**Q.** Given:  
  
3. public class Wind {  
4. public static void main(String[] args) {  
5. foreach:  
6. for(int j=0; j<5; j++) {  
7. for(int k=0; k< 3; k++) {  
8. System.out.print(" " + j);  
9. if(j==3 && k==1) break foreach;  
10. if(j==0 || j==2) break;  
11. }  
12. }  
13. }  
14. }  
  
What is the result?

**1)**. 0 1 2 3

**2)**. 1 1 1 3 3

**3)**. **0 1 1 1 2 3 3**

**4)**. 1 1 1 3 3 4 4 4

**5)**. 0 1 1 1 2 3 3 4 4 4

**6)**. Compilation fails

**Solution** :  
option [3] is correct

**Q.** class A{  
A(){  
  System.out.println("a");  
 }  
 A(String a){  
    
  System.out.println("A");  
 }  
}  
class B extends A {  
 B(){  
    
  System.out.println("b");  
 }  
 B(String a){  
  this();  
  System.out.println("B"+a);  
 }  
}  
  
public class Test  {  
    
 public static void go(){  
  new B("C");  
 }  
 public static void main(String [] args) {  
 go();  
 }  
}  
  
Choose the correct one:

**1)**. **a b BC**

**2)**. b BC

**3)**. BC

**4)**. a AC

**5)**. AC

**Solution** :  
option [1] is correct

**Q.** class B {  
 B(){  
  System.out.println("J");  
 }  
 B(String s){  
  System.out.println("B");  
 }  
}  
public class Test extends B  {  
    
 public Test(String s) {   
    
  System.out.print("D");   
  }  
 public Test() {  
  // TODO Auto-generated constructor stub  
  System.out.println("A");  
 }  
 public static void main(String [] args) {  
 new Test("C");  
 System.out.println(" ");  
 }  
}  
  
Choose the correct one:

**1)**. **J D**

**2)**. D

**3)**. A D J B

**4)**. J B A D

**5)**. Compilation error

**Solution** :  
option [1] is correct

**Q.** Given:  
  
3. public class Circles {  
4. public static void main(String[] args) {  
5. int[] ia = {1,3,5,7,9};  
6. for(int x : ia) {  
7. for(int j = 0; j < 3; j++) {  
8. if(x > 4 && x < 8) continue;  
9. System.out.print(" " + x);  
10. if(j == 1) break;  
11. continue;  
12. }  
13. continue;  
14. }  
15. }  
16. }  
  
What is the result?

**1)**. 1 3 9

**2)**. 5 5 7 7

**3)**. 1 3 3 9 9

**4)**. **1 1 3 3 9 9**

**5)**. 1 1 1 3 3 3 9 9 9

**6)**. Compilation fails

**Solution** :  
option [4] is correct

**Q.** Given:  
  
1. public class Frisbee {  
2. // insert code here  
3. int x = 0;  
4. System.out.println(7/x);  
5. }  
6. }  
And given the following four code fragments:  
I. public static void main(String[] args) {  
II. public static void main(String[] args) throws Exception {  
III. public static void main(String[] args) throws IOException {  
IV. public static void main(String[] args) throws RuntimeException {  
  
If the four fragments are inserted independently at line 2, which are true? (Choose all that apply.)

**1)**. All four will compile and execute without exception

**2)**. All four will compile and execute and throw an exception

**3)**. Some, but not all, will compile and execute without exception

**4)**. **Some, but not all, will compile and execute and throw an exception**

**5)**. When considering fragments II, III, and IV, of those that will compile, adding a try/catch block around line 4 will cause compilation to fail

**Solution** :  
option [4] is correct

**Q.** public class Test  {  
 Test() { main("hi"); }  
 public static void main(String[] args) {  
 System.out.print("2 ");  
 }  
 public static void main(String args) {  
 System.out.print("3 " + args);  
 }  
}  
  
What is the result? (Choose all that apply.)

