

Aluno: \_\_\_\_\_ Nota: \_\_\_\_\_

## IIª Avaliação – 2024.2

As questões desta avaliação são baseadas no seguinte cenário de classes:

Uma empresa que trabalha com soluções corporativas resolveu desenvolver um **framework** para construção de aplicativos de gestão de calendários. A solução evoluiu para um cenário, onde deseja-se criar aplicações diferentes, contemplando tipos de calendários e de eventos diferentes, alguns deles ainda não definidos, nem implementados.

Visando simplificar o trabalho do desenvolvedor, a equipe de projeto utilizou o padrão Factory Method, de forma que, para construir a sua aplicação personalizada, precisa-se apenas herdar da classe **AplicacaoCalendario**, fornecendo o respectivo método fábrica. Além do método fábrica, os códigos de personalização da aplicação devem também ser inseridos nesta classe (no diagrama de classes do modelo, exibido na figura 1, representada pela classe **Aplicacao**).

O framework fornece ainda diversas classes utilitárias (**Evento**, **Calendario** e classes derivadas), além de Builders para sua construção (os Builders foram omitidos no diagrama de classes, por questões de simplificação).

É hora de evoluir o **framework** através da adaptação do projeto atual de forma a contemplar novos requisitos demandados pela área de projeto.

Deseja-se ver incorporados, os seguintes requisitos:

**QUESTÃO I:** Na construção da aplicação, deseja-se possibilitar que alguns calendários possam dinamicamente incorporar eventos oriundos de agendas online. Por exemplo, um calendário, quando da chamada dos métodos **from(Timestamp, Timestamp)**, **day(Day)** e **today()**, deve retornar, além dos eventos nele armazenados, os feriados municipais retornados por um site; enquanto, em outro momento, deverá incorporar além dos feriados municipais, os feriados estaduais retornados por um segundo site; ou ainda, numa terceira situação, deve retornar feriados municipais, estaduais, nacionais e ainda jogos de uma determinada equipe (cada um deste conjunto de eventos, tendo sua informação vinda de um site diferente).

Em um Calendário, os métodos **from**, **day** e **today** retornam, respectivamente, os Eventos iniciados entre os timestamps indicados, os Eventos que ocorram num determinado dia e os Eventos que ocorram hoje.

Para a conexão com o site e o retorno dos Eventos use o código a seguir:

```
CalendarioOnline online = new CalendarioOnline(uri);  
Collection<Evento> eventos = online.consultar(timestamp1, timestamp2); // ou online.consultar(day);
```

**QUESTÃO II:** O sistema acadêmico do IFBa possui uma classe que representa a aula de uma disciplina. Um extrato do código dessa classe é exibido abaixo. Deseja-se permitir que esta classe possa ser utilizada como um Evento no nosso framework. Neste caso a descricao do Evento Aula deve ser “**Aula de codDisciplina – descricaoDisciplina**”. Considere que, quando não for possível ser mais específico, a Aula possui os valores default indicados para um Evento padrão (indicados na classe AbstractEvento).

```
public class Aula {  
    public String getCodDisciplina(){...};  
    public String getDescricaoDisciplina(){...};  
    public String getProfessor(){...};  
    public Geolocalizacao getSala(){...};  
    public LocalDateTime[] getHorario(){...}; // retorna o horário de inicio e fim da aula  
}
```

Identifique e escreva os padrões adequados para resolver os problemas apresentados.

### ATENÇÃO

- As soluções devem ser esboçadas em código JAVA
- Caso modifique as relações entre as classes do modelo, apenas apresente a nova assinatura da classe, por exemplo:  
`public class Classe extends ClasseAbstrata implements Interface1, Interface2{}`
- Identifique claramente o padrão que será aplicado em cada questão, bem como quais são os papéis (Participantes) no padrão, assumidos pelas classes do projeto.

Figura I – Diagrama de Modelo do Framework

