

Forum: [Hack Forums](#)

Thread: [Print out all odd numbers in random set of numbers ?](#)

cryptobassboost - February 6, 2019, 8:31 am

Hey guys,

so I'm doing some assignments in the R programming language, and I think it's kinda similar to java so hoping to get help here.

Suppose you have a random set of numbers, say 's <- c(1,5,4,2)'
How would I print out all odd numbers the integer way?

I can do it using booleans, like this:
`s[apply(s, "%%", 2) != 0]`

But how would I do it using integers?
Obviously, `s[c(1,2)]` wouldn't work because if the set of number changes, it doesn't hold anymore.

Help would be appreciated!

SIM-3364521 - February 6, 2019, 12:28 pm

in java:

```
int[] arr = new int[]{1,5,4,2};

for(int i=0; i<arr.length; i++){
    System.out.print(arr[i] * (arr[i]%2));
}
```

you cant make decisions without using booleans so it will print 0 if the number was even.
not really sure if that is what you wanted but it will print all odd numbers and will print 0 if the number was even
you would have to use a boolean to filter out the even numbers completely.

V For Vendetta - February 6, 2019, 12:31 pm

<https://stackoverflow.com/questions/18950437/how-to-print-out-an-odd-integers-in-r>

Talon - February 6, 2019, 7:27 pm

Code public class oddInt

```
{
    public static void main(String[] args)
    {
        int[] array = new int[]{5,2,4,6,1,3, 9, 7 ,3, 23,43,1,2,34,5,6,7,10};
        for(int i = 0; i < array.length; i++)
        {
            if(array[i] % 2 != 0)
            {
                System.out.println(array[i]);
            }
        }
    }
}
```

```
}
```

```
}
```

Output:

```
5
1
3
9
7
3
23
43
1
5
7
```

Caliban - February 7, 2019, 4:28 pm

(Feb 6, 2019 08:27 PM)T?lon Wrote: [CITE](#) Code public class oddInt

```
{
    public static void main(String[] args)
    {
        int[] array = new int[]{5,2,4,6,1,3, 9, 7 ,3, 23,43,1,2,34,5,6,7,10};
        for(int i = 0; i < array.length; i++)
        {
            if(array[i] % 2 != 0)
            {
                System.out.println(array[i]);
            }
        }
    }
}
```

Output:

```
5
1
3
9
7
3
23
43
1
5
7
```

You can also use this since Java 8:

```
Code public class OddInt {
    public static void main(String[] args) {
        int[] array = {5, 2, 4, 6, 1, 3, 9, 7, 3, 23, 43, 1, 2, 34, 5, 6, 7, 10};
        Arrays.stream(array).filter(x -> (x % 2 != 0)).forEach(System.out::println);
    }
}
```

Looks more sexy to me to do it in a functional style, and streaming can also boost performance