**1)**. **2 will be included in the output**

**2)**. 3 will be included in the output

**3)**. hi will be included in the output

**4)**. Compilation fails

**5)**. An exception is thrown at runtime

**Solution** :  
option [1] is correct

**Q.** class A{  
 String a="A";  
 public String getData(){  
    
  return "Hi";  
 }  
}  
class B extends A {  
 String a="B";  
 public String getData(){  
    
  return "Bye";  
 }  
}  
  
public class Test  {  
    
 public static void go(){  
  A a=new B();  
  System.out.println(a.a+" "+a.getData());  
 }  
 public static void main(String [] args) {  
 go();  
 }  
}  
  
Choose the correct output:

**1)**. **B Bye**

**2)**. **A Bye**

**3)**. A Hi

**4)**. B Hi

**5)**. Compilation error

**Solution** :  
option [2] is correct

**Q.** Given:  
  
3. public class Spock {  
4. public static void main(String[] args) {  
5. int mask = 0;  
6. int count = 0;  
7. if( ((5<7) || (++count < 10)) | mask++ < 10 ) mask = mask + 1;  
8. if( (6 > 8) ^ false) mask = mask + 10;  
9. if( !(mask > 1) && ++count > 1) mask = mask + 100;  
10. System.out.println(mask + " " + count);  
11. }  
12. }  
  
Which two are true about the value of mask and the value of count at line 10? (Choose two.)

**1)**. mask is 0

**2)**. mask is 1

**3)**. **mask is 2**

**4)**. mask is 10

**5)**. mask is greater than 10

**6)**. **count is 0**

**Solution** :  
option [3,6] are correct

**Q.** public class Test {  
  int z=10;  
 public static void main(String[] args){  
 Test myScope = new Test();  
 int z = 6;  
 System.out.println(z);  
 doStuff();  
 System.out.println(z);  
 System.out.println(myScope.z);  
 }  
 static void doStuff() {  
 int z = 5;  
 doStuff2();  
 System.out.println(z);  
 }  
 static void doStuff2()  
 {  
 int  z = 4;  
 }  
}  
  
Choose the correct one:

**1)**. **6 5 6 10**

**2)**. 6 6 6 6

**3)**. 6 4 5 4

**4)**. Compilation error

**Solution** :  
option [1] is correct

**Q.** interface Myservice{  
 void getData() throws SQLException;  
}  
  
public class Test implements Myservice {  
    
 public static void main(String[] args){  
 }  
  
 //code inserted   
}  
  
What is the valid code to be inserted in commented area?

**1)**. **public void getData() throws SQLException { // TODO Auto-generated method stub }**

**2)**. **public void getData() { // TODO Auto-generated method stub }**

**3)**. public void getData() throws IOException { // TODO Auto-generated method stub }

**4)**. public void getData() throws Exception { // TODO Auto-generated method stub }

**Solution** :  
option [1,2] are correct

**Q.** Given:  
  
3. public class OverAndOver {  
4. static String s = "";  
5. public static void main(String[] args) {  
6. try {  
7. s += "1";  
8. throw new Exception();  
9. } catch (Exception e) { s += "2";  
10. } finally { s += "3"; doStuff(); s += "4";  
11. }  
12. System.out.println(s);  
13. }  
14. static void doStuff() { int x = 0; int y = 7/x; }  
15. }  
  
What is the result?

**1)**. 12

**2)**. 13

**3)**. 123

**4)**. 1234

**5)**. Compilation fails

**6)**. **An exception is thrown with no other output**

**Solution** :  
option [6] is correct

**Q.** public class Test  {  
   
 public static void main(String[] args) {  
  ArrayList<String> myList = new ArrayList<String>();  
  myList.add("One");  
  myList.add(3,"Two");  
  myList.add("Three");  
  myList.add("Four");  
}  
}  
  
What is the result?

**1)**. **java.lang.IndexOutOfBoundsException**

**2)**. One Three Four Two

**3)**. Compilation fails

**4)**. **RunTime Exception**

**5)**. One Two Three Four

**Solution** :  
option [1] is correct

**Q.** class A{  
A(){  
  System.out.println("a");  
 }  
 A(String a){  
  System.out.println("A");  
 }  
}  
class B extends A {  
 B(){  
  System.out.println("b");  
 }  
 B(String a){  
  System.out.println("B");  
 }  
}  
  
public class Test  {  
    
 public static void go(){  
  new B();  
 }  
 public static void main(String [] args) {  
 go();  
 }  
}  
  
Choose Correct one:

**1)**. **a b**

**2)**. **A B**

**3)**. a B

**4)**. b

**5)**. a

**Solution** :  
option [1] is correct

**Q.** Given:  
  
class Fork {  
public static void main(String[] args) {  
if(args.length == 1 | args[1].equals("test")) {  
System.out.println("test case");  
} else {  
System.out.println("production " + args[0]);  
}  
}  
}  
  
And the command-line invocation:  
  
java Fork live2  
  
What is the result?

