control expressions are valid for an if statement? |
| 1.  an integer expression |
| 2.  a boolean expression |
| 3.  Both 1 and 2 |
| 4.  Neither 1 nor 2 |
| **Correct Answer : 2** |
| Your Answer : 1 |
| QuestionID  :  1200         Subject Name  Core Java |
| Q2. Consider the following code: char c = `a`; switch(c) { case `a`: System.out.println("A"); case `b`: System.out.println("B"); default: System.out.println("C");} For this code, which of the following statements is true? |
| 1.  output will be A |
| 2.  output will be A followed by B |
| 3.  output will be A, followed by B, and then followed by C. |
| 4.  code is illegal and therefore will not compile |
| **Correct Answer : 3** |
| Your Answer : 2 |
| QuestionID  :  1204         Subject Name  Core Java |
| Q3. What is the error in the following code? class Test {abstract void display( );} |
| 1.  No error |
| 2.  Method display( ) should be declared as static |
| 3.  Test class should be declared as abstract |
| 4.  Test class should be declared as public |
| **Correct Answer : 3** |
| Your Answer : 2 |
| QuestionID  :  1221         Subject Name  Core Java |
| Q4. What will be the output of the following program? class Main1{ public static void main(String args[]){ boolean b = true; System.out.println("XXX"); return; System.out.println("YYY");}} |
| 1.  XXX |
| 2.  YYY |
| 3.  XXX followed by YYY |
| 4.  Error. Won`t compile |
| **Correct Answer : 4** |
| Your Answer : 2 |
| QuestionID  :  1227         Subject Name  Core Java |
| Q5. Given the following code: class Base{int x = 10;} class Derived extends Base{ int x = 20; public static void main(String args[]){ Base B = new Base(); Derived d = new Derived(); Base bd = new Derived(); System.out.println(b.x. + " " + d.x + " " + bd.x);}} The above code will produce the output |
| 1.  10 20 20 |
| 2.  10 20 10 |
| 3.  20 10 20 |
| 4.  20 20 10 |
| **Correct Answer : 2** |
| Your Answer : 2 |
| QuestionID  :  1250         Subject Name  Core Java |
| Q6. Package p1 contains the following code: package p1; public class Student{Body of student} class Test{Body of Test}  Now consider the following code: import p1.\*; class Result{ Student s1; Test t1;}  This code will not compile because |
| 1.  Class Result should be declared public. |
| 2.  Student class is not available |
| 3.  Test class is not available. |
| 4.  Result body is not fully defined. |
| **Correct Answer : 3** |
| Your Answer : 2 |
| QuestionID  :  1255         Subject Name  Core Java |
| Q7. Declarations can appear anywhere in the body of a Java method. |
| **Correct Answer : T** |
| Your Answer : T |
| QuestionID  :  1265         Subject Name  Core Java |
| Q8. When present, package must be the first noncomment statement in the file |
| **Correct Answer : T** |
| Your Answer : T |
| QuestionID  :  1282         Subject Name  Core Java |
| Q9. It is perfectly legal to assign a subclass object to a superclass referacence. |
| **Correct Answer : T** |
| Your Answer : T |
| QuestionID  :  1330         Subject Name  Core Java |
| Q10. DataInput is : |
| 1.  an abstract class defined in java.io |
| 2.  a class we can use to read primitive data types |
| 3.  an interface that defines methods to open files |
| 4.  an interface that defines methods to read primitive data types. |
| **Correct Answer : 4** |
| Your Answer : 2 |
| QuestionID  :  1352         Subject Name  Core Java |
| Q11. The default layoutmanager of the content pane of a JFrame is the flowlayout manager. |
| **Correct Answer : F** |
| Your Answer : 0 |
| QuestionID  :  1358         Subject Name  Core Java |
| Q12. A JPanel may not contain another JPanel. |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  1381         Subject Name  Core Java |
| Q13. The methods wait( ) and notify( ) are defined in |
| 1.  java.lang.String |
| 2.  java.lang.Runnable |
| 3.  java.lang.Object |
| 4.  java.lang.Thread |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  1384         Subject Name  Core Java |
| Q14. Every call to wait has a corresponding call to notify that will eventually end the waiting. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  1395         Subject Name  Core Java |
| Q15. Which exception is thrown by the read() method of InputStream class? |
| 1.  FileNotFoundException |
| 2.  ReadException |
| 3.  IOException |
| 4.  None of the above. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  1400         Subject Name  Core Java |
