

```

Q> int data[] = {2010,2013,2014,2015,2014};
    int key = 2014;
    int count=0;
    for(int e:data){
        if(e!=key){
            continue;
            count++;
        }
    }
    System.out.println(count+" found");

```

What is the result?

- A. Compilation fails
- B. 0 found
- C. 1 found
- D. 3 found

Answer: A (unreachable code becoz of continue)

```

Q> String[] arr = {"A","B","C","D"};
for(int i =0;i< arr.length;i++){
    System.out.print(arr[i]+" ");
    if(arr[i].equals("C"))
        continue;
    System.out.println("Work done");
    break;
}

```

- A. A B C Work done
- B. A B C D Work done
- C. A Work done
- D. Compilation fails

```

arr[0] = "A", arr[1]= "B", arr[2] ="C", arr[3]="D"
i =0, arr.length=4
0<4(true)
A Work done

```

Answer: C

```

Q> String[] str = new String[2];
int idx = 0;
for(String s: str){
    str[idx].concat(" element " + idx);
    idx++;
}
for(idx =0;idx<str.length;idx++){
    System.out.print(str[idx]);
}

```

- A. element 0
element 1
- B. null element 0
null element 1
- C. null
null
- D. NullPointerException is thrown at runtime

Answer: D

Q>

```
String[][] arr = {"A","B","C"}, {"D","E"};
for(int i =0;i< arr.length;i++){
    for(int j=0;j<arr[i].length;j++){
        System.out.print(arr[i][j]+" ");
        if(arr[i][j].equals("B"))
            break;
    }
    continue;
}
```

- A. A B C
- B. A B C D E
- C. A B D E
- D. Compilation fails

Answer: C

Q>

```
public class Test {
    public static void main(String[] args) {
        StringBuffer sb =new StringBuffer("java");
        String s = "java";
        if (sb.toString().equals(s.toString()))
            System.out.println("Match -1");
        else if(sb.equals(s))
            System.out.println("Match -2");
        else
            System.out.println("No Match");
    }
}
```

- A. Match -1
- B. Match -2
- C. No Match
- D. Null Pointer excpetion at runtime

Answer: A

Q> int[] a =new int[]? What is the array size ?

- A. 0
- B. 4
- C. 1
- D. Can't find it results in Compiletime error
- E. exception at the runtime

Answer: D

Q> int[] a =new int[0]; will the code compile?

- A. yes
- B. no

Answer: A

Q> int[] a=new int[-5];? What it the size of the array?

- A. 5
- B. -5
- C. 0
- D. 1

- F. Compilation error as size is negative
- G. NegativeArraySizeException is occurred
- H. ArrayIndexOutOfBoundsException is occurred

Answer: G

ArrayIndexOutOfBoundsException => Array is already created and if we go out of index while reading or writing to array we get.

NegativeArraySizeException is occurred => During creation of an Array it would result.

Q> Consider below code:

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println("ONE");  
    }  
    public static void main(Integer[] args) {  
        System.out.println("TWO");  
    }  
    public static void main(byte [] args) {  
        System.out.println("THREE");  
    }  
}
```

What will be the result if Test class is executed by below command?

java Test 10

- A. TWO
- B. ONE
- C. THREE
- D. Compilation Error

Answer: B

Q>

```
public class Test {  
    public static void main(String[] args) {  
        if ((isItSmall(3)) || (isItSmall(7))) { //line-3  
            System.out.println("Result is true");  
        }  
        if ((isItSmall(6)) || (isItSmall(9))) { //line-6  
            System.out.println("Result is true");  
        }  
    }  
    public static boolean isItSmall(int i) {  
        if (i < 5) {  
            System.out.println("i < 5");  
            return true;  
        } else {  
            System.out.println("i >= 5");  
            return false;  
        }  
    }  
}
```

What is the result?

- A. Compilation Error at line 3
- B. Compilation Error at line 6
- C. i<5
Result is true
- D. i<5
Result is true

```
i>=5  
Result is true
```

E. i<5
Result is true
i>=5
i>=5
Result is true

F. i<5
Result is true
i>=5
i>=5

Answer: F

```
Q>  
class Feline {  
    public static void main(String[] args) {  
        Long x = 42L;  
        Long y = 44L;  
        System.out.print(" " + 7 + 2 + " "); // 72  
        System.out.print(foo() + x + 5 + " "); //foo425  
        System.out.println(x + y + foo()); //86foo  
    }  
    static String foo() { return "foo"; }  
}
```

What is the result?

- A. 9 foo47 86foo
- B. 9 foo47 4244foo
- C. 9 foo425 86foo
- D. 9 foo425 4244foo
- E. 72 foo47 86foo
- F. 72 foo47 4244foo
- G. 72 foo425 86foo
- H. 72 foo425 4244foo

Answer: G

