1.

What will be the output of the following class:

```
public class TestClass
{
        public void testRefs(String str, StringBuffer sb)
        {
        str = str + sb.toString();
        sb.append(str);
        str = null;
        sb = null;
        }
        public static void main(String[] args)
        {
        String s = "aaa";
        StringBuffer sb = new StringBuffer("bbb");
        new TestClass().testRefs(s, sb);
        System.out.println("s="+s+" sb="+sb);
        }
}
```

Select 1 correct option.

a s=aaa sb=bbb

b s=null sb=null

c s=aaa sb=null

d s=null sb=bbbaaa

e s=aaa sb=bbbaaabbb

2.What will be the output of the following lines ?

```
System.out.println("" +5 + 6);   //1
 System.out.println(5 + "" +6);   // 2
 System.out.println(5 + 6 +"");   // 3
 System.out.println(5 + 6);       // 4
```

Select 1 correct option.

a 56, 56, 11, 11

b 11, 56, 11, 11

c 56, 56, 56, 11

d 56, 56, 56, 56

e 56, 56, 11, 56

3. What will be the result of attempting to compile the following program?

```
public class TestClass
{
   long l1;
   public void TestClass(long pLong) {l1 = pLong ; }  //(1)
   public static void main(String args[])
   {
      TestClass a, b ;
      a = new TestClass();  //(2)
      b = new TestClass(5);  //(3)
   }
}
```

a A compilation error will be encountered at (1)

b A compilation error will be encountered at (2)

c A compilation error will be encountered at (3).

d The program will compile correctly.

e It will not compile because parameter type of the constructor is different than the type of value passed to it.

4.

Carefully examine the following code.

```
public class StaticTest
{
   static
   {
      System.out.println("In static");
   }
   {
      System.out.println("In non - static");
   }
   public static void main(String args[ ])
   {
       StaticTest st1;                    //1
       System.out.println(" 1 ");
       st1 = new StaticTest();            //2
       System.out.println(" 2 ");
       StaticTest st2 = new StaticTest(); //3
   }
}
```

What will be the output?

Select 1 correct option.

a In static, 1, In non - static, 2, In non - static   : in that order.

b Compilation error.

c 1, In static, In non - static, 2, In non - static   : in that order.

d In static, 1, In non - static, 2, In non - static   : in unknown order.

e None of the above.

5.

What will the following code snippet print?

```
int index = 1;
 String[] strArr = new String[5];
 String   myStr  = strArr[index];
 System.out.println(myStr);
```

Select 1 correct option.

A It will print nothing.

B It will print 'null'

C It will throw ArrayIndexOutOfBounds at runtime.

D It will print NullPointerException at runtime.

E None of the above.

6.

Consider the following code:

```
class A
{
   A() {  print();   }
  void print() { System.out.println("A"); }
}
class B extends A
```

```
{
  int i =   Math.round(3.5f);
  public static void main(String[] args)
  {
    A a = new B();
    a.print();
  }
  void print() { System.out.println(i); }
}
```

What will be the output when class B is run ?

Select 1 correct option.

a It will print A, 4.

b It will print A, A

c It will print 0, 4

d It will print 4, 4

e None of the above.

7.

Consider the following class.

```
public class Test
{
   public int id;
}
```

Which of the following is the correct way to make the variable 'id' read only for other classes.

Select 1 correct option.

a Make 'id' private.

b Make 'id' private and provide a public method getId() which will return it's value.

c Make 'id' static and provide a public static method getId() which will return it's value.

d Make id 'protected'

8. What will the following code print?

```
public class Test
{
        public int luckyNumber(int seed)
        {
        if(seed > 10) return seed%10;
        int x = 0;
        try
        {
                if(seed%2 == 0) throw new Exception("No Even no.");
                else return x;
        }
        catch(Exception e)
        {
                return 3;
        }
        finally
        {
                return 7;
        }
        }
        public static void main(String args[])
        {
        int amount = 100, seed = 6;
        switch( new Test().luckyNumber(6) )
        {
                case 3: amount = amount * 2;
                case 7: amount = amount * 2;
                case 6: amount = amount + amount;
                default :
        }
        System.out.println(amount);
        }
}
```

Select the correct option

A. It will not compile.
B. It will throw an exception at runtime.
C. It will print 800
D. It will print 200
E. It will print 400

9. Identify the correct constructs.

Select 1 Option

A.

```
try {
  for( ;; );
}finally { }
```

B.

```
try {
  File f = new File("c:\a.txt");
} catch {  f = null; }
```

C.

```
int k = o
try {
  k = callValidMethod();
}
System.out.println(k);
catch {  k = -1; }
```

D.

```
try {
  try {
     Socket s = new ServerSocket(3030);
  }catch(Exception e) {
    s = new ServerSocket(4040);
  }
}
```

E.

```
try {
      s = new ServerSocket(3030);
} catch(Exception t){ t.printStackTrace(); }
 catch(IOException e) {
    s = new ServerSocket(4040);
} catch(Throwable t){ t.printStackTrace(); }
```

F.

```
int x = validMethod();

try {

  if(x == 5) throw new IOException();

  else if(x == 6) throw new Exception();

}finally {

 x = 8;

}

catch(Exception e){ x = 9; }
```

10

Consider the following code ...

```
class A
{
   public void doA(int k) throws Exception { // 0
      for(int i=0; i< 10; i++) {
         if(i == k) throw new Exception("Index of k is "+i); // 1
      }
   }
   public void doB(boolean f) { //2
      if(f) {
         doA(15); //3
      }
      else return;
   }
   public static void main(String[] args) { //4
      A a = new A();
      a.doB(args.length>0); //5
   }
}
```

Which of the following statements are correct?