| Q16. It is an error to catch the same type of exception in two different catch blocks |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9297         Subject Name  Core Java |
| Q17. you can freely intermix the java 1.0 containment event model with  the java 1.1 delegation based event model. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9314         Subject Name  Core Java |
| Q18. Which of the following are true? |
| 1.  The event-inheritance model has the event-delegation model. |
| 2.  The event-inheritance model is more efficient than event- delegation model. |
| 3.  The event-delegation model uses the event listeners to define methods of event handling classes. |
| 4.  The event-delegation model uses the handleEvent() method to support event handling. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9318         Subject Name  Core Java |
| Q19. A \_\_\_\_\_\_ is an object that originates or "fire" events. |
| 1.  Event |
| 2.  Source |
| 3.  Trigger |
| 4.  None of above. |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9340         Subject Name  Core Java |
| Q20. The event listener corresponding to handling keyboard events is the \_\_\_\_\_\_\_\_\_. |
| 1.  Event Listener |
| 2.  Key Listener |
| 3.  Mouse Listener |
| 4.  None. |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9354         Subject Name  Core Java |
| Q21. given the following method, which statements will throw an exception, assuming assertions are enabled?  static int inv(int value) {  assert value > -50 : value < 100;  return 100/value;  } |
| 1.  inv(-50); |
| 2.  inv(0); |
| 3.  inv(50); |
| 4.  both (a) nad (b). |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9367         Subject Name  Core Java |
| Q22. which statement are true? |
| 1.  the class thrad is abstract. |
| 2.  the class thread implements runnable |
| 3.  a program terminates when the last non-daemon thread ends |
| 4.  both 2 and 3 |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9369         Subject Name  Core Java |
| Q23. Which of the following methods helps in getting the local host ip? |
| 1.  getLocalHost() |
| 2.  getRemoteHost() |
| 3.  getIpAddress() |
| 4.  getHost() |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9391         Subject Name  Core Java |
| Q24. What will happen when you attempt to compile and run the following code?  class Background implements Runnable  {  int i=0;  public int run()  {  while (true)  {  i++;  System.out.println("i="+i);  }//End while  reurn 0;  }//End run  } |
| 1.  It will compile and the run method will print out the increasing value of i |
| 2.  It will compile and calling start will print out the increasing value of i |
| 3.  The code will cause an error at the compile time |
| 4.  Compilation will cause an error because while cannot take parameter of true |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9437         Subject Name  Core Java |
| Q25. which of the following are true? |
| 1.  the AWT automatically causes a window to be repainted when a portion of a window has been minimized and then maximised |
| 2.  the AWT automatically causes a window to be repainted when application data is changed |
| 3.  The AWT does not support repainting operations |
| 4.  All of the above |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9441         Subject Name  Core Java |
| Q26. import java.io.\*;  class TransientWriter implements Externalizable  {  private transient String s="Hope I can never be persistant!";  public void writeExternal(ObjectOutput oOut)throws IOException  {  oOut.writeObject(s);  }  public void readExternal(ObjectInput oIn)throws IOException,ClassNotFoundException  {  s=(String)oIn.readObject();  }   public String toString()  {  return s;  }  };   public class K  {  public static void main(String args[]) throws IOException,ClassNotFoundException  {  TransientWriter tw=new TransientWriter();  ObjectOutputStream out=new ObjectOutputStream(new FileOutputStream("tw.out"));  out.writeObject(tw);  ObjectInputStream in=new ObjectInputStream(new FileInputStream("tw.out"));  TransientWriter tw2=(TransientWriter)in.readObject();  System.out.println(tw2);  }  };  Attempting to compile and run the above code |
| 1.  will cause a compiler error due to the attempt to write a transient object. |
| 2.  will cause a runtime exception when an attempt is made to write a transient object. |