**1)**. test case

**2)**. production live2

**3)**. **test case live2**

**4)**. Compilation fails

**5)**. **An exception is thrown at runtime**

**Solution** :  
option [5] is correct

**Q.** Given:  
  
3. class Box {  
4. int size;  
5. Box(int s) { size = s; }  
6. }  
7. public class Laser {  
8. public static void main(String[] args) {  
9. Box b1 = new Box(5);  
10. Box[] ba = go(b1, new Box(6));  
11. ba[0] = b1;  
12. for(Box b : ba) System.out.print(b.size + " ");  
13. }  
14. static Box[] go(Box b1, Box b2) {  
15. b1.size = 4;  
16. Box[] ma = {b2, b1};  
17. return ma;  
18. }  
19. }  
  
What is the result?

**1)**. **4 4**

**2)**. 5 4

**3)**. 6 4

**4)**. 4 5

**5)**. 6 6

**Solution** :  
option [1] is correct

**Q.** Given:  
  
3. public class Gotcha {  
4. public static void main(String[] args) {  
5. // insert code here  
6.  
7. }  
8. void go() {  
9. go();  
10. }  
11. }  
  
And given the following three code fragments:  
  
I. new Gotcha().go();  
II. try { new Gotcha().go(); }  
catch (Error e) { System.out.println("ouch"); }  
III. try { new Gotcha().go(); }  
catch (Exception e) { System.out.println("ouch"); }  
  
When fragments I-III are added, independently, at line 5, which are true?  
(Choose all that apply.)

**1)**. Some will not compile

**2)**. **They will all compile**

**3)**. All will complete normally

**4)**. None will complete normally

**5)**. **Only one will complete normally**

**6)**. Two of them will complete normally

**Solution** :  
option [2,5] are correct

**Q.** Given:  
  
4. public class SpecialOps {  
5. public static void main(String[] args) {  
6. String s = "";  
7. boolean b1 = true;  
8. boolean b2 = false;  
9. if((b2 = false) | (21%5) > 2) s += "x";  
10. if(b1 || (b2 == true)) s += "y";  
11. if(b2 == true) s += "z";  
12. System.out.println(s);  
13. }  
14. }  
  
Which are true? (Choose all that apply.)

**1)**. Compilation fails

**2)**. x will be included in the output

**3)**. **y will be included in the output**

**4)**. z will be included in the output

**5)**. An exception is thrown at runtime

**Solution** :  
option [3] is correct

**Q.** public class Test  {  
   
 public static void main(String[] args) {  
 System.out.print("answer is "+3+6+7);  
 System.out.print("answer is "+(3+6+7));  
 }  
   
}  
  
Choose the correct One:

**1)**. **answer is 367answer is 16**

**2)**. answer is 16 answer is 16

**3)**. compilation error

**4)**. Runtime exception

**Solution** :  
option [1] is correct

**Q.** Given:  
  
3. class Dozens {  
4. int[] dz = {1,2,3,4,5,6,7,8,9,10,11,12};  
5. }  
6. public class Eggs {  
7. public static void main(String[] args) {  
8. Dozens [] da = new Dozens[3];  
9. da[0] = new Dozens();  
10. Dozens d = new Dozens();  
11. da[1] = d;  
12. d = null;  
13. da[1] = null;  
14. // do stuff  
15. }  
16. }  
  
Which two are true about the objects created within main(), and which are eligible for garbage  
collection when line 14 is reached?

