

Aula - Construtores

Objetivos:

- Construtores
- Condicionais em java
- Entrada de Dados
- O método toString
- Introduzir o this
- Objetos como atributos de outros objetos: Cliente na ContaCorrente.

1. Classe Cliente do minimundo da agência bancária com atributos nome, cpf, gênero, idade.
2. Main que cria um cliente e inicializa os dados.
 - a. Ao criar a variável, esquecer o new e perguntar se está tudo certo.
 - b. Tentar fazer um print para mostrar os dados do Cliente na tela e comentar que ao rodar o programa, o que apareceria seria algo como "Cliente@36baf30c". Falar do método toString(), implementar com eles e explicar que o método é invocado automaticamente no método System.out.println ou sempre que for necessário transformar o objeto em uma representação na forma de String.
3. Uso do Scanner para ler os dados do cliente.

```
import java.util.Scanner;

(...)

public static void main(String[] args) throws Exception {
    // Criação do scanner para ler dados do teclado. O System.in indica que os dados devem
    // ser lidos da entrada padrão, isto é, do teclado.
    Scanner scanner = new Scanner(System.in);

    // O scanner possui vários métodos .next<Tipo>() como .nextDouble(), nextInt(), etc.
    // Para ler uma String até o próximo caractere de espaço, use o .next(). Para ler até o fim
    // da linha, use .nextLine(). O primeiro é útil para ler itens como o CPF que não possuem
    // espaço. O segundo é usado para ler, por exemplo, nomes completos de pessoas que
    // possuem espaços.
    System.out.print("Nome: ");
    String nome = scanner.nextLine();

    System.out.print("CPF: ");
    String cpf = scanner.next();

    System.out.print("Genero (m ou f): ");
    char genero = scanner.next().charAt(0); // le uma palavra e seleciona o primeiro char

    System.out.print("Idade: ");
    int idade = scanner.nextInt();
}
```

```
// fechamento do Scanner
scanner.close();
}
```

4. [Exercício] Escrever a main para ler os dados da conta corrente.
5. Teoria de Construtores
6. Construtor da classe Cliente que recebe os valores de todos os atributos.
7. [Exercício] Pedir que eles criem a classe ContaCorrente de novo e que façam o construtor.
8. Pode existir mais de um construtor e eles são escolhidos usando a quantidade e os tipos dos argumentos. Fazer um construtor que inicialize o saldo e o limite com valores padrão de 0 e 100.
9. Condicionais em Java. Adicionar um condicional para verificar se o cliente possui mais de 18 anos partindo da hipótese que apenas a partir desta idade pessoas podem abrir contas.
10. [Exercício] Criar os métodos:
 - a. depósito(double valor) na ContaCorrente que retorna um valor booleano indicando se a operação foi realizada com sucesso. Se o valor for menor que 0, exibir uma mensagem e não atualizar saldo.
 - b. saque(double valor) na ContaCorrente que retorne um booleano indicando se o saque pode ser realizado ou não. Usar o método disponível() para verificar o cliente pode sacar o valor. Se puder, decrementar o saldo.