

01

**Quais dos seguintes não são nomes de variáveis válidos? (Escolha dois.)**

**A. \_ | B. \_blue | C. 2blue  
D. blue\$ | E. Blue**

02

**Qual é o valor da gorjeta depois de executar o seguinte trecho de código?**

```
int almoco = 5; //lin1  
int gorjeta = 2; //lin2  
var total = almoco + (almoco>6 ? gorjeta++ : gorjeta--); //lin3
```

- A. 1 | B. 2 | C. 3 | D. 7**  
**E. Nenhuma das anteriores**

03

Qual é equivalente a `var q = 4.0f; ?`

- A.** float q = 4.0f; | **B.** Float q = 4.0f;  
**C.** double q = 4.0f; | **D.** Double q = 4.0f;  
**E.** Object q = 4.0f;

04

**Qual é a saída do seguinte?**

```
12: var b = "12";  
13: b += "3";  
14: b.reverse();  
15: System.out.println(b.toString());
```

- A.** 12 | **B.** 123 | **C.** 321  
**D.** O código não compila.

05

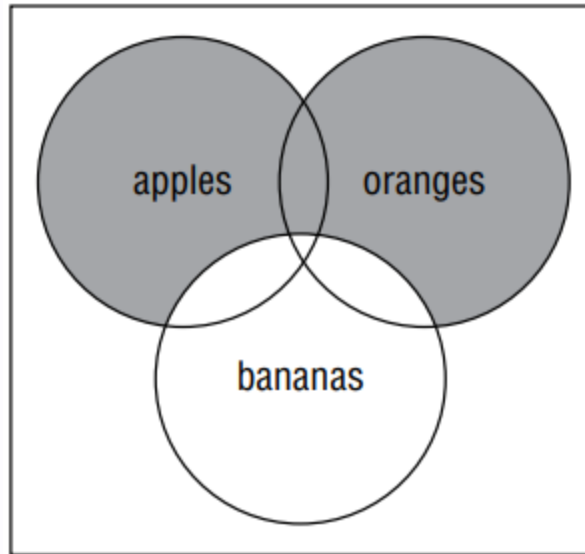
## Qual é a saída do seguinte?

```
5: var line = new StringBuilder("-");  
6: var anotherLine = line.append("-");  
  
7: System.out.print(line == anotherLine);  
8: System.out.print(" ");  
9: System.out.print(line.length());
```

- A.** false 1 | **B.** false 2
- C.** true 1 | **D.** true 2
- E.** O código não compila.

06

**Qual expressão representa mais de perto a região  
preenchida do diagrama?**



# 06

- A.** apples && oranges && !bananas
- B.** orange || (oranges && !bananas)
- C.** (apples || bananas) && oranges
- D.** oranges && apples
- E.** (apples || oranges) && !bananas
- F.** apples ^ oranges

07

## Qual é a saída do seguinte?

```
5: var line = new String("-");  
6: var anotherLine = line.concat("-");  
7: System.out.print(line == anotherLine);  
8: System.out.print(" ");  
9: System.out.print(line.length());
```

- A.** false 1 | **B.** false 2
- C.** true 1 | **D.** true 2
- E.** O código não compila.



08

**Qual pode preencher o espaço em branco?  
(Escolha dois.)**

```
public void math() {  
    _____ pi = 3.14;  
}
```

**A.** byte | **B.** double | **C.** float  
**D.** short | **E.** var

09

**Preencha os espaços em branco: Os operadores !=, \_\_\_\_\_, \_\_\_\_\_, \_\_\_\_\_ e ++ estão listados no mesmo ou aumentando o nível de precedência do operador. (Escolha dois.)**

- A.** ==, \*, ! | **B.** /, %, \* | **C.** !, \*, %  
**D.** +=, &&, \* | **E.** \*, <, /

10

## Quantos desses compilam?

```
18: Comparator<String> c1 = (j, k) -> 0;  
19: Comparator<String> c2 = (String j, String k) -> 0;  
20: Comparator<String> c3 = (var j, String k) -> 0;  
21: Comparator<String> c4 = (var j, k) -> 0;  
22: Comparator<String> c5 = (var j, var k) -> 0;
```

**A. 0 | B. 1 | C. 2**  
**D. 3 | E. 4 | F. 5**

**FIM**