**1)**. Three objects were created

**2)**. Four objects were created

**3)**. **Five objects were created**

**4)**. **Zero objects are eligible for GC**

**5)**. **One object is eligible for GC**

**6)**. **Two objects are eligible for GC**

**Solution** :  
option [3,6] are correct

**Q.** Given:  
  
3. interface Vessel { }  
4. interface Toy { }  
5. class Boat implements Vessel { }  
6. class Speedboat extends Boat implements Toy { }  
7. public class Tree {  
8. public static void main(String[] args) {  
9. String s = "0";  
10. Boat b = new Boat();  
11. Boat b2 = new Speedboat();  
12. Speedboat s2 = new Speedboat();  
13. if((b instanceof Vessel) && (b2 instanceof Toy)) s += "1";  
14. if((s2 instanceof Vessel) && (s2 instanceof Toy)) s += "2";  
15. System.out.println(s);  
16. }  
17. }  
  
What is the result?

**1)**. 0

**2)**. 01

**3)**. **012**

**4)**. 02

**Solution** :  
option [3] is correct

**Q.** public abstract interface Frobnicate { public void twiddle(String s); }  
  
Which is a correct class? (Choose all that apply.)

**1)**. public abstract class Frob implements Frobnicate { public abstract void twiddle(String s) { } }

**2)**. **public abstract class Frob implements Frobnicate { }**

**3)**. public class Frob extends Frobnicate { public void twiddle(Integer i) { } }

**4)**. public class Frob implements Frobnicate { public void twiddle(Integer i) { } }

**5)**. **public class Frob implements Frobnicate { public void twiddle(String i) { } public void twiddle(Integer s) { } }**

**Solution** :  
option [2,5] are correct

**Q.** public class Test  {  
   
 public static void main(String[] args) {  
  X x=new X();  
  X y=new Y();  
    
  if(x instanceof Object){  
   System.out.println("True");  
  }  
  if(y instanceof Y){  
   System.out.println("True");  
  }  
  Y y1=(Y) x;  
  if(y1 instanceof X){  
   System.out.println("True");  
  }  
   
}  
}  
  
What is the result?

**1)**. **True True Exception**

**2)**. True True True

**3)**. Compilation fails due to multiple errors

**4)**. RunTime Exception

**5)**. True Exception

**Solution** :  
option [1] is correc

**Q.** Given:  
  
1. class Dims {  
2. public static void main(String[] args) {  
3. int[][] a = {{1,2}, {3,4}};  
4. int[] b = (int[]) a[1];  
5. Object o1 = a;  
6. int[][] a2 = (int[][]) o1;  
7. int[] b2 = (int[]) o1;  
8. System.out.println(b[1]);  
9. } }  
  
What is the result? (Choose all that apply.)

**1)**. 2

**2)**. 4

**3)**. **An exception is thrown at runtime**

**4)**. Compilation fails due to an error on line 4

**5)**. Compilation fails due to an error on line 5

**6)**. Compilation fails due to an error on line 6

**Solution** :  
option [3] is correct

**Q.** public class Test  {  
   
 public static void main(String[] args) {  
  int[][] ba = {{1,2,3,4}, {1,2,3}};  
  System.out.println(ba[1].length + " " + ba.length);  
}  
}  
  
What is the result?

**1)**. **3 2**

**2)**. 4 3

**3)**. Compilation error

**4)**. **Run Time Exception**

**5)**. 4 6

**6)**. 4 4

**Solution** :  
option [1] is correct

**Q.** class Sub extends Exception{  
   
}  
class SubSub extends Sub{  
   
}  
class A{  
 void getData() throws Sub{  
  System.out.println("byee");  
 }  
}  
public class Test  extends A{  
   
 public static void main(String[] args) {  
  new Test().getData();  
}  
void getData() throws Exception {  
  System.out.println("Hi");  
 }  
}  
  
Choose the correct One?

**1)**. **Compilation Error**

**2)**. Run Time Exception

**3)**. Hi byee

**4)**. Byee

**5)**. Hi

**Solution** :  
option [1] is correct

**Q.** class X { void do1() { } }  
class Y extends X { void do2() { } }  
  
public class Test  {  
  public static void main(String [] args) {  
  X x1 = new X();  
   X x2 = new Y();  
   Y y1 = new Y();  
   //insert code here  
 }  
}  
  
Which of the following, inserted at line 9, will compile? (Choose all that apply.)

