**Core Java**

**Post Training Assessment**

1. A method is defined in a class as :  
   void processUser(int i) { }  
   If this method is overridden in a sub class,\_  
   1. the new method should return int
   2. (b) the new method can return any type of values
   3. the argument list of new method should exactly match that of overridden method
   4. the return type of new method should exactly match that of overridden method
2. What will be the output?

public class Test1{

public static void main(String[] args){

int arr[] = new int[3];

for(int i = 0;i < arr.length;i++){

System.out.println(arr[i]); }}}

* 1. 0 0 0
  2. ArrayIndexoutOfBoundsException
  3. NullPointerException
  4. null null null

1. Which of the following are valid array declarations?  
   1. int arr[] = new int[];
   2. float arr[10] = new fl
   3. double []arr = new double[10];
   4. None Of the Above.
2. What will be the output ?  
     
   public class Test5{  
   public void Test5(){  
   System.out.println("Constructor1");}  
    public Test5(){   
    System.out.println("Constructor2"); }  
    public static void main(String[] args){  
   Test5 t5 = new Test5();}}
   1. Constructor1 b. Constructor2 c. Constructor1 Constructor2 d. Compiler Error
3. Which of the following are not wrapper classes?
   1. String
   2. Integer
   3. StringBuffer
   4. Boolean
4. Java Identifiers must start with a letter, dollar sign ($), or underscore (\_).

True / False

1. What is the output ?

class C{  
int i;  
public static void main (String[] args) {  
int i;   
private int a = 1;   
protected int b = 1;   
public int c = 1;   
System.out.println(a+b+c); }

* 1. compiletime error at lines 1,2,3,4,5
  2. compiletime error at lines 2,3,4,5
  3. compiletime error at lines 2,3,4
  4. prints 3
  5. None of the above

|  |
| --- |
| 1. What is the result of attempting to compile and run the program ?. Class C { public static void main(String[] args) { int[]a1[]=new int[3][3]; int a2[4]={3,4,5,6};  int a2[5];}}     1. compiletime error at lines 3,4,5    2. compiltime error at line 4,5    3. compiletime error at line 3    4. Runtime Exception    5. None of the above      1. class A{  static String m(float i) {return "float";} static String m(double i) {return "double";} public static void main (String[] args) { int a1 = 1; long b1 = 2; System.out.print(m(a1)+","+ m(b1));}}     1. prints float,foat    2. prints float,double    3. prints double,double    4. compile time error    5. None of the above 2. The length of an Array ‘abc’ can be found using :     1. abc.length    2. abc.length()    3. abc.size    4. abc.size() 3. The constructor of super class can be called using      1. this();    2. super.constructorname();    3. super();    4. super(Object);      1. What is the result of following code ?   class Animal { void method () throws Exception  { }  Animal( String name )  {  System.out.println(name); } }  class dog extends Animal {   void method () throws Exception { } }  class test {  public static void main ( String [] args )  {  new Animal("Giraffe");  }  }   1. Prints Giraffe 2. Compile error 3. Runtime error 4. Prints nothing 5. None of the above |

1. class sup

{  
 static void method(){System.out.println("Super");} // 1  
}

class test extends sup  
{  
 public static void main(String args[]) {}  
 static void method(){System.out.println("test");}  
}

What class modifier(s) can be inserted at line 1 to prevent method() from being overridden without causing compile time errors ?

* 1. final
  2. private
  3. protected
  4. default
  5. Hiding cannot be prevented.

