



Java Avançado

Exercícios Propostos

Generics

1 Exercício

Implemente uma classe `Map` que possibilita mapear chaves a valores. Os tipos das chaves e valores são parametrizáveis (podem ser tipos diferentes) e devem ser fornecidos no momento da criação do objeto. Além de inserir um par de chave e valor, a classe deve permitir também a recuperação de um valor com base em uma chave.

Internamente, a classe `Map` deve ter duas listas (`ArrayList`), onde uma armazena o conjunto de chaves e a outra armazena o conjunto de valores. Quando um item é adicionado, ele deve ocupar o mesmo índice nas duas listas. Desta forma fica fácil fazer a recuperação do valor com base na chave (uma vez achado o índice na lista de chaves, basta ler o item que se encontra na mesma posição, mas na lista de valores).

Para que esta implementação seja feita com sucesso, é necessário ter cuidado com a inserção de elementos repetidos. Chaves iguais não são permitidas. Se o usuário tentar inserir uma chave já existente, o código deve identificar este cenário e substituir o valor existente da chave pelo novo valor fornecido.

A classe `Map` deve possuir os seguintes métodos públicos:

- `boolean put(K, V)`: Insere um par de chave e valor (`K` é a chave e `V` o valor). Se a chave não existia anteriormente, retorna `true`. Se ela já existia e o valor da chave foi atualizado, retorna `false`.
- `V get(K)`: Retorna o valor associado à chave fornecida como parâmetro `K`. Se a chave não existir, retorna `null`.
- `void clear()`: Remove todos os elementos do `Map`.

2 Exercício

Crie uma classe chamada `ServiceFactory` que possui um método estático chamado `newInstance()`. Este método recebe como parâmetro um objeto `Class` de algum tipo parametrizável, cria um objeto com base neste tipo (utilizando reflexão) e retorna o objeto recém-criado. Um detalhe importante é que apenas classes que implementam a interface `Service` (que também deve ser definida por você) devem ser aceitos na parametrização do tipo.