| 3.  will not cause any runtime error and the transient object is written to the file named "tw.out". |
| 4.  will not cause any runtime error and the transient object is not written to the file named "tw.out".The program prints ablank line on the screen. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9453         Subject Name  Core Java |
| Q27. Which of the following true,select the three correct answer. |
| 1.  A static method mat be invoked before even a single instance of the class is constucted. |
| 2.  A static method cannot access non-static method of the class. |
| 3.  Abstact modifier can apper before a class or a methode but not before a veriable |
| 4.  Final modifier can apper before a class or a veriable but not before a methode. |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9472         Subject Name  Core Java |
| Q28. Which statement is true about a method ? |
| 1.  A method is an implementation of an abstraction |
| 2.  A method is an attribute difining the properties of particular abstraction |
| 3.  A method is a catogery of object |
| 4.   A method is an operation defining the behaviour for a particular abstraction. |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9481         Subject Name  Core Java |
| Q29. Which method is used to determine the class of an object? |
| 1.  getClass() |
| 2.  getObject() |
| 3.  instanceOf() |
| 4.  forName() |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9490         Subject Name  Core Java |
| Q30. What, if anything , is wrong with the following code ?   abstract class MyClass {  transient int j ;  synchronized int k;   final void MyClass () {}  static void f() {}  } |
| 1.  1. The class MyClass can not be declared abstract. |
| 2.  2. The field j can not be declared transient . |
| 3.  3. The field k can not be declared synchronized . |
| 4.  4. Method f() can not be declared static. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9497         Subject Name  Core Java |
| Q31. class Base  {  Base()  {  System.out.println("Message 1");  }  }   abstract class Derived1 extends Base  {  Derived1()  {  System.out.println("Message 2");  }  }   public class Derived2 extends Derived1  {  public Derived2()  {  System.out.println("Message 3");  }  public static void main(String args[])  {  Derived2 d2 = new Derived2();  }  } |
| 1.  will cause a compiler error-The non Avstract classes cannot be extended to the abstract classes. |
| 2.  will cause a compiler error-The abstract classes cannot have constructors defined |
| 3.  will not cause any compiler error.The lines"Message 1" and "Message 3" will be printed on the screen. |
| 4.  will not cause any compiler error.The lines"Message 1","Message 2" and "Message 3" will be printed on the screen. |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9502         Subject Name  Core Java |
| Q32. what command in the java 2SDK should be used to execute the main() method of class named smallprog ? |
| 1.  java Smallprog |
| 2.  javac Smallprog |
| 3.  java Smallprog.java |
| 4.  javac Smallprog.java |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9523         Subject Name  Core Java |
| Q33. class mp  {  int x,y;  mp(int x, int y)  {  this.x=x;  this.y=y;  }  public int addme(int x, int y)  {  return this.x+x+y+this.y;  }  public int addme(mp mpp)  {  return addme(mpp.x,mpp.y);  }  };  class mc extends mp  {  int z;  mc(int x, int y,int z)  {  super(x,y);  this.z=z;   }  public int addme(int x, int y,int z)  {  return this.x+x+y+this.y+this.z;  }  public int addme(mc mcc)  {  return addme(mcc.x,mcc.y,mcc.z);  }  public int addme(int x, int y)  {  return this.x+x+y+this.y;  }  }  public class test  {  public static void main(String[] args)  {  mc mcc=new mc(10,20,30);  mp mpp=new mp(10,20);  int x=mcc.addme(10,20,30);  int y=mcc.addme(mcc);  int z=mpp.addme(mpp);  System.out.println("Hello World!"+(x+y+z));  }  } |
| 1.  300 |
| 2.  240 |
| 3.  120 |
| 4.  compile time error |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9537         Subject Name  Core Java |
| Q34. Given the following code within a method, which statement is true?   int a,b;  b=5;   a. Local variable a is not declared.  b. Local variable b is not declared.  c. Local variable a is declared but not initialised.  d. Local variable b is declared but not initialised.  e. Local variable b is initialised but not declared. |
| 1.  a & b |
| 2.  c & d both |
| 3.  a,d,e |