**1)**. x2.do2();

**2)**. (Y)x2.do2();

**3)**. **((Y)x2).do2();**

**4)**. None of the above statements will compile

**Solution** :  
option [3] is correct

**Q.** public class Test  {  
   
 public static void main(String[] args) {  
   Days d1 = Days.TH;  
   Days d2 = Days.M;  
     
  for(Days d: Days.values()) {  
  if(d.equals(Days.F)) break;  
  d2 = d;  
  }  
  System.out.println((d1 == d2)?"same old" : "newly new");  
  }  
  enum Days {M, T, W, TH, F, SA, SU};  
   
}  
  
What is the result?

**1)**. **same old**

**2)**. **newly new**

**3)**. Compilation fails due to multiple errors

**4)**. RunTime Exception

**Solution** :  
option [1] is correct

**Q.** public class Test  {  
   
 public static void main(String[] args) {  
   Days d1 = Days.TH;  
   Days d2 = Days.M;  
     
  for(Days d: Days.values()) {  
  if(d.equals(Days.F)) break;  
  d2 = d;  
  }  
  System.out.println((d1 == d2)?"same old" : "newly new");  
  }  
  enum Days {M, T, W, TH, F, SA, SU};  
   
}  
  
What is the result?

**1)**. **same old**

**2)**. **newly new**

**3)**. Compilation fails due to multiple errors

**4)**. RunTime Exception

**Solution** :  
option [1] is correct

**Q.** Given:  
  
2. class MyException extends Exception { }  
3. class Tire {  
4. void doStuff() { }  
5. }  
6. public class Retread extends Tire {  
7. public static void main(String[] args) {  
8. new Retread().doStuff();  
9. }  
10. // insert code here  
11. System.out.println(7/0);  
12. }  
13. }  
And given the following four code fragments:  
I. void doStuff() {  
II. void doStuff() throws MyException {  
III. void doStuff() throws RuntimeException {  
IV. void doStuff() throws ArithmeticException {  
  
When fragments I-IV are added, independently, at line 10, which are true? (Choose all that apply.)

**1)**. None will compile

**2)**. They will all compile

**3)**. **Some, but not all, will compile**

**4)**. **All of those that compile will throw an exception at runtime**

**5)**. None of those that compile will throw an exception at runtime

**6)**. Only some of those that compile will throw an exception at runtime

**Solution** :  
option [3,4] are correct

**Attempted** :

**Q.** What will the result of below?  
  
class Star {  
public void doStuff() {  
System.out.println("Twinkling Star");  
}  
}  
interface Universe {  
public void doStuff();  
}  
class Sun extends Star implements Universe {  
public void doStuff() {  
System.out.println("Shining Sun");  
}  
}  
public class Bob {  
public static void main(String[] args) {  
Sun obj2 = new Sun();  
Star obj3 = obj2;  
((Sun) obj3).doStuff();  
((Star) obj2).doStuff();  
((Universe) obj2).doStuff();  
}  
}

**1)**. **Shining Sun Shining Sun Shining Sun**

**2)**. Shining Sun Twinkling Star Shining Sun

**3)**. Compilation error

**4)**. A ClassCastException is thrown at runtime.

**Solution** :  
option [1] is correct

**Q.** Given the code snippet:  
  
String s = "bob";  
String[] sa = {"a", "bob"};  
final String s2 = "bob";  
StringBuilder sb = new StringBuilder("bob");  
// switch(sa[1]) { // line 1  
// switch("b" + "ob") { // line 2  
// switch(sb.toString()) { // line 3  
// case "ann": ; // line 4  
// case s: ; // line 5  
// case s2: ; // line 6  
}  
  
And given that the numbered lines will all be tested by un-commenting one switch statement  
and one case statement together, which line(s) will FAIL to compile? (Choose all that apply.)

**1)**. line 1

**2)**. **line 2**

**3)**. line 3

**4)**. line 4

**5)**. **line 5**

**6)**. line 6

**Solution** :  
option [5] is correct

**Q.** class B extends Test{  
 public void getData(){  
  System.out.println("Hiii");  
 }  
}  
  
public class Test  {  
    
 public static void main(String[] args){  
  Test t=new B();  
  B b=new B();  
  t.getData();//Line 1  
  ((B) t).getData(); //Line 2  
  ((B)) t.getData();//Line3  
  b.getData();//Line4  
 }  
}  
  
Which of the line will give compilation error?

**1)**. **Line1**

**2)**. Line2

**3)**. **Line3**

**4)**. **Line4**

**Solution** :  
option [1,3] are correct