Please select 1 option.

A. This will compile and run without any errors or exception.
B. This will compile if 'throws Exception' is added at line //2

C. This will compile if 'throws Exception' is added at line //4
D. This will compile if 'throws Exception' is added at line //2 as well as //4
E. This will compile if line marked // 0 is enclosed in a try - catch block.

11

What is the result of compiling and running this code?

```
class MyException extends Throwable{}

class MyException1 extends MyException{}

class MyException2 extends MyException{}

class MyException3 extends MyException2{}

public class ExceptionTest

{

        void myMethod() throws MyException

        {

        throw new MyException3();

        }

        public static void main(String[] args)

        {

        ExceptionTest et = new ExceptionTest();

        try

        {

                et.myMethod();

        }

        catch(MyException me)

        {

                System.out.println("MyException thrown");

        }
        catch(MyException3 me3)

        {

                System.out.println("MyException3 thrown");

        }

        finally

        {

                System.out.println(" Done");

        }

        }

}
```

Please select 1 option

A. MyException thrown
B. MyException3 thrown
C. MyException thrown Done
D. MyException3 thrown Done
E. It fails to compile

12

Consider the following hierarchy of Exception classes :

```
java.lang.RuntimeException

             +-------- IndexOutOfBoundsException

            +---------ArrayIndexOutOfBoundsException
```

Which of the following statements are correct for a method that can throw ArrayIndexOutOfBounds as well as StringIndexOutOfBounds Exceptions but does not have try catch blocks to catch the same?

Please select 3 options

A. The method calling this method will either have to catch these 2 exceptions or declare them in it's throws clause.
B. It is ok if it declares just 'throws ArrayIndexOutOfBoundsException'

C. It must declare 'throws ArrayIndexOutOfBoundsException, StringIndexOutOfBoundsException'
D. It is ok if it declares just 'throws IndexOutOfBoundsException'
E. It does not need to declare any throws clause.

13.

Consider the following class:

```
public class IntPair

{

    private int a;

    private int b;

    public void setA(int i){ this.a = i; }

    public int getA(){ return this.a; }

    public void setB(int i){ this.b = i; }

    public int getB(int b){ return b; }

    public boolean equals(Object obj)

    {

        return ( obj instanceof IntPair && this.a == ((IntPair) obj).a );

    }

    public int hashCode()

    {

        //1

    }

}
```

Which of the following options would be valid at //1?

Select 4 correct options

a return 0;

b return a;

c return a+b;

d return a*a;

e return a/2;

14.

Given:

1. import java.util.*;

2. public class PQ {

3. public static void main(String[] args) {

4. PriorityQueue<String> pq = new PriorityQueue<String>();

5. pq.add(”carrot”);

6. pq.add(”apple”);

7. pq.add(”banana”);

8. System.out.println(pq.poll() +”:” + pq.peek());

9. }

10. }

What is the result?

A. apple:apple

B. carrot:apple

C. apple:banana

D. banana:apple

E. carrot:carrot

F. carrot:banana

15

Consider the following class:

public class IntPair

```
{
   private int a;
   private int b;
   public void setA(int i){ this.a = i; }
   public int getA(){ return this.a; }
   public void setB(int i){ this.b = i; }
   public int getB(int b){ return b; }
   public boolean equals(Object obj)
   {
      return ( obj instanceof obj && this.a == ((IntPair) obj).a  && this.b == ((IntPair) obj).b );
   }
   public int hashCode()
   {
      //1
   }
}
```

Which of the following options would NOT be valid at //1?

Select 1 correct option.

a return a;

b return a*b;

c return a+b;

d return b;

e None of these is invalid.

16.

Which of the following are valid implementation of equals() method of a class TestClass?

1.

```
public boolean equals(TestClass tc)
{
   return this == tc;
}
```

2.

```
public boolean equals(TestClass tc)
{
   return this != tc;
}
```

3.

```
public boolean equals(Object tc)
{
   return this == tc;
}
```

4.

```
public boolean equals(Object tc)
{
   if( tc instanceof TestClass && this.someVar == ( (TestClass)tc).someVar )
   {
      if(this != tc) return true;
      else return false;
   }
   else return false;
}
```

Select 1 correct option.

a 1

b 2

c 3

d 4

e None of these.

17.

Given:

```
public static Iterator reverse(List list) {
Collections.reverse(list);
return list.iterator();
}
public static void main(String[] args) {
List list = new ArrayList();
list.add("1"); list.add("2"); list.add("3");
for (Object obj: reverse(list))
System.out.print(obj + ", ");
}
```

What is the result?

