**FINALLL**

1. Two Threads A and B hit two different synchronized methods in an object at the same time. Will they both continue?

Which of the following option gives the valid outcome for the above question?

1. Yes, because the methods are synchronized
2. No, only one method can acquire the lock
3. No, because the methods are synchronized
4. Yes, two different threads hit
5. Which of the following option gives the super type of all Annotation types
6. Java.lang.annotation.Annotation
7. Java.lang.Annotation
8. Java.annotation.Annotation
9. There is no super type for an annotation type.
10. Which of the following options give the valid methods that can be used for executing DDL statements?(Choose 2)
11. execute()
12. executeQuery()
13. executeResult()
14. executeUpdate()
15. Which of the following statement is true regarding constructors?

Abstract classes cannot have constructors

Can be overloaded across inherited classes

Default constructors are optional only for the class that does not have constructors

Default constructors are optional for all classes

1. Which of the following statement is true regarding parameterized constructor?

Parameterized constructors cannot call he default constructor

Parameterized constructors cannot accept final argument as parameters

Parameterized constructors cannot accept its same class type parameters

Parameterized constructors can accept variable argument as their parameters

1. A. Employee() {}

B. public Employee () {}

C. private Employee () {}

Statements

1. No-argument constructor, that does not allow instantiation from within the package
2. No-argument constructor, that does not allow instantiation from outside the package
3. No-argument constructor

A-II, B-III, C-I

A-III, B-I, C-II

A-II, B-I, C-III

A-I, B-II, C-III

1. HashMap is a Collection class.

TRUE

FALSE

1. Which of the following statement gives the significance of overriding equals() method in user defined classes?

Comparing creation time of two objects

Comparing object IDs of two objects

Comparing contents of two objects

Comparing memory reference of two objects

1. Which of the following statements is false about for-each loop in JAVA?

For-each loop does the automatic typecasting

For-each loop is an alternative to Enumeration

For-each loop is an alternative to Iterator

For-each loop can work only wit generic Collections

1. 1)Vector

2) HashSet

3) TreeSet

4) ArrayList

A) It is not ordered or sorted

B) It compass and stores the data in sorting order

C) Permits all elements including NULL

D) It tries to optimize storage management by maintaining a capacity and a capacity increment

1. 2-A, 4-B, 3-C, 1-D
2. 2-A, 3-B, 4-C, 1-D
3. 1-A, 2-B, 3-C, 4-D
4. 1-A, 3-B, 4-C, 2-D
5. Which of the following are true regarding RuntimeException?(Choose 2)
6. RuntimeException can be handled using a catch that handles error
7. Any class that derives the RutimeException will always be an unchecked exception
8. RuntimeException does not require a throws declaration
9. If RuntimeException is declared using throws clause, then the calling method should handle it using try-catch block
10. A) Differing by signatures

B) Code that executes before main() method

C) Code that executes before constructor

I. Instant block

II.Method Overloading

III.Static block

Ans :A-II, B-III, C-I

1. Separation of object interactions from classes and inheritance into distinct layers of abstraction is called as
2. Abstraction
3. Cohesion
4. Composition
5. Decoupling
6. Assume that there is a package structure as follows: com.testpack

Which contains a class called TestPack, with some static and non-static methods and a static inner class?

Which of the following options give the valid import statements?(Choose 2)

1. Import com.testpack.TestPack;
2. Import com.testpack;
3. Import com.testpack.TestPack.\*;
4. Import static com.testpack.TestPack;
5. 1)long test = 045;

2) float f = -12;

3) int value = (int) rue

4) double d = 0x12345678;

5) short s = 20;

Which of the following option gives the legal assignments?

1. 1, 2, 4, 5
2. 1, 2,3,4,5
3. 1,2,3,4
4. 1,3,4,2
5. Set of services provided by a component or by a class is called \_\_\_\_\_
6. Interfaces
7. Objects
8. Components
9. Containers
10. “ Shyam has a best friend who is a Tree” :

Which of the following option represents the above relationship correctly?

1. Class Shyam extends Tree {}
2. Class Shyam {private BestFriend Tree ;}
3. Class Shyam {private Tree bestFriend ;}
4. Class Shyam implements Tree { }
5. Which of the following statement is true?
6. Has-a relationships always rely on instance variables
7. Has-a relationships always requires at least two class types
8. Has-a relationships always rely on inheritance
9. Has-a relationships always rely on polymorphism
10. interface A { void main(String[] args); }

interface B {publicvoid main(String[] args); }

interface C {public static void main(String[] args); }

interface D {protected void main(String[] args); }

interface E {private void main(String[] args); }

Which of the following option gives the valid interface declaration that will compile successfully?