| 4.  c only |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9538         Subject Name  Core Java |
| Q35. The class java.lang.Exception \_\_\_\_\_\_\_\_\_ .  i. is public  ii. extends Throwable.  iii. Implemets Throwable.  iv. is Serializable. |
| 1.  i,iii,iv. |
| 2.  i,ii. |
| 3.  i,ii,iv. |
| 4.  i,iii. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9545         Subject Name  Core Java |
| Q36. Name the Collection interface implemented by the hash table class |
| 1.  Map |
| 2.  Hash Map |
| 3.  List |
| 4.  Set |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9553         Subject Name  Core Java |
| Q37. In which of these variable declarataions will be the variable remain  uninitialised unless explicitely initialised |
| 1.  Declaration of instance variable of type int. |
| 2.  Declaration of static variable of type float. |
| 3.  Declaration of local variable of type float. |
| 4.  Declaration of static variable of type object. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9562         Subject Name  Core Java |
| Q38. What will be the result of executing the following code?  boolean a=true;  boolean b=true;  boolean c=true;  if(a==true)  if(a==true)  if(b == true)  if(c==true)  System.out.println("Some things are true in this world");  else  System.out.println("Nothings is true in this world");  else if(a && (b==c))  System.out.println("Its too confusing to tell what is true and what is false");   else  System.out.println("Hey this wont compile"); |
| 1.  The code wont compile |
| 2.  "Some things are true in this world" will be printed |
| 3.  "Hey this wont compile" will be printed |
| 4.  None |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9564         Subject Name  Core Java |
| Q39. Is this a complete and legal comment?   /\* // \*/ |
| 1.  No the block comment(/\*...\*/) is not ended since the single-line comments(//...)comments out the closing part. |
| 2.  It is completely valid comment.The // part is ignored by the compiler. |
| 3.  This is combination of comments is illegal and the compiler will reject it. |
| 4.  None of above |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9586         Subject Name  Core Java |
| Q40. Which of these are valid declarations of the main() method in order to start the execution   of a java application?   a. static void main(String[] args) {/\* .... \*/}  b. public static int main(String[] args) {/\* .... \*/}  c. public static void main(String args) {/\* .... \*/}  d. final public static void main(String[] arguments) {/\* .... \*/}  e. public int main(String[] args,int argc) {/\* .... \*/}  f. static public void main(String args[]) {/\* .... \*/} |
| 1.  a,c,f only |
| 2.  d and f only |
| 3.  b,c,d,e only |
| 4.  all of these. |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9594         Subject Name  Core Java |
| Q41. which of the following most closely describe the process of overriding? |
| 1.  A class with the same name replace the functionality of a class defined earlier in the hierarchy |
| 2.  A method with the same name completely replaces the functionality of a method earlier in the hierarchy |
| 3.  A method with the same name but different parameters gives multiple uses for the same method name |
| 4.  A class is prevented from accessing methods in its immediate ancestor |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9595         Subject Name  Core Java |
| Q42. Class class1  {  static{  System.out.println(Thread.getCurrentThread().getName());   }   int meth1(int i)  {  System.out.println(i);  return 0;  }  public static void main(String[] args)  {  System.out.println(Class.for.getName());  int i=0;  i=i++ + meth1(i);  System.out.println(i);   }   } |
| 1.  prints jvm main thread name,class1,1,0. |
| 2.  prints null,name,class1,1,0. |
| 3.  program compiles successfully and give the thread name . |
| 4.  none of above. |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9602         Subject Name  Core Java |
| Q43. which of the following syntax for suggesting that the JVM performs garbage collection: |
| 1.  System.free(); |
| 2.  System.setGarbageCollection(); |
| 3.  System.out.gc(); |
| 4.  System.gc(); |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9641         Subject Name  Core Java |
| Q44. which of he following is illegal identifier |
| 1.  number |
| 2.  $$\_100 |
| 3.  all@hands |
| 4.  Number |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9649         Subject Name  Core Java |
