

Lista de Exercícios 5: Introdução a classes e objetos

Questão 1

As datas são representações construídas para registrar os dias em que vivemos. O atual calendário que utilizamos, o Gregoriano, possui 365 dias e, de 4 em 4 anos, 366 dias. Os dias são separados em 12 meses. Vamos criar uma classe chamada Data que possui três atributos: o dia (inteiro), o mês (inteiro) e o ano (inteiro). Os atributos devem ser privados. Crie os métodos setters e getters e declare o construtor padrão que cria a data 01/01/1970.

Questão 2

Qualquer instituição de ensino precisa guardar informações sobre os seus estudantes. Você está modelando um sistema para a sua instituição de ensino e algumas informações sobre os estudantes precisam ser representadas: a matrícula, o nome completo da(o) estudante, cpf, rg, nome das(os) genitoras(es), data de nascimento (use aqui um variável do tipo Data que você criou na questão 1). Os atributos precisam ser privados. Crie os métodos setters e getters.

Questão 3

Todo mundo tem o seu time do coração. Tem gente que torce para um time de verdade (como o Bahia, por exemplo) e tem gente que sonha em torcer para um time (como os torcedores do Vitória). A verdade é que o amor ao time é algo que nasce e a gente nem sabe por onde. Dito isso, você deve criar uma classe chamada Time que possui um nome e uma data de fundação. Os atributos precisam ser privados. Crie os métodos setters e getters.

Questão 4

O sonho de qualquer jogador da base é virar profissional, ter sua chance como titular. Quem não pensa nisso? Mas um jogador só é um jogador quando ele tá num time. Sendo assim, vamos criar uma classe chamada Jogador que possui um nome, um número de camisa e o time que ele joga. Os atributos precisam ser privados. Crie os métodos setters e getters.

Questão 5

Você criou a classe Data na primeira questão, mas não definiu nenhum construtor, sem ser o padrão. Crie um construtor que inicialize a data, o mês e o ano que a programador que quiser usar a data resolver informar.

Questão 6

Criar um objeto de uma classe deve ser algo fácil, mas deve ser definido pelo programador que criou a classe os vários métodos de se criar um objeto. Você vai definir os vários modos de se criar objetos do tipo Estudante. Como?! Você vai definir um construtor que inicialize a matrícula, o nome completo da(o) estudante, cpf, nome da genitora e a data de nascimento; outro construtor que inicialize a matrícula, o nome completo da(o) estudante, cpf, rg, nome da genitora e a data de nascimento; e, por último, um construtor que inicialize a matrícula, o

nome completo da(o) estudante, cpf, rg, nome das(os) genitoras(es) 1 e 2 e a data de nascimento.

Questão 7

Se não for criado, o time do coração não vai existir. Defina três construtores na classe Time: o construtor padrão, o construtor que inicializa apenas o nome e coloca a data de fundação como 01/01/1970 e o construtor que inicializa o nome do time e a sua data de fundação.

Questão 8

Você já sabe quem vai ganhar construtores, né? O Jogador, é claro! Vamos criar dois construtores para o jogador: o padrão e o construtor que inicializa o nome, o número da camisa e o time que ele vai jogar.

Questão 9

Como saber se duas datas são iguais? Primeiro a gente sabe que com variáveis de objeto não dá para o usar o operador ==, afinal variáveis de objeto guardam como valor o endereço na memória heap da Máquina Virtual do Java (**Java Virtual Machine - JVM**). Ou seja, dois objetos podem ter os mesmo valores nos atributos, mas estarem em endereços diferentes na memória. Então, como resolver isso? Nós sobrescrevemos um comportamento dos objetos que se chama **equals** (**igual em inglês**). É um método que toda classe deve definir para sabermos se um objeto é igual ou não ao outro. O que você vai fazer? Você vai definir o método **equals** para a classe Data. (Conselho: faça esse código você mesmo. Procure na internet sobre a palavra reservada **instanceof** e para quê ela serve. Esse método vai lhe dar a base de quase toda abstração que vamos precisar daqui para frente. Esforce-se bastante em fazer por conta própria, anote todas as suas dúvidas).

Questão 10

Achou que eu ia pedir para fazer o método equals da classe Estudante, não foi? Quase acertou, mas eu quero mesmo é que você faça o método equals da classe Time.

Questão 11

Crie o método equals da classe Jogador, por favor. Não esqueça que para comparar dois times usamos o método equals definido na classe Time.

Questão 12

Como imprimir um objeto? Como eu faço para fazer a seguinte linha de código: `System.out.println(data)` e não aparecer o endereço armazenado na variável de objeto data, mas sim os valores armazenados nos seus atributos? Nós sobrescrevemos um comportamento dos objetos que se chama **toString()** (para String, em inglês). É um método que toda classe deve definir para sabermos como imprimir em texto os dados armazenados no objeto. O que você acha que vai fazer? Você vai definir o método **toString** para a classe Data. Teste se esse método está funcionando como esperado, criando vários objetos e imprimindo-os no método main de uma classe.

Questão 13

Crie um método toString para as classes Estudantes, Time e Jogador.