

Java Avançado

Exercícios Propostos

Annotations e Reflection API

www.softblue.com.br

# Exercício

A integração entre annotations e a Reflection API é bastante utilizada na criação de frameworks e APIs para serem utilizadas por desenvolvedores. A ideia deste exercício é a criação de um framework simplificado que permite a criação de objetos e a invocação automática de métodos de acordo com as anotações presentes no mesmo.

O primeiro componente do framework é a anotação @Init. Ela pode ser utilizada somente em métodos e possui um elemento booleano chamado runOnInstantiation, que assume true por padrão.

Se determinado método for anotado com @Init e o elemento runOnInstantiation for true, ele deverá ser invocado logo após o objeto ser criado. Métodos que não têm a anotação ou o elemento runOnInstantiation possui valor *false* não devem ser invocados. Outro detalhe é que esta lógica é válida também se a classe possuir mais de um método anotado com @Init. Neste caso, cada método com a anotação e com runOnInstantiation com o valor true devem ser invocados.

O segundo componente do framework é a classe ObjectCreator. Ela possui apenas um método estático que possui a seguinte assinatura:

public static <T> T create(Class<T> clazz)

Este método recebe um objeto Class como parâmetro que define qual o tipo do objeto que será criado. O retorno do método create() é um objeto do tipo especificado pelo parâmetro clazz. Para criar este objeto a Reflection API deve ser utilizada.

Antes de retornar o objeto, este método deve procurar os métodos da classe anotados com @Init e invocá-los, caso seja necessário, de acordo com as regras já explicadas acima.

Crie uma classe com um método main() que testa o comportamento do framework.

**Dica:** Este exercício depende da utilização do generics para ser implementado da mesma forma que a resolução apresentada. Portanto é recomendado que os conceitos do generics sejam estudados antes de fazer este exercício.

# Exercício

Crie uma classe Property que possui métodos estáticos para atribuir e obter valores de atributos de uma classe através de chamadas aos métodos getters e setters correspondentes via reflexão.

Observe um exemplo:

Pessoa p = new Pessoa();

Property.set(p, "nome", "Pedro");

Este código cria um objeto da classe Pessoa e, na sequência atribui o valor “Pedro” ao atributo nome do objeto. O método set() deve fazer isto via reflexão, chamando o método setNome() no objeto p.

Para ler o valor do atributo funciona da mesma forma. Observe:

String nome = Property.get(p, "nome", String.class);

Agora o método get() retorna o valor do atributo *nome* do objeto p. O parâmetro String.class indica o tipo de dado a ser retornado, neste caso uma String.

**Dica:** Este exercício depende da utilização do generics para ser implementado da mesma forma que a resolução apresentada. Portanto é recomendado que os conceitos do generics sejam estudados antes de fazer este exercício.