

EECS1022 Programming for Mobile Computing
(Winter 2021)

Q&A - Exam

Friday, April 16

- EECS2030 (F21) ^{2 sections} B → E
- Exam PDF Guide (M,N,O eClass site)
- Between 40 and 60 questions (≥ hours)
- Tracing questions: your judgement of tracing by hands or by Eclipse

single answer

-
-
-
-

- multiple
- -
 -
 -

Short answers

--- ?

— ?

```
String a = "Hello";  
String b = "Hello";  
System.out.print(a == b);
```

printed ==
strings. equals

the correct answer to this is: true. But I think it should be false.

Because String is a reference type and "==" compares the address not the value.

Lab 7 solutions
Lab 8

```
String s1 = "Jackie";  
String s2 = "Jackie";
```

s10 = "Jackie"s

```
System.out.println(s1 == s2);  
System.out.println(s1.equals(s2));
```

```
String s3 = new String("Jackie");  
String s4 = new String("Jackie");
```

```
System.out.println(s3 == s4); F  
System.out.println(s3.equals(s4));
```

```
Scanner input = new Scanner(System.in);  
System.out.println("Enter a string:");  
String s5 = input.nextLine();  
System.out.println("Enter another string:");  
String s6 = input.nextLine();
```

```
System.out.println(s5 == s6); F  
System.out.println(s5.equals(s6)); T
```

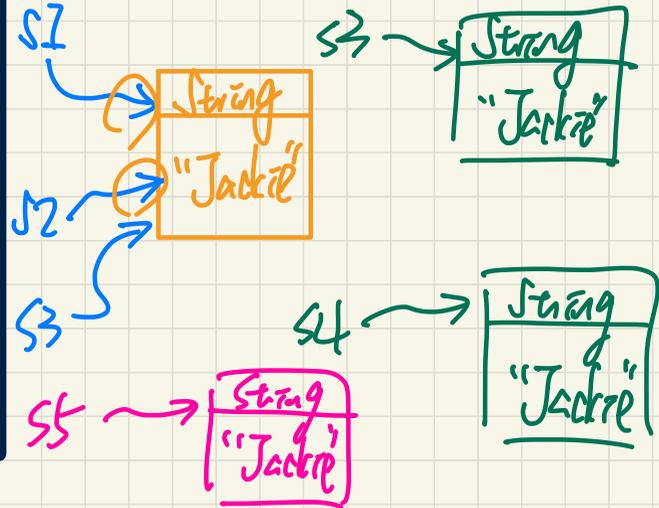
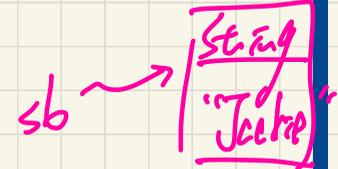
Console

E. a. S.:

Jackie

E. a. S.

Jackie



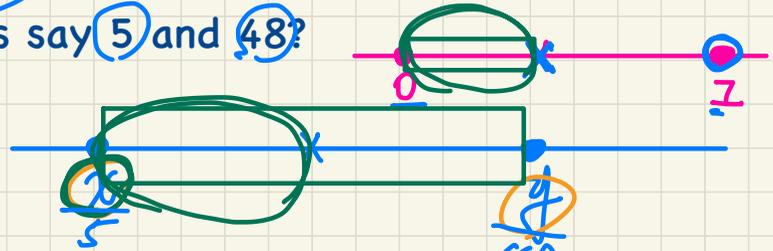
In the week 12 tutorial, we created a method called `indexOf`. I was wondering, why didn't we use the ArrayList method `indexOf`, instead?

→ !: you'll be comparing BB- against array.

For the `random()` method in Math class, how do we find a random number between 2 numbers other than 0 and 1. i know about the multiplication but how do we find a number between lets say 5 and 48?

casting.

`(int) (x + (y - x) * Math.random())`



✓ // Randomly generate 21 unique numbers from the range [1, 52].

```
int[] randomNumbers = new Random().ints(1, 52).distinct().limit(21).toArray();
```

In the ArrayList class, why does `add(E e)` method, have the return type boolean?

int

```
p.getDistanceFromOrigin();
```

no compilation error
but not useful.

this is a correct option, but I think it will have compiling error, because getDistanceFromOrigin() is a accessor method with a return value. And this option does not assign this return value to any variable.

Examples: `add` and `remove` of ArrayList

```
list.add("a");
```

```
list.remove("b");
```