| Q45. which will following expression evaluted using floating-point arithematic  select one answer |
| 1.  2.0\*3.0 |
| 2.  2\*3 |
| 3.  2/3 + 5/7 |
| 4.  4\*5 |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9668         Subject Name  Core Java |
| Q46. Which of the following are legal identifier names in Java. |
| 1.  %abcd |
| 2.  1abcd |
| 3.  \_ab\_cd |
| 4.  \*abcd |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9676         Subject Name  Core Java |
| Q47. which stetement would couse a compliation error if inserted in the local indicated in the following program  public class parameterUse{  static void main(String[] args){  int a=0;  final int b=1;  int[] c={2};  final int[] d={3};  useArgs(a,b,c,d);  }  static void useArgs(final int a,int b,final int[] c,int[] d){  //Inser statement here,  }} |
| 1.  b=a; |
| 2.  d[0]++; |
| 3.  b++; |
| 4.  c=d; |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9694         Subject Name  Core Java |
| Q48. class AA{}  class BB extends AA{}  class Q6  {  public static void main(String[] args)  {  AA a=null;  BB b=(BB)a;  System.out.println(b);  System.out.println(b instanceof BB);  System.out.println(b instanceof AA);  }  } |
| 1.  program compiles correctly and print null,true,false. |
| 2.  program compiles correctly and print null,false,false. |
| 3.  compile time error at line no.8 as null value cannot be casted |
| 4.  run time error at line no. 8 as null value cannot be casted. |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9796         Subject Name  Core Java |
| Q49. Which of the following are true about the class defined inside an interface |
| 1.  a. it is not possible in the java Language. |
| 2.  b. The class is always public. |
| 3.  c. The class is always static. |
| 4.  d. The class method cannot call the method declared in the interface. |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9820         Subject Name  Core Java |
| Q50. Which of the Listener classes have corresponding adpter classes.  1.ContainerListener  2.TextListener  3.ItemListener  4.MouseMotionListener |
| 1.  1,2 |
| 2.  2,4 |
| 3.  3,4 |
| 4.  1,4 |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9835         Subject Name  Core Java |
| Q51. Which of the following is correct? |
| 1.  128 >> 1 gives 64 |
| 2.  128 >>> 1 gives 64 |
| 3.  128 >>> 1 gives –64 |
| 4.  both (a) and (b) |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9836         Subject Name  Core Java |
| Q52. Which of the following wrapper classes have a booleanValue() method? |
| 1.  All wrapper classes |
| 2.  All wrapper classes except void |
| 3.  All wrapper classes that also implement the compareTo() method |
| 4.  only the class Boolean |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9856         Subject Name  Core Java |
| Q53. Which of the following is a legal return type of a method,overloading the following method:  public void add(int a) {…} |
| 1.  int |
| 2.  void |
| 3.  Can be anything |
| 4.  none of the above |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9860         Subject Name  Core Java |
| Q54. Java supports multidimensional arrays. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9874         Subject Name  Core Java |
| Q55. The following code will give   1: class Test  2: {  3: static void show()  4: {  5: System.out.println("Static method in Test");  6: }  7: }  8: public class Q4 extends Test  9: {  10: void show()  11: {  12: System.out.println("Overridden static method in Q4");  13: }  14: public static void main(String[] args)  15: {  16: }  17: } |
| 1.   Compilation error at line 3. |
| 2.  Compilation error at line 10. |
| 3.  No compilation error, but runtime exception at line 3. |
| 4.  No compilation error, but runtime exception at line 10. |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9881         Subject Name  Core Java |
| Q56. What will be the output ?  class abc  {  public static void main(String[] args)  {  StringBuffer sb1=new StringBuffer("abcd");  StringBuffer sb2=sb1;  if (sb1.equals(sb2))  {  System.out.println("OK");  }  else  System.out.println("NOT OK");  }  } |
| 1.  Compile-Error. |
| 2.  OK |
| 3.  NOT OK |
| 4.  Runtime-Error |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9892         Subject Name  Core Java |
| Q57. The code outside of the class Outer can instantiate class Inner. |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  9896         Subject Name  Core Java |
