

Trabalho Prático 1

Observação: não deve ser utilizado o recurso de pacotes e de exceções em Java (isto será visto e requerido no próximo trabalho prático; tais recursos permitirão uma estruturação ainda mais modular das camadas da aplicação).

Requisitos gerais

Crie uma aplicação que auxilie o trabalho de um centro de ensino. Este centro oferece diversos cursos em áreas variadas. Os alunos, depois de cadastrados, podem se matricular nos cursos ofertados. A aplicação deve oferecer operações de cadastro (cadastrar, descadastrar, atualizar, procurar) para alunos e cursos. Além disso, a aplicação deve permitir a matrícula, cancelamento, e consulta (procura) de matrículas. Por fim, o sistema deve permitir a impressão de alguns relatórios relacionados a alunos, cursos, e matrículas.

Requisitos detalhados

Mais precisamente, a especificação sobre as informações que devem fazer parte do estado dos objetos das classes básicas de negócio é a seguinte. O estado do `Aluno` tem nome (`String`), endereço (`String`), telefone (`String`) e idade (`int`). O estado do `Curso` tem código (`String`), nome (`String`) e instrutor (`String`). O estado de uma `Matricula` tem aluno (`Aluno`), curso (`Curso`), e número (`int`), que deve ser uma constante (não-estática) .

Além das classes básicas, a aplicação deve ter uma fachada, três coleções de negócio (uma para cada repositório), três repositórios (uma para cada classe básica de negócio), e uma ou mais classes de interface textual com o usuário. A instanciação das classes básicas de negócio deve ocorrer apenas na(s) classe(s) de interface com o usuário. Tal instanciação deve fornecer a tais objetos todas informações necessárias para que eles tenham o estado completo, conforme descrito anteriormente. Para tal, a interface com o usuário deve solicitar ao usuário tais informações. Em particular, com relação à instanciação de `Matricula`, o número (`int`) que faz parte do estado da matrícula deve ser inicializado chamando-se um método estático desta classe, que incrementa um atributo estático desta classe, a partir de zero.

As coleções de negócio devem implementar as seguintes regras de negócio:

- não é permitido cadastrar aluno ou curso mais de uma vez;
- não é permitido ao aluno se matricular mais de uma vez no mesmo curso.

A matrícula precisa do nome do aluno e, opcionalmente, do código do curso. Se apenas o nome do aluno é fornecido, a matrícula é feita no curso com menos alunos matriculados. Além disso, não se deve permitir matricular alunos se alguma das seguintes duas condições ocorrerem: 1) o código do curso (caso especificado) não é código de curso algum; 2) o nome do aluno é tal que ele não esteja cadastrado.

O descadastramento e a procura de aluno/curso devem ter como parâmetro o nome do aluno ou o código do curso. **Adicionalmente, o descadastramento de um aluno ou de um curso implica o descadastramento de todas as matrículas que envolvam o aluno ou o curso descadastrado.**

Já a atualização de aluno/curso deve ter como parâmetro um novo objeto aluno/curso, cujo estado deve ser preenchido com informações especificadas pelo usuário à interface textual da aplicação. Tal objeto deve ser utilizado como base para alterar o estado do objeto correspondente previamente existente no cadastro. A procura e o cancelamento de matrícula devem especificar o nome do aluno e o nome do curso.

Por fim, a aplicação deve ter poder emitir as seguintes consultas específicas:

- lista de todas as matrículas feitas;
- lista de todos os alunos cadastrados;
- lista de todos os cursos cadastrados;
- lista de todos os alunos de um curso cadastrado (o código do curso é parâmetro);
- lista de todos os alunos de todos os cursos cadastrados;
- lista de todos os cursos de um aluno cadastrado (o nome do aluno é parâmetro);
- lista de todos os cursos de todos os alunos cadastrados;
- lista de todas as matrículas de um aluno cadastrado (o nome do aluno é parâmetro);
- lista de todas as matrículas de um curso cadastrado (o código do curso é parâmetro);
- lista dos alunos cadastrados sem curso, ou seja, lista de todos os alunos que não têm matrícula em algum curso;
- lista dos cursos cadastrados sem alunos, ou seja, lista de todos os cursos que não têm matrícula de aluno algum.

Comportamento da interface com o usuário

A aplicação deve exibir a seguinte tela ao usuário mostrando as opções de operação. O usuário deverá fornecer uma opção; após isso, a aplicação deve perguntar a ele os dados necessários para a execução da operação; a aplicação, então, executa a operação selecionada pelo usuário e imprime novamente a tela com opções esperando nova seleção de opção pelo usuário, e assim por diante. A aplicação deve terminar a execução apenas quando o usuário selecionar a opção de número 23.

*** Centro de Ensino XYZ ***

Operações disponíveis:

1- Cadastrar aluno
2- Procurar aluno
3- Atualizar aluno
4- Descadastrar aluno

5- Cadastrar curso
6- Procurar curso
7- Atualizar curso
8- Descadastrar curso

9- Matricular aluno
10- Procurar matrícula
11- Cancelar matrícula

12- listar todas as matrículas feitas;
13- listar todos os cursos cadastrados;
14- listar todos os alunos cadastrados;
15- listar todos os alunos de um curso;
16- listar todos os alunos de todos os cursos;
17- listar todos os cursos de um aluno;
18- listar todos os cursos de todos os alunos;
19- listar todas as matrículas de um aluno;
20- listar todas as matrículas de um curso;
21- listar alunos cadastrados sem curso;
22- listar cursos cadastrados sem alunos.

23- SAIR

Favor escolher uma opção: _

Você deve definir na interface com o usuário um método estático com a assinatura

```
public static void reportar_erro(String mensagem_de_erro)
```

que deve ser chamado em todos os lugares da aplicação (ou seja, em qualquer contexto da aplicação) em que se deseje indicar uma situação de erro de negócio, de repositório, ou de interface com o usuário. Consequentemente, qualquer chamada ao método `System.out.println()` somente pode ocorrer em duas situações: 1) dentro do método `reportar_erro` ;2) para imprimir a interface textual exibida acima.