**Q1**

What is the output of the following program?  
  
class A {  
  private void print() {   
    System.out.println("Inside A's print method.");   
  }     
  public void test() {   
    this.print();   
  }   
}   
class B extends A {   
  private void print() {   
    System.out.println("Inside B's print method.");   
  }             
  public static void main(String args[]) {   
    A a = new B();   
    a.test();   
  }   
}

*Single Choice - Select one correct answer from the options list.*

Inside A's print method.

Inside B's print method.

Inside A's print method.   
     Inside B's print method.

Compilation error

**Q2**

"Books has pages" is an example of \_\_\_\_\_\_\_\_\_\_.

*Single Choice - Select one correct answer from the options list.*

Generalization

composition

aggregation

None of the above

**Q3**

Which annotation can be used to carry out cleanup activities and to execute the method   
after all tests have finished ?

*Single Choice - Select one correct answer from the options list.*

@After

@Before

@AfterClass

@BeforeClass

**Q4**

Suppose you want to test object response for not null.   
Which of the following code would you use in a test to verify than an object is not null?

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

assertNotNull("Response is null", response)

assertNotNull(response)

if(response = = null) {throw newExcpetion("The response is null");}

assertNull("Response is null", response)

**Q5**

Given:  
  
class CardBoard {  
Short story = 5;  
CardBoard go(CardBoard cb) {  
cb = null;  
return cb;  
}  
public static void main(String[] args) {  
CardBoard c1 = new CardBoard();  
CardBoard c2 = new CardBoard();  
CardBoard c3 = c1.go(c2);  
c1 = null;  
// do Stuff  
} }  
  
When // doStuff is reached, how many CardBoard objects are eligible for GC?

*Single Choice - Select one correct answer from the options list.*

0

1

2

None of the above

**Q6**

Given:  
  
1. class Crivitch {  
2. public static void main(String [] args) {  
3. int x = 0;  
4. // insert code here  
5. do { } while (x++ < y);  
6. System.out.println(x);  
7. }  
8. }  
  
Which statement, inserted at line 4, produces the output 12?

*Single Choice - Select one correct answer from the options list.*

int y = 10;

int y = 11;

int y = 12;

int y = 13;

**Q7**

Given:  
  
class Mixer {  
Mixer() { }  
Mixer(Mixer m) { m1 = m; }  
Mixer m1;  
public static void main(String[] args) {  
Mixer m2 = new Mixer();  
Mixer m3 = new Mixer(m2); m3.go();  
Mixer m4 = m3.m1; m4.go();  
Mixer m5 = m2.m1; m5.go();  
}  
void go() { System.out.print("hi "); }  
}  
  
What is the result?

*Single Choice - Select one correct answer from the options list.*

hi hi

hi hi hi

Compilation fails

hi hi, followed by an exception

**Q8**

Integer i = new Integer(10);  
int i2 = 10;  
System.out.println( i == i2 );  
  
What will be the output of the above code fragment?

*Single Choice - Select one correct answer from the options list.*

Will print true

Compilation Error

CastException

Runtime Exception

Will print false

**Q9**

class ArrayTest {  
    public static void main(String args[]) {  
      int arr[] = new int[2];  
      System.out.println(arr[0]);  
    }  
}  
  
What will be the result of compiling and executing the above code?

*Single Choice - Select one correct answer from the options list.*

The program does not compile because arr[0] is being read before being initialized.

The program generates a runtime exception because arr[0] is being read before being initialized.

The program compiles and prints 0 when executed

The program compiles and prints 1 when executed.

The program compiles and runs but the results are not predictable because of un-initialized   
memory being read

**Q10**

A system integrator needs to know the scalability of the system. Which UML building block will be helpful   
for examining the same?

*Single Choice - Select one correct answer from the options list.*

Structural view

Behavioral View

Implementation View

Environment View

**Q11**

When testing an error flow, what test would you write?

*Single Choice - Select one correct answer from the options list.*

public void testErrorCase() throws Exception  
     {  
     methodToTest();   
     // Asserts here }

public void testErrorCase(){ try{   
     methodToTest();   
     }  
     catch (Exception expected)  
     {   
     // Asserts her   
     }   
     }

public void testErrorCase(){   
     try{   
     methodToTest();    
     fail("Should not reach this point")  
     catch (Exception expected)  
     {    
     // Asserts here   
     }   
     }

None of the above

**Q12**

@FunctionalInterface  
public interface MyInterface {  
 void checkName();  
 void checkSalary();  
}  
  
Smith has created the above interface, but he is getting some compilation error.  
  
Which of the following options can be used to correct the error?    