1. Interface A,B,C
2. Interface B,C,D
3. Interface B,C,D
4. Interface A,B
5. interface iOne

{ inti=10;}

interfaceiTwo

{inti=10;}

public class TestInterface implements iOne,iTwo{

public static void main(String[] args) {

System.out.println(i); }}

Which of the following option gives the valid output for the above code ?

1. 0
2. Compilation error at line no. 7
3. Runtime Error at line no. 7
4. 10
5. Which of the following gives the use of CLASSPATH?
6. Holds the location of Java Extension Library
7. Holds the location of User defined classes, packages and JARs
8. Holds the location of Java Software
9. Holds the location of core Java class library (Bootstrap classes)
10. Public class TestOverloading{

int \_length(String s) {

returns.length();

}

float \_length(String s) {

return (float)s.length();

}

}

Which of the following statement is true regarding the above code?

1. Both the length () methods are duplicate methods
2. Both the length () methods are overloaded methods
3. Overloaded methods cannot start with a special character like ‘\_’
4. Overloaded methods should be declared as public
5. Which of the following is true about StringBuffer class?
6. StringBuffer can be extended since it is mutable
7. StringBuffer is a mutable class
8. StringBuffer is a sub class of String class
9. StringBuffer is a Wrapper to the existing String class
10. Class Smile {

Protected joy h;

}

Class Happy extends Smile { }

Class joy { }

Which of the following statement is correct regarding the above given code?

1. Happy is-a joy and has-a Smile
2. Joy has-a Happy and Happy is-a Smile
3. Happy has-a joy and Happy is-a Smile
4. Smile is-a Happy and has-a joy
5. Which of the following option can be matched against the example ‘Television’?
6. Inheritance
7. Encapsulation
8. Polymorphism
9. Abstraction
10. A. a method declared as final

B. a method declared as abstract

C. a method declared as private

I. will not be available in sub classes

II. will den overriding the method

III. will not allow instantiating the class

Which of the following option gives the exact matches of above listed items and statements?

1. A-II, B-III, C-I
2. A-III, B-II, C-I
3. A-II, B-I, C-III
4. A-I, B-II, C-III
5. Public class Eatable {

Eatable () {system.out.print…..

….

…

Void make chocolate (){

..

}}

Which of the following gives the valid output for the above code?

1. Eat Chocolate enjoy
2. Compilation error
3. Runtime error
4. Chocolate enjoy
5. JDBC API allows to connect more than one database from a Java Application simultaneously

TRUE

FALSE

1. Which of the following is true about finalize method ?
2. Finalize will run when an object becomes unreachable
3. Finalize allows a programmer to free memory allocated to an project
4. Finalize may run before or after an object is garbage collected
5. Finalize will always run before an object is garbage collected
6. Which of the following are true regarding try-catch-finally? (Choose 2)
7. A catch block can have another try block nested inside
8. An exception which is not handled by a catch block will be handled by subsequent catch blocks
9. Finally block cannot have a try block with multiple catch blocks
10. An exception which is not handled by a catch block cab be handled by writing another try catch block inside finally block
11. static

{

try

{

Class.forName ("oracle.jdbc.OracleDriver");

}

Catch (ClassNotFoundException cnf)

{

System.out.println ("Driver Not Found");

}

}

}

Which of the following is true ?

It loads OracleDriver class, instantiates it and registers it with DriverManager class

1. Constructors can be declared as private

TRUE

FALSE

1. Protected methods can be accessed from outside the package

TRUE

FALSE

1. Which of the following is true about packages?

Packages can contain both classes and interfaces

1. A...Every floating point literal is implicitly a double, not a float.

B...In the declaration byte b=120; int literal 120 is implicitly converted to byte.

Both A and B are TRUE

1. A…Anonymous inner class can extend a class and implement an interface at the same time.

B…Anonymous class can have their own members.