1. What is the o/p of following code?

public class test

{

public static void main( String [] args )

{

new test();

}

test()

{

test(2);

}

test(int x)

{

System.out.println(x);

}

}

1. Prints 2
2. Prints 0
3. Does not compile
4. Runtime Exception.
5. None of the above.
6. Which of the following are not legal identifiers ?
   1. String #baby;
   2. char c123;
   3. Byte $wombat;
   4. Long long;
   5. Short ~english;
7. Identify the super class of all errors and Exceptions in the Java language.
   1. Exception
   2. Error
   3. Throwable
   4. Runnable
8. Which one of the following options is a collection of related classes and Interfaces providing access protection and name space management in java ?  
   1. Class
   2. Package
   3. Interface
   4. Collection
9. \_\_\_\_\_\_\_\_ is the capability of a method to do different things based on the object that is acting upon it.
   1. Encapsulation
   2. Abstraction
   3. Put()
   4. Polymorphism
10. Select the Odd one out.
    1. Private
    2. Final
    3. Public
    4. Default
11. Any exception that is thrown out of a method must be specified by a \_\_\_\_\_\_\_\_ clause.
    1. Catch
    2. Finally
    3. Throw
    4. Throws
12. \_\_\_\_\_ represents a string that can be modified ?
    1. String [];
    2. String
    3. String [] []
    4. StringBuffer
13. \_\_\_\_\_\_ has an interpreter component that enables communication between Java byte code and a computer’s operating system  
    1. JDK
    2. JVM
    3. Java Compiler
    4. Just InTime Compiler
14. The \_\_\_\_\_ keyword allows the main() method to be called, without needing to create an instance of the class.
    1. abstract
    2. implements
    3. synchronize
    4. static
15. An \_\_\_\_\_\_\_\_\_\_ method is a method that don’t have a method definition.
16. Final
17. Private
18. Abstract
19. Interface

25. What is the result of the code ?

class bike

{

}

class arr extends bike{

public static void main(String[] args) {

arr[] a1=new arr[2];

bike[] a2;

a2=a1; //3

arr[] a3;

a3=a1; //5

}}

1. compile time error at line 3
2. compile time error at line 5
3. Runtime exception
4. The code runs fine
5. None of the above

26. When Subclass type refers to the object of Parent class, it is known as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

1. Up casting
2. Inheritance
3. Object
4. Down casting

27. An \_\_\_\_\_\_\_\_\_ is a collection of abstract methods.

1. Interface
2. Class
3. Sub Class
4. Singleton Class

28.  What is the output of this program?

class overload

{

int x;

int y;

void add(int a)

{

x = a + 1; }

void add(int a, int b)

{

x = a + 2;

} }

class Overload\_methods

{

public static void main(String args[])

{

overload obj = new overload();

int a = 0;

obj.add(6);

System.out.println(obj.x);

} }

1. 5
2. 6
3. 7
4. 8

29.  Which two classes use the Shape class correctly?

A. public class Circle implements Shape

{ private int radius; }

B. public abstract class Circle extends Shape

{ private int radius; }

C. public class Circle extends Shape

{

private int radius;

public void draw(); }

D. public abstract class Circle implements Shape

{ private int radius;

public void draw(); }

E. public class Circle extends Shape

{ private int radius;

public void draw()

{ /\* code here \*/ } }

F. public abstract class Circle implements Shape

{ private int radius;

public void draw()

{

/\* code here \*/ } }

1. B,E
2. A,C
3. C,E
4. T,H

30. What would be the result if a class extends two interfaces and both have a method with same name and signature?

1. Runtime error
2. Compile time error
3. Code runs successfully
4. First called method is executed successfully

31. At line number 2 below, choose 3 valid data-type attributes/qualifiers among “final, static, native, public, private, abstract, protected”

public interface Status

{

/\* insert qualifier here \*/ int MY\_VALUE = 10;

}

1. final, native, private
2. final, static, protected
3. final, private, abstract
4. final, static, public

32. Which of these data types is used to store command line arguments?

1. Array
2. Stack
3. String
4. Integer

33. Which of these class is used to create an object whose character sequence is mutable?

1. String()
2. StringBuffer()
3. String() & StringBuffer()
4. None of the mentioned

34. What would be behavior if the constructor has a return type?  
  
 a. Compilation error  
 b. Runtime error

c. Compilation and runs successfully  
 d. Only String return type is allowed

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*