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

Functional Interface should not contain more than one method so one method should be remove   
to resolve compilation error

Functional Interface should not contain more than one method so if Smith does not want to use this interface   
for Lambda Expression then FunctionalInterface annotation should be removed to resolve the compilation error

Functional Interface should not contain method with void return type so method type should be changed   
to resolve the compilation error

Functional Interface must contain a default static method so default static method should be included   
to resolve the compilation error

**Q13**

Consider Question and answer are modelled as classes. A single Question may have Single answers.  
How do you define this relationship ?

*Single Choice - Select one correct answer from the options list.*

Aggregation

Composition

Both the above

None of the above

**Q14**

Read the following example. Which of the following options relates to this example?  
  
"Customer needs to know what is the interest he is earning; and may not need to know how the bank is calculating this interest."

*Single Choice - Select one correct answer from the options list.*

Encapsulation

Inheritance

Polymorphism

Abstraction

**Q15**

\_\_\_\_\_\_ is the relationship between an “interface” and the “class” that provides the interface's services.

*Single Choice - Select one correct answer from the options list.*

Visualization

Realization

association

composite

**Q16**

Given:  
  
class Alpha{  
   Alpha doStuff(char c){  
     return new Alpha();  
   }  
}  
  
class Beta extends Alpha{  
  Beta doStuff(char c){  
    return new Beta();  
  }  
}  
  
Is it a valid overriding?

*Single Choice - Select one correct answer from the options list.*

True

False

**Q17**

Which method puts the current thread on wait until the thread on which it's called is dead?

*Single Choice - Select one correct answer from the options list.*

sleep()

join()

start()

stop()

**Q18**

Consider the following entries in the log4j property file:  
  
log4j.appender.testAppender=org.apache.log4j.ConsoleAppender  
log4j.appender.testAppender.layout=org.apache.log4j.PatternLayout  
log4j.appender.testAppender.layout.conversionPattern=%m%n  
  
Which of the following statements is true about the above entries?

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

set the appender named testAppender to be a console appender

set the name of Appender to ConsoleAppender

the layout of ConsoleAppender is set to PatternLayout

set the layout for the appender testAppender to PatternLayout

**Q19**

If Single files are to be read one after the other simultaneously, the appropriate class to be used is:

*Single Choice - Select one correct answer from the options list.*

SequenceInputStream

RandomAccessFile

FileReader

FileInputStream

DataInputStream

**Q20**

What is the result of trying to compile and run the following program?  
  
 class  Phone implements Cloneable{  
public static void main(String[] args) {  
  Phone p= new Phone();  
  if (p instanceof Object)  
     System.out.println("Object");  
if (p instanceof Cloneable)  
     System.out.println("Cloneable");  
}  
}

*Single Choice - Select one correct answer from the options list.*

The program does not compile

The program compiles and runs and writes Object to the standard ouput

The program compiles and runs and writes both Object and Cloneable to the standard ouput.

The program compiles and runs and writes Cloneable to the standard ouput.

**Q21**

What will be the output of the following program?  
  
import java.util.TreeSet;  
public class Employee {  
  private int empno;  
  private String name;  
  
  public static void main(String args[]) {  
    TreeSet empset = new TreeSet();  
    empset.add(new Employee());  
    empset.add(new Employee());  
    System.out.println("Size: " + empset.size());  
  }  
}

*Single Choice - Select one correct answer from the options list.*

Compilation Error

Runtime Exception

Size: 1

Size: 2

**Q22**

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:

*Single Choice - Select one correct answer from the options list.*

throw Exception

throws Exception

new Exception

No need to specify anything

**Q23**

Consider the following code which is used to print a list of integers stored in "numbers" list.   
  
numbers.forEach((Integer value) -> System.out.println(value));  
  
Which one of the following built-in interface referenced by above lambda expression?

*Single Choice - Select one correct answer from the options list.*

Function<Integer>

Consumer<Integer>

Supplier<Integer>

Predicate<Integer>

**Q24**

What will be the output of the following code snippet?  
  
String str = new String("PACE");  
String str1 = str;  
str=null;  
System.out.println("Length of the string is :" + str1.length());

*Single Choice - Select one correct answer from the options list.*

4

3

0

It will generate NullPointerException

**Q25**

Given:  
  
public interface SuperInterface {....................}  
public interface SubInterface extends SuperInterface {...........}  
public class MyClass implements SuperInterface {.............}  
MyClass obj = new MyClass();  
  
With how many types, reference obj is compatible with including the default type "Object"?