| Q58. Comparison/Logical operators are used for testing and magnitude. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9906         Subject Name  Core Java |
| Q59. The following code will print   1: if( new Boolean("true") == new Boolean("true"))  2: System.out.println("True");  3: else  4: System.out.println("False"); |
| 1.  Compilation error. |
| 2.  No compilation error, but runtime exception. |
| 3.  Prints "True". |
| 4.  Prints "False". |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  9907         Subject Name  Core Java |
| Q60. A reference type or object can never be casted to primitive type. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9912         Subject Name  Core Java |
| Q61. Which statements are true about modifiers?   1.Abstract classes can contain final methods.  2.Fields can be declared native.  3.Non-abstract methods can be declared in abstract classes.  4.Classes can be declared native. |
| 1.  1,2,3,4 |
| 2.  1,3 only |
| 3.  2,3 only |
| 4.  1 only |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9921         Subject Name  Core Java |
| Q62. There is no conversion between boolean and other primitive types. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9926         Subject Name  Core Java |
| Q63. What is the effect of issuing a wait() method on an object |
| 1.  If a notify() method has already been sent to that object then it has no effect |
| 2.  The object issuing the call to wait() will halt until another object sends a notify() or notifyAll() method |
| 3.  An exception will be raised |
| 4.  The object issuing the call to wait() will be automatically synchronized with any other objects using the receiving object. |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  9930         Subject Name  Core Java |
| Q64. What will be output?   class Demo  {  public static void main(String[] args)  {  System.out.println("2"+3+4);   System.out.println(2+3+"4");  }  } |
| 1.  234 & 54 |
| 2.  27 & 234 |
| 3.  none of the above |
| 4.  234 & 234 |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  9942         Subject Name  Core Java |
| Q65. What will be the result of invoking the wait()method on object without ensuring that the current thread holds the lock of the object? |
| 1.  The code will compile to fail |
| 2.  Nothing special will happen |
| 3.  An IllegalMonitorStateException will be thrown if the wait() method is called while the current thread does not hold the lock of the object. |
| 4.  The thread will be blocked until it gains the lock of the object |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9944         Subject Name  Core Java |
| Q66. All the values of the expressions on the right-hand side of the  assignments are implicitly promoted to type int. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9960         Subject Name  Core Java |
| Q67. Which of the following are true? |
| 1.  Component extends Container |
| 2.  MenuItem extends Component |
| 3.  Container extends Component |
| 4.  MenuComponent extends Component |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9968         Subject Name  Core Java |
| Q68. Which methods can be called on objects implementing the Map interface?  (a)contains(Object o)  (b)addAll(Collection c)  (c)remove(Object o)  (d)values()  (e)toArray() |
| 1.  a,b and e |
| 2.  b and c |
| 3.  c and d |
| 4.  b,c,d and e |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  9970         Subject Name  Core Java |
| Q69. Using System.exit(1); in try block will not allow finally code to execute. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  9997         Subject Name  Core Java |
| Q70. class Demo  {  static int total = 10;  public void divide(int a, int b) {   try {   int c = a / b;   } catch (Exception e) {   System.out.print("Exception ");   } finally {   System.out.println("Finally");   }  }    public static void main (String args []) {   new Demo().divide(4,0);   }   }    What will be the output? |
| 1.  Exception |
| 2.  Finally |
| 3.  Exception  Finally |
| 4.  None of above |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  10000         Subject Name  Core Java |
| Q71. what will be the output of the following code?  public class TestClass implements Runnable  {  public void run()  {  while(true)  {   }  }  public static void main (String args[])  {  TestClass nt1=new TestClass();  TestClass nt2=new TestClass();  TestClass nt3=new TestClass();  nt1.run();  nt2.run();  nt3.run();  }  } |
