Table of contents

Java- First Problem Statement

Question:

Write the main method in the Solution class.

In the main method, read an integer value and print "TRUE" if it contains at least 3 odd digits. Else it should print "FALSE".

For example, if the value is 123456 and it contains 3 odd digits such as 1,3,5. So it should print "TRUE".
The output should be in the format of sample output.

Sample input1:
123456

Output:
TRUE

Sample input2:
123

Output:
FALSE

Sample code snippet for reference:
Please use below code to build your solution.

```
public class Solution
{

        public static void main(String[] args)
        {
                //code to read values
                //code to display the result
        }

}
```

Solution:

```
import java.util.Scanner;

public class SolutionSimple {

      public static void main(String[] args) {

            Scanner sc = new Scanner(System.in);
            int count = 0;
            int no = sc.nextInt();
            while(no>0){
                  int remainder = no%10;
                  if(remainder%2!=0){
                        count++;
                  }
                  no = no/10;
            }

            if(count>=3){
                  System.out.println("TRUE");
            }else{
                  System.out.println("FALSE");
            }

      }
}
```

Testcases:

```
TestCase1:
Input:
123456

Output:
TRUE
```

```
TestCase2:
Input:
123


Output:
FALSE
```

```
TestCase3:
Input:
24680


Output:
FALSE

TestCase4:
Input:
135


Output:
TRUE
```

Java- Second Problem Statement

Question:

Create a class Projector with below attributes:

projectorId - int
projectorName - String
price - int (per unit)
rating - int

availableIn - String (Amazon, Flipkart, TataCliq)

The above attributes should be private, write getters, setters and parameterized constructor as required.

Create class Solution with main method.

Implement one static method - findMaximumPriceByRating in Solution class.

findMaximumPriceByRating method:
The method will return the maximum priced Projector object from array of Projector objects whichever has rating greater than the given rating and which is available in TataCliq.
If no Projector with the above conditions is present in the array of Projector objects, then the method should return null.

Note : No two Projector will have the same price.
All the searches should be case insensitive.

The above mentioned static method should be called from the main method.

For findMaximumPriceByRating method - The main method should print the projectorId from returned object if the returned value is not null else it should print "No such Projector".

Eg: 1001
where 1001 is the projectorId which has the maximum price.

Before calling these static methods in main, use Scanner object to read the values of four Projector objects referring attributes in the above mentioned attribute sequence.
Next, read the value of one int parameter for capturing rating value.

Consider below sample input and output:

SampleTestcase1:
Input:
1001
Epson
30000
5
TataCliq
1002
BenQ
40000
4
Amazon
1003
Sony
80000

```
80000
5
TataCliq
1004
```

```
Optomo
7000
3
Flipkart
3
```

Output:

```
1003
```

Sample Testcase2:

Input:

```
1001
Epson
30000
5
Flipkart
1002
BenQ
40000
4
Amazon
1003
Sony
80000
5
TataCliq
1004
Optomo
7000
3
Flipkart
5
```

Output:

```
No such Projector
```

Sample code snippet for reference:

Please use below code to build your MyClass.

```
import java.util.Scanner;
public class Solution
{
  public static void main(String[] args)
 {
  //code to read values
  //code to call required method
  //code to display the result









 }

 //code the first method



}

//code the class
```

Note on using Scanner object:

Sometimes scanner does not read the new line character while invoking methods like nextInt(), nextDouble() etc.

Usually, this is not an issue, but this may be visible while calling nextLine() immediately after those methods.

Consider below input values:

```
1001
Savings
```

Referring below code:

```
Scanner sc = new Scanner(System.in);
int x = sc.nextInt();
```

String str = sc.nextLine(); -> here we expect str to have value Savings.Instead it may be "".

If above issue is observed, then it is suggested to add one more explicit call to nextLine() after reading numeric value.

Solution:

```
import java.util.Scanner;

public class Solution {

    public static void main(String[] args) {

        Projector[] proj = new Projector[4];

        Scanner sc = new Scanner(System.in);
        for (int i = 0; i < proj.length; i++) {
            int projectorId = sc.nextInt();
            sc.nextLine();
            String projectorName = sc.nextLine();
            int price = sc.nextInt();
            int rating = sc.nextInt();
            sc.nextLine();
            String availableIn = sc.nextLine();

            proj[i] = new Projector(projectorId, projectorName, price,
rating, availableIn);
        }
        int givenRating = sc.nextInt();
        Projector min = findMaximumPriceByRating(proj, givenRating);
        if (min != null) {
            System.out.println(min.getProjectorId());
        } else {
            System.out.println("No such Projector");
        }

        sc.close();
    }

    public static Projector findMaximumPriceByRating(Projector[] proj, int
givenRating) {

        Projector projector = null;
        int price = Integer.MIN_VALUE;
        for (int i = 0; i < proj.length; i++) {
            if ((proj[i].getPrice() > price) && (proj[i].getRating() >
givenRating) && proj[i].getAvailableIn().equalsIgnoreCase("TataCliq")) {
                price = proj[i].getPrice();
                projector = proj[i];
            }
        }
        return projector;

    }
}

class Projector {

    private int projectorId;
    private String projectorName;
    private int price;
    private int rating;
    private String availableIn;
    public int getProjectorId() {
        return projectorId;
    }
    public void setProjectorId(int projectorId) {
        this.projectorId = projectorId;
    }
    public String getProjectorName() {
        return projectorName;
    }
    public void setProjectorName(String projectorName) {
        this.projectorName = projectorName;
    }
    public int getPrice() {
        return price;
    }




    public void setPrice(int price) {
        this.price = price;
```

```
            this.price = price;
      }
      public int getRating() {
            return rating;
      }
      public void setRating(int rating) {
            this.rating = rating;
      }
      public String getAvailableIn() {
            return availableIn;
      }
      public void setAvailableIn(String availableIn) {
            this.availableIn = availableIn;
      }
      public Projector(int projectorId, String projectorName, int price, int
rating, String availableIn) {
            this.projectorId = projectorId;
            this.projectorName = projectorName;
            this.price = price;
            this.rating = rating;
            this.availableIn = availableIn;
      }

}
```

TestCases:

```
Testcase1:
Input:
1001
Epson
30000
5
TataCliq
1002
BenQ
40000
4
Amazon
1003
Sony
80000
5
TataCliq
1004
Optomo
7000
3
Flipkart
3
```

```
Output:
1003

Testcase2:
Input:
1001
Epson
30000
5
Flipkart
1002
BenQ
40000
4
Amazon
1003
Sony
80000
5
TataCliq
1004
Optomo
7000
3
Flipkart
5


Output:
No such Projector


Testcase3:
Input:
12345
Epson
30000
5
```

```
TataCliq
54321
BenQ
40000
4
Amazon
12789
Sony
80000
5
TataCliq
98765
Optomo
7000
3
Flipkart
3


Output:
12789

Testcase4:
Input:
12345
Epson
30000
5
TataCliq
54321
BenQ
40000
4
Amazon
12789
Sony
80000
5
TataCliq
98765
Optomo
700000
5
TataCliq
3


Output:
98765
```