

# Generation

## BRASIL

1. Como é feita a **entrada** e **saida** de dados no C#?

☐ A

leia()  
e  
escreva()

☐ B

input()  
e  
print()



Console.ReadLine();  
e  
Console.WriteLine();

2. O que significa uma linguagem ser **fortemente tipada**?

☐ A

Não possui tipos para definir suas  
variáveis



Variáveis e constantes têm um tipo, assim  
como cada expressão que é avaliada  
como um valor

☐ C

Somente algumas variáveis possuem  
atributos tipados

3. **C#** possui 4 tipos de variáveis, sendo elas: Tipos internos, tipos personalizados, tipos literais e tipos genéricos.



Verdadeiro

☐ B

Falso

4. A definição: Representam inteiros, valores de ponto flutuante, expressões booleanas, caracteres de texto, valores decimais e outros tipos de dados. É aplicada a qual tipo de variável?



Tipos Internos

☐ B

Tipos Personalizados

☐ C

Tipos Literais

☐ D

Tipos genéricos

5. A definição: Utilizado para criar seus próprios tipos personalizados. É aplicada a qual tipo de variável?

☐ A

Tipos Internos



Tipos Personalizados

☐ C

Tipos Literais

☐ D

Tipos genéricos

6. A definição: Você pode especificar como um literal numérico deve ser digitado anexando uma letra ao final do número. É aplicada a qual tipo de variável?

☐ A Tipos Internos ☐ B Tipos Personalizados  
☒ C Tipos Literais ☐ D Tipos genéricos

7. A definição: Um tipo pode ser declarado com um ou mais parâmetros de tipo que servem como um espaço reservado para o tipo real (o tipo concreto). É aplicada a qual tipo de variável?

☐ A Tipos Internos ☐ B Tipos Personalizados  
☐ C Tipos Literais ☒ D Tipos genéricos

8. É possível declarar variáveis de diferentes formas no C#, quais das alternativas abaixo está declarando corretamente a variável nome?

☐ A string nome; ☒ B string nome = "Gustavo Boaz";  
☐ C string nome = Console.ReadLine(); ☐ D string nome = "Gustavo" + " " + "Boaz";

9. Os Operadores (+, -, \*, /, %), são operadores:

☐ A Lógico ☐ B Relação  
☐ C Atribuição ☒ D Aritméticos

10. Os Operadores (=, +=, -=, \*=, /=, %=), são operadores:

☐ A Lógico ☐ B Relação  
☒ C Atribuição ☐ D Aritméticos





11. Os Operadores (==, !=, <, <=, >, >=), são operadores:

☐ A Lógico ☒ B Relação  
☐ C Atribuição ☐ D Aritméticos

12. Os Operadores (&&, ||), são operadores:




☒ A Lógico ☐ B Relação  
☐ C Atribuição ☐ D Aritméticos

13. Quais as finalidades de escrever uma boa **nomenclatura** ao desenvolver um código?

- |   |   |   |   |
|---|---|---|---|
| <input checked="" type="checkbox"/>  | Criam uma aparência consistente para o código, para que os leitores possam se concentrar no conteúdo e não no layout. | <input checked="" type="checkbox"/>  | Permitem que os leitores entendam o código com mais rapidez, fazendo suposições com base na experiência anterior. |
| <input checked="" type="checkbox"/>  | Facilitam a cópia, a alteração e a manutenção do código.  | <input checked="" type="checkbox"/>  | Demonstram as práticas recomendadas do C#.  |

14. O que esta acontecendo no trecho de código:

```
float preco = (float) Convert.ToDouble(Console.ReadLine());
```

- |   |   |  |   |
|---|---|--|---|
| <input checked="" type="checkbox"/>  | Esta convertendo de string para double, e depois convertendo para float     | <input type="checkbox"/>  | Esta convertendo de float para double, e depois fazendo um cast para string |
| <input type="checkbox"/>             | Esta convertendo de string para double, e depois fazendo um cast para float |  |   |