A. 3, 2, 1,

B. 1, 2, 3,

C. Compilation fails.

D. The code runs with no output.

E. An exception is thrown at runtime.

18.

What will be the result of attempting to compile and run the following program?

```
public class MyThread implements Runnable
{
  String msg = "default";
  public MyThread(String s)
  {
    msg = s;
  }
  public void run( )
  {
    System.out.println(msg);
  }
  public static void main(String args[])
  {
    new Thread(new MyThread("String1")).run();
    new Thread(new MyThread("String2")).run();
    System.out.println("end");
  }
}
```

Select 1 correct option.

a The program will compile and print only 'end'.

b It will always print 'String1' 'String2' and 'end', in that order.

c It will always print 'String1'   'String2' in random order followed by 'end'.

d It will always print 'end' first.

e No order is guaranteed.

19.

The following program will always terminate.

```
class Base extends Thread
{
        static int k = 10;
}
class Incrementor extends Base
{
        public void run()
        {
        for(; k>0; k++)
        {
                System.out.println("IRunning...");
        }
        }
}
class Decrementor extends Base
{
        public void run()
        {
        for(; k>0; k--)
        {
                System.out.println("DRunning...");
        }
        }
}
public class TestClass
{
        public static void main(String args[]) throws Exception
        {
        Incrementor i = new Incrementor();
```

```
            Decrementor d = new Decrementor();
            i.start();
            d.start();
            }
}
```

Select 1 correct option.

a True

b False

20.

```
public class TestClass
{
  static StringBuffer sb1 = new StringBuffer();
  static StringBuffer sb2 = new StringBuffer();
  public static void main(String[] args)
  {
      new Thread
      (
         new Runnable()
         {
            public void run()
            {
                synchronized(sb1)
                {
                    sb1.append("X");
                    synchronized(sb2)
                {
                  sb2.append("Y");
                }
              }
              System.out.println(sb1);
            }
          }
        ).start();
        new Thread
        (
          new Runnable()
          {
            public void run()
            {
              synchronized(sb2)
              {
                sb1.append("Y");
                synchronized(sb1)
                {
                  sb2.append("X");
                }
              }
             System.out.println(sb2);
            }
          }
        ).start();
      }
```

```
}
```

Select 1 correct options

a It will print 'XX' followed by 'YY'

b It will print 'YY' followed by XX'

c It will print 'XY' followed by 'YX'

d The above code may result in a dead lock and so nothing can be said about the output.

21.

Consider the following program...

```
public class TestClass implements Runnable
{
  int x = 5;
  public void run()
  {
    this.x = 10;
  }
  public static void main(String[] args)
  {
     TestClass tc = new TestClass();
     new Thread(tc).start(); // 1
     System.out.println(tc.x);
  }
}
```

What will it print when run?

Select 1 correct option.

a 5

b 10

c It will not compile.

d Exception at Runtime.

e The output cannot be determined.

22.

What happens when a thread executes wait() method on an object without owning the object's lock?

Select 1 correct option.

a It pauses till somebody releases the lock and then it acquires the lock.

b It causes the object to release all it's locks.

c It acquires the lock and proceeds.

d It forces the other thread that owns the lock to release it.

e It throws an exception.

23

Given:

```
1. public class TestOne implements Runnable {
2. public static void main (String[] args) throws Exception {
3. Thread t = new Thread(new TestOne());
4. t.start();
5. System.out.print(”Started”);
6. t.join();
7. System.out.print(”Complete”);
8. }
9. public void run() {
10. for (int i= 0; i< 4; i++) {
```

```
System.out.print(i);
}
}
}
```

What can be a result?

A. Compilation fails.

B. An exception is thrown at runtime.

C. The code executes and prints “StartedComplete”.

D. The code executes and prints “StartedComplete0123”.

E. The code executes and prints “Started0l23Complete”.

24.

Consider the following classes...

```
class Car
{
  public int gearRatio = 8;
  public String accelerate() {  return "Accelerate : Car";  }
}
class SportsCar extends Car
{
  public int gearRatio = 9;
  public String accelerate() {  return  "Accelerate : SportsCar";  }
  public static void main(String[] args)
  {
    Car c = new SportsCar();
    System.out.println( c.gearRatio+"  "+c.accelerate() );
```

```
   }
}
```

What will be printed when SportsCar is run?

Select 1 correct option.

a 8   Accelerate : Car

b 9 Accelerate : Car

c 8 Accelerate : SportsCar

d 9 Accelerate : SportsCar

e None of the above.

25.

Which statements, when inserted at line 1, will cause a runtime exception ?

```
class B {}
class B1 extends B {}
class B2 extends B {}
public class ExtendsTest
{
  public static void main(String args[])
  {
     B b = new B();
     B1 b1 = new B1();
     B2 b2 = new B2();
     // insert statement here
  }
```

Select 1 correct option.

a b = b1;

b b2 = b;

c b1 = (B1) b;

d b2 = (B2) b1;

e b1 = (B) b1;