Both A and B are TRUE

B is TRUE, A is FALSE

1. A private method can be declared as abstract.

TRUE

FALSE

1. Which of the following option gives the valid collection implementation class that implements the List interface and also provides the additional methods to get, add and remove elements from the head and tail of the list without specifying an index?
2. LinkedList
3. ArrayList
4. List
5. Collection
6. Which of the following are true? (Choose 2)
7. Final modifier is not applicable for abstract classes
8. Private members of a class are not accessible through its object instance
9. Package level members in a public class will be available to the inherited class outside package
10. Static modifier is applicable only for inner classes
11. An annotation type can extend another annotation type

TRUE

FALSE

1. Which of the following is true regarding the throw declarations for overriden methods?
2. The overriding method cannot declare additional exception which is not declared in super class version
3. The overriding methods cannot declare to throw the Super class types of those exception declared in the super class methods
4. The overriding method cannot re-declare the unchecked exception, that are declared by super class method
5. A…wait, notify, notifyAll methods are not called on Thread,they are called on object.

B…these methods can only be called from synchronized code, or an IllegalMonitorStateException will be thrown.

Both A and B are true

1. Int x=100, y=200;

Integer i=100, j=200;

Which of the following is true ?(Choose 3)

1. Expression (x<y) and (i<j) are functionally same.
2. Expression (j-i) evaluates another integer wrapper type object with value 100
3. Expression (x==y) and (i==j) are functionally same
4. Expression (x!=j) and (i!=y) evaluates to the same result
5. Declaring a volatile Java variable means, the value of this variable will never be cached thread-locally: all read and writes will go straight to “main memory”; TRUE/FALSE

TRUE

FALSE

1. Which of the following are invalid names in Java? (Choose 2)
2. $char
3. \_int
4. 1MyNumber
5. Case
6. Which of the following statement is true?
7. FileInputStream cannot read text files; you can only use it to read image files
8. When an object is serialized, the whole object state (all the static and non static data variables with their values) are saved
9. When an object is serialized; the whole class definition is saved
10. None of the listed options
11. Consider the following code:

1 Public class FinallyCatch {

2 Public static void main(String args[]){

3 Try{

4 Throw new java.io.IOException();

5 }

6 }

7 }

Which of the following is true regarding the above code?

1. Shows unhandled exception type IOException at line number 5
2. Demands a finally block at line number 4
3. Shows unhandled exception type IOException at line number 4
4. Demands a finally block at line number 5
5. which of the following statements are true? (Choose three)

a. readers and writers are used for i/o on 16-bit Unicode characters

b. FileInputStream and FileOutputStream can be used to read image file

c. FileInputStream and FileOutputSream can be used to read text files

d. FileInputStream and FileOutputSream can be used to handle i/o on 16-bit Unicode characters

**CODES EXECUTED**

1.

public class LogicTest {

public static void main (String[] args) {

int i=5;

int j=10;

int k=15;

if((i<j)||(k-- >j))

{

System.out.println(" First if, value of k:"+k);

}

if((i<j)&&(--k<j))

{

System.out.println(" Second if, value of k:"+k);

}

System.out.println(" out of if, k:"+k);

}

}

Run: First if, value of k:15

out of if, k:14

2.

class Cognizant {

public static void main(String[] args) {

foreach: for(int j=0; j<5; j++)

{

for(int k=0; k<3; k++)

{

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

if(j==3 && k==1)

{

break foreach;

}

if(j==0||j==2)

{

break;

}

}

}

}

}

Run: 0 1 1 1 2 3 3

3.

class SuperClass

{

SuperClass() {}

SuperClass(int i)

{

System.out.println("The value of i is"+i);

}

}

class SubClass extends SuperClass

{

public SubClass(int j) {

System.out.println("The value of j is"+j);

super(j);

}

}

class Test {

public static void main(String[] args) {

SubClass sub= new SubClass(5);

}

}

Compilation Fails (call to super() must be the first line in the block)

4.

public class javac {

public static void main(String[] args) {

for(String str:args)

{ System.out.println(str) }

}}

At command line : java javac java java javac

Ans: java java javac

5.

import java.util.ArrayList;

import java.util.Iterator;

public class Bit9 {

public static void main(String[] args) {

ArrayList<Integer> list= new ArrayList<Integer>();

list.add(new Integer(1));

list.add(new Integer(2));

list.add(new Integer(3));

Iterator<Integer> itr=list.iterator();

for(Integer wij:list)

{ System.out.println("number:"+wij) }

}

}

Run: number:1

number:2

number:3

6.

public class Bit13

{

public static void main(String[] args) {

String str="null";

if(str==null){ System.out.println("null"); }

else(str.length()==0){ System.out.println("zero"); }

else{ System.out.println("some"); }

}

}

Compilation Fails

7.

