

Jogo da Adivinhação em Java:

Exercício

Objetivo: Criar um programa Java que simula o jogo da adivinhação, onde o jogador tenta adivinhar um número secreto entre 1 e 100. **Nível:** Iniciante

Instruções:

1. Definindo as variáveis: Java

```
private int numeroSecreto;  
private int palpite; private  
int tentativas;
```

2. Gerando o número secreto: Java

```
private void gerarNumeroSecreto() {  
    numeroSecreto = (int) (Math.random() * 100) + 1;  
}
```

3. Lendo o palpite do jogador:

Java

```
private void lerPalpite() {  
    Scanner scanner = new Scanner(System.in);
```

```
System.out.print("Digite seu palpite: ");  
palpite = scanner.nextInt();  
}
```

4. Comparando o palpite com o número

secreto: Java

```
private void adivinhar() {  
    tentativas = 0;  
    while (palpite != numeroSecreto) {  
  
tentativas++;  
        if (palpite < numeroSecreto) {  
            System.out.println("Seu palpite é menor que o número secreto!");  
        } else {  
            System.out.println("Seu palpite é maior que o número secreto!");  
        }  
    }  
}
```

```
System.out.println("Parabéns! Você adivinhou o número secreto em " +  
tentativas + " tentativas!");  
  
lerPalpite();  
}  
}
```

5. Exibindo o resultado: Java

```
public void mostrarResultado() {  
    System.out.println("O número secreto era: " + numeroSecreto);  
    System.out.println("Você utilizou " + tentativas + " tentativas!");  
}
```

6. Simulando o jogo:

Java

```

public static void main(String[] args) {

    JogoAdivinhacao jogo = new JogoAdivinhacao();
    jogo.gerarNumeroSecreto();
    while (true) {

        jogo.lerPalpite();
        jogo.adivinhar();

        jogo.mostrarResultado();

        System.out.print("Deseja
jogar novamente? (s/n): ");
        Scanner scanner = new
Scanner(System.in);
        String resposta =
scanner.nextLine();
        if (!resposta.equalsIgnoreCase("s")) {
            break;
        }
    }
}

```

Desafio 1: Criar o código do jogo

Em primeiro lugar gere o código completo, usando os snippets acima mencionados, para criar um jogo de adivinhação em Java

Agora, responda a estas perguntas:

Perguntas sobre o Jogo da Adivinhação em Java:

1. Variáveis (Passo 1):

- Quais variáveis são utilizadas no código?

```
numeroSecreto  
palpite  
tentativas  
resposta
```

- Qual é o tipo de dado de cada variável?

```
int tentativas  
int palpite  
int numeroSecreto  
String resposta
```

- Porquê que esses tipos de dados foram escolhidos?
- Porque fazia sentido usar ou integers ou strings,
- Qual é o escopo de cada variável?
Variavel de instancia
- As variáveis são inicializadas antes de serem usadas?
Sim
- Como as variáveis são nomeadas?
De forma pertinente

2. Gerando o número secreto (Passo 2):

- Qual método é utilizado para gerar o número secreto?
private void GerarNumeroSecreto()
- Qual é a função desse método?
Gerar um numero aleatória de 1 a 100
- Como o método gera um número aleatório entre 1 e 100?
Utilizando o math.random
- Quais outros métodos poderiam ser utilizados para gerar o número secreto?

```
int.rand
```

3. Lendo o palpite do jogador (Passo 3):

- Qual método é utilizado para ler o palpite do jogador?
private void lerPalpite()
- Qual é a função desse método?
Guardar o user input na variável palpite
- Como o método lê o palpite do jogador?
Criando um objecto na class scanner

- Quais erros podem ocorrer ao ler o palpite do jogador?
[Inserir uma string em vez de int](#)
- Como o código trata esses erros?

[Com o parse, com uma exception](#)

4. Comparando o palpite com o número secreto (Passo 4):

- Qual método é utilizado para comparar o palpite com o número secreto?
`private void adivinhar()`
- Qual é a função desse método?
[Testar ate o palpite for igual ao numero secreto](#)
- Como o método compara o palpite com o número secreto?
`while (palpite != numeroSecreto)` com um ciclo while
- Que tipo de loop é utilizado para comparar o palpite com o número secreto?
[While](#)
- Por que esse tipo de loop foi escolhido?

[Porque executa ate que a condição seja verdadeira\(neste caso negativa\)](#)

5. Exibindo o resultado (Passo 5):

- Qual método é utilizado para exibir o resultado?
`public void mostrarResultado()`
- Qual é a função desse método?
[Exibir o resultado](#)
- Que informações o método exibe?
[O numero secreto e o nr de tentativas](#)
- Como o método formata a saída?
[Com o nr secreto no fim e o nr de tentativas no meio](#)
- Como o método pode ser modificado para exibir mais informações?

[Pode mostrar o palpite tambem](#)

6. Simulando o jogo (Passo 6):

- Qual método é utilizado para simular o jogo?
[Método main](#)
- Qual é a função desse método?
[Executar o código](#)
- Como o método simula o jogo?
[Fazendo um objecto da classe](#)
- Como o método pode ser modificado para permitir que o jogador jogue novamente?
[O método já permite que o jogador jogue novamente, através do ciclo while e do if](#)
- Como o método pode ser modificado para aumentar a dificuldade do jogo?

[Gerar um número de 1 a 1000.](#)

Bom trabalho!