| 1.  The code does not compile |
| 2.  the code compiles  and runs 3 non ending non demon threads. |
| 3.  the code compiles  and runs only 1 non ending non demon thread. |
| 4.  non of the above |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  10016         Subject Name  Core Java |
| Q72. Which are the valid ways to create DataInputStream streams? |
| 1.  new DataInputStream(); |
| 2.  new DataInputStream("in.dat"); |
| 3.  new DataInputStream(new File("in.dat")); |
| 4.  new DataInputStream(new FileInputStream("in.dat")); |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  10019         Subject Name  Core Java |
| Q73. public class Test {   public void print() {   System.out.println("Non-static print");   }  public static void print()  {  System.out.println("Static print");  }   public static void main(String args []) {   Test st1 = new Test();   st1.print();    }   } |
| 1.  Non-static print |
| 2.  Static print |
| 3.  Compilation error |
| 4.  Runtime Error |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  10034         Subject Name  Core Java |
| Q74. Operand stack in java is a 32 bit wide |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10047         Subject Name  Core Java |
| Q75. Does the following piece of code compile ? if yes , what is output ?  class Test09  {  static int tomb ( char a )  {  throw new NumberFormatException();  }  public static void main(String args[])  {  try  {  tomb(`a`);  }  catch ( Exception e )  {  System.out.println ( " Done " ) ;  }  }  } |
| 1.  Compiler error ,throw new NumberFormatException() : checked Exceptions  must be declared in the throws clause or must be caught in the corresponding  catch block. |
| 2.  compiler error , System.out.println( " Done " ):Statement not reached . |
| 3.  NumberFormatException thrown at runtime |
| 4.   output = Done. |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  10049         Subject Name  Core Java |
| Q76. standard parameters of scrollbar are :  scrollbar(int orientation,int value,int visible,int minium,int maximum) |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10075         Subject Name  Core Java |
| Q77. In Hash map will allow the null key pair where in Hasp Table is not Allow |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10107         Subject Name  Core Java |
| Q78. A subclass inherits the superclass constructors. |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  10108         Subject Name  Core Java |
| Q79. You are creating application that has a form with a text entry field used to enter a person age.Which of he following is appropriate for caputring this information. |
| 1.  USe the Text field of a TextField parse the result using Integer |
| 2.  Use the getInteger method of the TextField |
| 3.  Use the getText method of the TextBox and parse the result using the getInt method of Integer class |
| 4.  Use the getText method of the TextField parse the result using the parseInt method of Integer class |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  10123         Subject Name  Core Java |
| Q80. "this()" can be used to call the constructor of same class. |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10143         Subject Name  Core Java |
| Q81. Which statements are true? |
| 1.  The class thread is abstract. |
| 2.  The class Thread implements runnable. |
| 3.  A program terminates when the last non-daemon thread ends. |
| 4.  both option 2 and 3. |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  10158         Subject Name  Core Java |
| Q82. The chained exception feature |
| 1.  allows U to associate another exception with an exception |
| 2.  In this second exception describes the cause of the first exception |
| 3.  Both 1 & 2 |
| 4.  None of above |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  10159         Subject Name  Core Java |
| Q83. What will be the output:  class Employee  {  private String name;  public Employee(String n)  {  name=n;  }  public static void swap(Employee x,Employee y)  {  Employee temp=x;  x=y;  y=temp;  }  public String getName()  {  return name;  }  public static void main(String [] args)  {  Employee A=new Employee("Sita");  Employee B=new Employee("Gita");  System.out.println("Before :A="+A.getName());  System.out.println("Before :B="+B.getName());  swap(A,B);  System.out.println("After :A="+A.getName());  System.out.println("After :B="+B.getName());  }  } |
