Aman Behl

Q6.

|  |
| --- |
| class ABC  {      public static void main(String [] args)      {          ABC g = new ABC();          g.start();      }        void start()      {          long [] a1 = {3, 4, 5};          long [] a2 = fix(a1);          System.out.print(a1[0] + a1[1] + a1[2] + " ");          System.out.println(a2[0] + a2[1] + a2[2]);      }        long [] fix(long [] a3)      {          a3[1] = 7;          return a3;      }  } |

options:  
a) 12 15  
b) 15 15  
c) 3 4 5 3 7 5  
d) 3 7 5 3 7 5

b)

Q7. class BitShift

{

    public static void main(String [] args)

    {

        int x = 0x80000000;

        System.out.print(x + " and  ");

        x = x >>> 31;

        System.out.println(x);

    }

}

options:  
a) -2147483648 and 1  
b) 0x80000000 and 0x00000001  
c) -2147483648 and -1  
d) 1 and -2147483648

a)

Q8.

|  |
| --- |
| import java.util.\*;    public class Test {  public static void main(String[] args)      {          int[] x = new int[3];          System.out.println("x[0] is " + x[0]);      }  } |

A. The program has a compile error because the size of the array wasn’t specified when declaring the array.  
B. The program has a runtime error because the array elements are not initialized.  
C. The program runs fine and displays x[0] is 0.  
D. The program has a runtime error because the array element x[0] is not defined.

C.

Q9.

|  |
| --- |
| import java.util.\*;    public class Test {  public static void main(String[] args)      {          int[] x = { 120, 200, 016 };          for (int i = 0; i < x.length; i++)              System.out.print(x[i] + " ");      }  } |

**Options :**  
A. 120 200 16  
B. 120 200 14  
C. 120 200 016  
D. 016 is a compile error. It should be written as 16.

B.

Q10.

|  |
| --- |
| import java.util.\*;    public class Test {  public static void main(String args[])      {          String s1 = "java";          String s2 = "java";          System.out.println(s1.equals(s2));          System.out.println(s1 == s2);      }  } |

A. false true  
B. false false  
C. true false  
D. true true

D.

Q11.Which of these is static variable defined in Collections?  
a) EMPTY\_SET  
b) EMPTY\_LIST  
c) EMPTY\_MAP  
d) All of the mentioned

d)

Q12. . Which of this interface must contain a unique element?  
a) Set  
b) List  
c) Array  
d) Collection

a)

Q13. Which of these interface handle sequences?  
a) Set  
b) List  
c) Comparator  
d) Collection

b)

Q14. import java.util.\*;

class Array

{

public static void main(String args[])

{

int array[] = new int [5];

for (int i = 5; i > 0; i--)

array[5 - i] = i;

Arrays.sort(array);

for (int i = 0; i < 5; ++i)

System.out.print(array[i]);;

}

}

a) 12345

b) 54321

c) 1234

d) 5432

a)

Q15. import java.util.\*;

class Collection\_Algos

{

public static void main(String args[])

{

LinkedList list = new LinkedList();

list.add(new Integer(2));

list.add(new Integer(8));

list.add(new Integer(5));

list.add(new Integer(1));

Iterator i = list.iterator();

Collections.reverse(list);

Collections.sort(list);

while(i.hasNext())

System.out.print(i.next() + " ");

}

}

a) 2 8 5 1

b) 1 5 8 2

c) 1 2 5 8

d) 2 1 8 5

c)

Q16. import java.util.\*;

class Collection\_Algos

{

public static void main(String args[])

{

LinkedList list = new LinkedList();

list.add(new Integer(2));

list.add(new Integer(8));

list.add(new Integer(5));

list.add(new Integer(1));

Iterator i = list.iterator();

Collections.reverse(list);

Collections.shuffle(list);

while(i.hasNext())

System.out.print(i.next() + " ");

}

}

a) 2 8 5 1

b) 1 5 8 2

c) 1 2 5 8

d) Any random order

d)

Q17. package com.getSizehashset;

import java.util.HashSet;

import java.util.Iterator;

public class HashsetExample{

public static void main(String[] args) {

HashSet<Integer> hashSet = new HashSet();

hashSet.add(1);

hashSet.add(2);

hashSet.add(3);

hashSet.add(4);

hashSet.add(5);

hashSet.add(6);

hashSet.add(7);

hashSet.add(8);

System.out.println("Size of HashSet after addition : " + hashSet.size());

System.out.println("Hashset contains");

Iterator it=hashSet.iterator();

while(it.hasNext()){

System.out.println(it.next());

} }}

a)Size of HashSet after addition 8

b)Size of HashSet after addition 7

c)compilation error

d)none of these

Q18.