**1. CLASSES DO TIPO INTERFACE**

**1.1 Definição**

“Interface é um tipo de classe que contém apenas assinaturas de métodos. É praticamente como um contrato entre a classe e o mundo externo. Ao herdar uma interface, uma classe está comprometida a fornecer o comportamento publicado pela interface, ou seja , escrever todos os métodos assinados, definidos pela interface. Devido a esta obrigação pela implementação da interface. Devido a esta obrigação pela implementação da interface, utiliza-se o termo "a classe X implementa a interface Y.". (ARAUJO, 2017).

“Se a classe não tiver nenhum método não-abstrato, podemos criá-la como uma interface, que segue um modelo de declaração diferente do usado para classes mas tem funcionalidade similar a de classes abstratas. Assim como uma classe abstrata, uma interface não pode ser instanciada. Todos os métodos na interface são implicitamente abstract e public, e não podem ser declarados com seus corpos. Campos, se houverem, serão implicitamente considerados como static e final, devendo, portanto, ser inicializados na sua declaração.”. (SANTOS, 2001)

**1.2 Utilidade**

“A diferença essencial entre classes abstratas e interfaces é que uma classe herdeira somente pode herdar de uma única classe (abstrata ou não), enquanto que qualquer classe pode implementar várias interfaces simultaneamente. Interface então, um mecanismo simplificado de implementação de **herança múltipla em Java**, permitindo que mais de uma interface determine métodos que uma classe herdeira deve implementar. Interfaces também podem ser úteis para implementar bibliotecas de constantes: já que todos os campos de uma interface devem ser declarados como static e final, podemos escrever interfaces que somente contém campos, e qualquer classe que implementar esta interface terá acesso a estes campos.” (SANTOS, 2001).

**1.4 O que são métodos abstratos**

“Um método com uma declaração, mas sem instruções nem corpo, é chamado de método abstrato. As classes que herdam precisam implementar os métodos abstratos, da mesma forma quando herdam de uma interface.” (STELLMAN, GREENE, 2011).

**Referências:**

ARAUJO, Everton. **Orientação a Objetos em C#: Conceitos e implementações em .NET.** Editora Casa do Código, São Paulo, 2017.

SANTOS, Rafael. **Introdução a Programação Orientada a Objetos usando Java**, 2001.

STELLMAN, Andrew; GREENE, Jennifer. **O guia amigo do seu cérebro. Use a Cabeça C#**. Editora Alta Books, Rio de Janeiro, 2011.