| 1.  Before :A=Sita  Before :B=Gita  After :B=Gita  After :A=Sita |
| 2.  Before :A=Gita  Before :B=Sita  After :A=Gita  After :B=Sita |
| 3.  Before :A=Sita  Before :B=Gita  After :A=Sita  After :B=Gita |
| 4.  Compilation Error |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  10162         Subject Name  Core Java |
| Q84. What is the result of compiling and running the following code:   public class Test {   static int total = 10;   public static void main (String args []) {   new Test();   }   public Test () {   System.out.println("In test");   System.out.println(this);   int temp = this.total;   if (temp > 5) {   System.out.println(temp);   }   }   } |
| 1.  The class will not compile |
| 2.  The compiler reports and error at line 2 |
| 3.  The compiler reports an error at line 9 |
| 4.  The value 10 is one of the elements printed to the standard output |
| **Correct Answer : 4** |
| Your Answer : |
| QuestionID  :  10206         Subject Name  Core Java |
| Q85. which of these are core interfaces in collection framework |
| 1.  set,collection,map |
| 2.  bag |
| 3.  linkedlist |
| 4.  none |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  10227         Subject Name  Core Java |
| Q86. a final class must be complete i.e.(has implementations of all its methods)where as abstract class is considered incomplete(i.e.implemetations of all methods is not provided). |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10250         Subject Name  Core Java |
| Q87. jre is instance of jvm |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  10257         Subject Name  Core Java |
| Q88. JVM(Java Virtual Machine) is an instance of JRE(Java Runtime Environment) |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10274         Subject Name  Core Java |
| Q89. int i=1;  int arr[]=new int[]{2,3,4};  arr[++i]= --i;  when we print array what is output |
| 1.  0,3,4 |
| 2.  2,3,1 |
| 3.  2,0,4 |
| 4.  2,3,3 |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  10278         Subject Name  Core Java |
| Q90. What is the legal range of a byte integral type? |
| 1.  0 - 65, 535 |
| 2.  (–128) – 127 |
| 3.  (–32,768) – 32,767 |
| 4.  (–256) – 255 |
| **Correct Answer : 2** |
| Your Answer : |
| QuestionID  :  10455         Subject Name  Core Java |
| Q91. what information is needed to create a TCP socket? |
| 1.  the local system s IP Address |
| 2.  the local systems port number |
| 3.  the remote system s IP Address |
| 4.  the remote system s port number |
| **Correct Answer : 3** |
| Your Answer : |
| QuestionID  :  10471         Subject Name  Core Java |
| Q92. Finalise wont be called more than once on the same object |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10473         Subject Name  Core Java |
| Q93. ClassLoader is a public class and extends object class |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  10495         Subject Name  Core Java |
| Q94. getCause() are methods of which class. |
| 1.  Throwable |
| 2.  IOException |
| 3.  RunTimeException |
| 4.  none of the above |
| **Correct Answer : 1** |
| Your Answer : |
| QuestionID  :  10498         Subject Name  Core Java |
| Q95. inner classes are visible to other classes in the same package. |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  10966         Subject Name  Core Java |
| Q96. A constructor must initialize all the fields of a class. |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  11001         Subject Name  Core Java |
| Q97. ClassLoader is a public class and extends object class |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  11003         Subject Name  Core Java |
| Q98. Externalizable is marker Interface |
| **Correct Answer : F** |
| Your Answer : |
| QuestionID  :  11009         Subject Name  Core Java |
| Q99. abstract classes can contain native methods |
| **Correct Answer : T** |
| Your Answer : |
| QuestionID  :  11036         Subject Name  Core Java |
| Q100. What is the output of the following program?  class test  {  int i=(int)34.3E38  System.out.println(i);   } |
| 1.  It is equal to maximum value of the integer; |
| 2.  It gives Runtime Error.Value is greator than limits of Integer . |
| 3.  It gives Compile time error.Value is greator than limits of Integer. |
| 4.  343000000 |
| **Correct Answer : 1** |
| Your Answer : |

Bottom of Form