*Single Choice - Select one correct answer from the options list.*

0

1

2

3

**Q26**

Given:  
  
class A {.......}  
class B extends A {.......}  
class C extends A {.......}  
  
What will be the output of the following code fragment'?  
  
A a1 = new B();//Line 1  
A a2 = new C();//Line 2  
B b1 = (B)a1;// Line 3  
B b2 = (B)a2;//Line 4

*Single Choice - Select one correct answer from the options list.*

Compile Time Error at Line 1

Compile Time Error at Line 3

Runtime Exception at Line 3

Runtime Exception at Line 4

**Q27**

Saratha is involved in the development of a Car sales purchase application.   
She created different packages named as com.carsales.service, com.carsales.dao, com.carsales.dto   
and com.carsales.mainclass. She created a class named "CarSales" in a package " com.carsales.dto"   
which consists of attributes such as carId, carName, and price and appropriate setters and getters.   
She also created a class named "CarSalesDAO" in a package " com.carsales.dao".   
Implementation of which statement allows to access attributes of "CarSales"  class from a   
"com.carsales.dto" package in a "CarSalesDAO" class available in a  "com.carsales.dao" package?

*Single Choice - Select one correct answer from the options list.*

Assigning attributes and a class with public.

Assigning private modifier to attributes and public modifier to class and also inheriting  CarSales class in   
CarSalesDAO class will helpful for performing this task.

Assigning public modifier only to attributes.

Assigning public modifier to class and private modifier to attributes and also importing CarSales class   
in CarSalesDAO class will helpful for performing this task.

**Q28**

Choose the correct lambda code to check the given value is divisible by 5.

*Single Choice - Select one correct answer from the options list.*

Predicate<Integer> divisble=(number)->number / 5 == 0;  
 System.out.println(divisble.test(11));

Predicate<int> divisble=(number)->number / 5 == 0;  
 System.out.println(divisble.test(11));

Predicate<Integer> divisble=(number)->number % 5 == 0;  
 System.out.println(divisble.test(11));

Predicate<int> divisble=(number)->number % 5 == 0;  
 System.out.println(divisble.test(11));

**Q29**

What will be the output of the following code :  
  
class Base  
{  
public void amethod()throws FileNotFoundException{}  
}  
public class ExcepDemo extends Base  
{  
   public static void main(String argv[])  
      {  
         ExcepDemo e = new ExcepDemo();  
      }  
   public void amethod(){}   
   protected ExcepDemo()  
   {  
      try  
       {  
          DataInputStream din = new    
             DataInputStream(System.in);   
          System.out.println("Pausing");   
          din.readByte();  
          System.out.println("Continuing");   
          this.amethod();  
       }  
       catch(IOException ioe) {}  
   }  
}

*Single Choice - Select one correct answer from the options list.*

Compile time error caused by protected constructor

Compile and run with output of "Pausing" and "Continuing" after a key is hit

Compile time error caused by amethod not declaring Exception

Runtime error caused by amethod not declaring Exception

**Q30**

What will be the output of the following code fragment?  
  
Vector v1 = new Vector(7, 3);  
 for(int num = 1;num<=15;num++)  
  v1.add(num);  
 System.out.println(v1.capacity());

*Single Choice - Select one correct answer from the options list.*

21

15

18

16

**Q31**

Which of the following line of code is suitable to start a thread ?  
  
class X implements Runnable   
{   
    public static void main(String args[])   
    {  
        /\* Missing code? \*/  
    }   
    public void run() {}   
}

*Single Choice - Select one correct answer from the options list.*

Thread thread = new Thread(MyThread);

Thread t = new Thread(MyThread);  
t.start();

MyThread run = new MyThread(); Thread t = new Thread(run); t.start();

Thread thread = new Thread();   
thread.run();

**Q32**

Which relationship enables factoring out infrequent behavior or error conditions?

*Single Choice - Select one correct answer from the options list.*

Include

exclude

extend

intend

**Q33**

Given:  
new FileWriter("report");  
  
What is the effect on the file "report" assuming that the file already exists?

*Single Choice - Select one correct answer from the options list.*

The data is appended to the file

An exception is raised as the file already exists

The file is replaced with a new file

The data is written to random locations within the file

**Q34**

What will be the output of the following code fragment?  
  
HashMap map = new HashMap();  
  map.put("obj1", null);  
  map.put("obj2", null);  
  map.put(null, null);  
  map.put("obj3", "obj3");  
  System.out.println(map.size());

*Single Choice - Select one correct answer from the options list.*

3

1

4

2

**Q35**

"Library contains books" is an example of \_\_\_\_\_\_\_\_\_\_\_\_.