public class MyOuterClass {

public static class MyNestedClass{

}

}

public class Bit12 {

public static void main(String[] args) {

MyOuterClass.MyNestedClass mn=new MyNestedClass(); // compilation fialed

MyOuterClass mo=new MyOuterClass(); MyOuterClass.MyNestedClass mn1=mo.new MyNestedClass(); //compilation failed

MyOuterClass.MyNestedClass mn2= new MyOuterClass.MyNestedClass(); //correct syntax

MyNestedClass mn=new MyOuterClass.MyNestedClass();// compilation failed

}

}

MyOuterClass.MyNestedClass mn2= new MyOuterClass.MyNestedClass();

8.

public class TestOne {

public static void main(String[] args) {

Thread.sleep(3000);

System.out.println("sleep");

}

}

Prints sleep

9.

class Cognizant{

static void alpha(){/\*more code here\*/}

void beta(){/\*more code here\*/}

} (choose 2)

Cognizant.alpha() is a valid invoation of alpha()

Method beta() can directly call method alpha()

10.

interface Cognizant{}

class Alpha implements Cognizant{}

class Beta extends Alpha{}

public class Delta extends Beta{

public static void main(String[] args) {

Beta x=new Beta();

//insert code here

}}

}

}

Options : which will cause classcastexception at //insert code here line?

1. Alpha a=x;
2. Cognizant f=(Delta)x;
3. Beta b=(Beta)(Alpha)x;
4. Cognizant g=(Alpha)x;

Run: Cognizant f=(Delta)x; causes java.lang.ClassCastException

11.

public class Bit24 {

public static void main(String[] args) {

HashMap press=new HashMap();

press.put("key1","some value");

press.put("key2","some other value");

press.put("key3","yet another value");

Set s=press.keySet();

///HEREEEE

s=new SortedSet(); is abstract can't be instantiated

s=new TreeSet(s);

Collections.sort(s); there is no sort() in Collections

Arrays.sort(s); -do-

}

}

ANS: S=new TreeSet(s);

12.

public class Question4 {

public static void main(String[] args) {

int[] x={0,1,2,3,4};

try{

System.out.println("x[6]:"+x[6]);

System.out.println("x[3]:"+x[3]);

}

catch(IndexOutOfBoundsException ie)

{

System.out.println("Some kind of index out of bound!");

}

catch(ArrayIndexOutOfBoundsException oe)

{

System.out.println("Array Index Out fo Bound");

}

finally{

System.out.println("finally must be executed");

}

System.out.println("x[0]:"+x[0]);

}

}

Compiler error occurs ( same type of exception is handled twice)

13.

public class Example1 {

public static void main(String[] args) {

try

{

int i1=3/0;

}

catch(Exception e)

{

System.out.println("Exception1");

}

catch(NullPointerException e)

{

System.out.println("Exception2");

}

finally{

System.out.println("finally");

}

}

}

Compile time error (same as above)

14.

public class ThreadOrder {

static int count=0;

public static void main(String[] args) {

Counter ct=new Counter();

Tracker trk1=new Tracker(ct,"Thread one");

Tracker trk2=new Tracker(ct,"Thread two");

trk1.start();

trk2.start();

}

}

class Tracker extends Thread{

Counter ct;

String message;

Tracker(Counter ct, String msg){

this.ct=ct;

message=msg;

}

public void run(){

System.out.println(message);

}

}

class Counter

{

private int count=0;

public int headCounter(){

synchronized(this){

count++;

return count;

}

}

}

Run: Thread one

Thread two

15.

public class MyThreadTester {

public static void main(String[] args) {

Counter t1=new Counter();

Counter t2=new Counter();

t1.start();

t2.start();

}

}

class Counter extends Thread{

void run()

{

System.out.println("Running");

}

}

Compilation error ( run() method must be overridden with public)

16.

enum Villages{Pharwala, Gohawar, Phagwara, Goraya}

public class MyEnumTest {

public static enum Colors(RED, BLUE, GREEN, YELLOW, ORANGE);

private enum weekend(Saturday, Sunday);

public static void main(String[] args) {

enum Currency(Dollars, Rupees, Franc, Euro);

System.out.println("hello");

}

}

Compilation fails ( all the enums declaration is wrong except the first one Villages)

**BACK TO FINALLL**

Which of the following Collection can you use to store key value pairs and is thread safe?

 Vector

 Hashmap

 **Hashtable**

 TreeMap