

Aluno: Rodolfo Almeida

Professor: Bruno Dias

Disciplina: MAP

Exercício - Factory

Discussão sobre o uso de Interface vs Herança

1. Comparação

Uma das grandes vantagens do uso de interfaces, é que ele não causa acoplamento, enquanto herança pode causar um grande acoplamento entre classes. Outra grande vantagem do uso de interfaces, é que a classe que a implementa, poderá usar o método da maneira que desejar, enquanto no exemplo da classe abstrata, se houver métodos já com escopo definido, a classe que herdar terá que usar o mesmo escopo.

Uma desvantagem que eu percebi foi que se eu adicionar uma assinatura de método na interface, todas as classes que implementam a interface, terá que adicionar o método, enquanto na classe abstrata não tem essa obrigação.

No site Devmedia, achei interessante essa comparação de características resumidas.

Característica	Interface	Classe Abstrata
Herança múltipla	Uma classe pode implementar diversas interfaces	Uma classe pode herdar somente uma classe
Implementação Padrão	Uma interface não pode conter qualquer tipo de código, muito menos código padrão.	Uma classe abstrata pode fornecer código completo, código padrão ou ter apenas a declaração de seu esqueleto para ser posteriormente sobrescrita.
Constantes	Suporte somente constantes do tipo estática.	Pode conter constantes estáticas e de instância.
Componentes de terceiros	Uma implementação de uma interface pode ser incluída a qualquer classe de terceiros.	Uma classe de terceiros precisa ser reescrita para estender somente a partir da classe abstrata.
Homogeneidade	Se todas as diversas implementações compartilham a assinatura do método então a interface funciona melhor.	Se as várias implementações são todas do tipo e compartilham um comportamento e status comum,

		então a classe abstrata funciona melhor.
Manutenção	Se o código do seu cliente conversa somente em termos de uma interface, você pode facilmente alterar a implementação concreta usando um método factory.	Idêntico
Velocidade	Lento, requer trabalho extra para encontrar o método correspondente na classe atual.	Rápido
Clareza	Todas as declarações de constantes em uma interface são presumidamente públicas ou estáticas.	Você pode por código compartilhado em uma classe abstrata. Você pode usar código para computar o valor inicial de suas constantes e variáveis de instância ou estáticas.
Funcionalidades Adicionais	Se você incluir um novo método em uma interface você precisa ajustar todas as implementações da interface.	Se você incluir um novo método em uma classe abstrata você tem a opção de fornecer uma implementação padrão para ele.

2. Conclusão

Em minha opinião, a escolha do que devemos usar, depende de caso para caso, mas acredito que na maioria dos casos, será mais adequado o uso de Interfaces, pois acoplamento tem um peso muito grande em qualquer projeto. As boas práticas de programação, prezam muito pelo baixo acoplamento, e isso acaba tendo um peso muito grande na escolha, porém como disse, é possível que possa existir casos específicos que uso de classes abstratas podem ser mais viáveis para o projeto.