*Single Choice - Select one correct answer from the options list.*

Generalization

composition

aggregation

None of the above

**Q36**

What Happens If a JUnit Test Method Is Declared as "private"?

*Single Choice - Select one correct answer from the options list.*

Compilation Error

No Error during compliation

Run Time Error

None of the above

**Q37**

public class MethodOverriding  
{  
  private void add(int operand1, int operand2)  
  {  
     System.out.println(operand1 + operand2);  
  }  
}  
public class Overridden extends MethodOverriding  
{  
  public void show()  
  {  
     add(10, 12);  
  }  
   public static void main(String args[])  
   {  
      Overridden ob = new Overridden();  
      ob.show();  
   }  
}  
  
What will be the output of above code when compiled and executed?

*Single Choice - Select one correct answer from the options list.*

Compile time error

Runtime error as add method is not defined in MethodOverriding class

Will compile and display 32

Will compile and dislay 1012

**Q38**

In Java, what happens if you try to compile code that looks like this:  
  
class MyString extends String {  
}

*Single Choice - Select one correct answer from the options list.*

The code compiles successfully.

The code does not compile because you have not defined a main() method.

The code does not compile because the String class is abstract.

The code does not compile because the String class is final.

**Q39**

The Junit framework automatically invokes the \_\_\_\_\_\_\_\_\_\_\_\_ annotated method   
after each test is run.

*Single Choice - Select one correct answer from the options list.*

@assertEquals

@tearDown

@After

@Before

**Q40**

Which diagrams define the static architecture of a model. They are used to model the “things”   
that make up a model. They are used to model the relationships and dependencies?

*Single Choice - Select one correct answer from the options list.*

Structural view

Behavioral view

Implementation view

Environment view

**Q41**

Read the code given below and identify the logger statements which are not executed.  
  
   Logger  logger = Logger.getLogger("com.foo");  
   logger.setLevel(Level.INFO);  
   Logger barlogger = Logger.getLogger("com.foo.Bar");  
   logger.warn("Low fuel level.");  
   logger.debug("Starting search for nearest gas station.");  
   barlogger.info("Located nearest gas station.");  
   barlogger.debug("Exiting gas station search");

*Single Choice - Select one correct answer from the options list.*

logger.debug("Starting search for nearest gas station.");

barlogger.debug("Exiting gas station search");

Both Option A and B

All are disabled

**Q42**

Here is log4j.properties file:  
  
1. log4j.rootLogger=ERROR, CA  
2. log4j.appender.CA = org.apache.log4j.ConsoleAppender  
3. log4j.appender.CA.layout=org.apache.log4j.HTMLLayout  
  
What will be the result of attempting to compile and run the following program ?  
  
public static void main(String[] args) {  
  PropertyConfigurator.configure("log4j.properties");  
logger.debug("Here goes the debug message");  
logger.error("Here goes the error message");  
logger.fatal("Here goes the fatal message");  
logger.info("Here goes the info message");  
logger.warn("Here goes the warn message");  
   }  
}

*Single Choice - Select one correct answer from the options list.*

Here goes the debug message  
Here goes the error message  
Here goes the fatal message  
Here goes the info message  
Here goes the warn message

Here goes the debug message  
Here goes the error message  
Here goes the fatal message

Here goes the error message  
Here goes the fatal message  
Here goes the info message

Here goes the error message  
Here goes the fatal message

Here goes the info message  
Here goes the error message  
Here goes the fatal message

Here goes the info message  
Here goes the warn message  
Here goes the error message  
Here goes the fatal message

**Q43**

A setUp() and tearDown() code can be run once for all of the tests by:

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

Adding a @BeforeClass annotation to a method to be run before all tests in a class

Adding a @AfterClass annotation to a method to be run after all tests in a class

This can not be done

setUp() and tearDown() will be called only once for all the tests

**Q44**

Select the false statement about Lambda Expressions:

*Single Choice - Select one correct answer from the options list.*

Increases the Lines of code

Provides sequential and parallel execution support

Allows to pass behaviors into methods

Higher efficiency with laziness

**Q45**

Consider the below code of an anonymous class:  
  
Comparator<Integer> intComparator = new Comparator<Integer>() {  
       @Override  
       public int compare(Integer o1, Integer o2) {  
              return o1.compareTo(o2);  
       }  
};  
Which of the following are valid lambda expressions to replace this anonymous class?

*Multiple Choice - This may have multiple correct answers. Select required answer(s) from the options list.*

(o1,o2) -> o1.compareTo(o2)

(o1,o2)->return o1.compareTo(o2)

(Integer o1, Integer o2)->o1.compareTo(o2)

(int o1, int o2)->o1.compareTo(o2)